

CBPF
CENTRO BRASILEIRO DE PESQUISAS FÍSICAS

Dissertação de Mestrado Profissional em Física
com Ênfase em Instrumentação Científica

IGOR MARQUES CARNEIRO DA SILVA

**DESENVOLVIMENTO DE UM AMPLIFICADOR
LOCK-IN DIGITAL BASEADO EM
MICROCONTROLADOR**

Prof. Dr. PABLO DINIZ BATISTA
Orientador

CBPF
CENTRO BRASILEIRO DE PESQUISAS FÍSICAS
2018

DESENVOLVIMENTO DE UM AMPLIFICADOR LOCK-IN DIGITAL BASEADO EM MICROCONTROLADOR

IGOR MARQUES CARNEIRO DA SILVA

Dissertação apresentada ao Programa de Pós-Graduação Stricto Sensu no Centro Brasileiro de Pesquisas Físicas como requisito parcial para obtenção do título de Mestre em Física com Ênfase em Instrumentação Científica.

Prof. Dr. PABLO DINIZ BATISTA

Orientador

Rio de Janeiro, RJ

2018

DEDICATÓRIA

A minha esposa, por ter permanecido ao meu lado, me incentivando a percorrer este caminho, por compartilhar angústias e dúvidas.

Agradecimentos

Primeiramente, gostaria de agradecer ao G.. A.. D.. U.., pela vida e a possibilidade de empreender esse caminho evolutivo. Por propiciar tantas oportunidades de estudos e por colocar em meu caminho pessoas amigas e preciosas.

Um grande obrigado à MINHA FAMÍLIA, especialmente a minha esposa e incondicional companheira Bárbara e minha filha Alice.

Ao meu irmão, José Eduardo Marques Carneiro, mesmo estando a alguns quilômetros de distância, se manteve incansável em suas manifestações e orientações. E também, não menos importante, à minha irmã, Carolina, um grande obrigado.

À MINHAS MÃEs e MEU PAI.

AO MEU ORIENTADOR, Dr. Pablo Diniz Batista, pela confiança, pela oportunidade de trabalhar ao seu lado, e por ser o incentivador na superação dos meus limites.

Agradeço também ao Rogério Passy, não só, por proporcionar os meios para que eu pudesse completar esta jornada. Mas também pelas orientações neste trabalho.

Um especial obrigado ao Moroni Guahyba Alexandre Barbosa Pedrosa, um amigo distante, mentor, fonte de inspiração pela eletrônica, minha imensa gratidão.

AO PROGRAMA DE MESTRADO PROFISSIONAL DO CBPF, todos os professores que fizeram parte desse caminhar.

Enfim, a todos aqueles que de uma maneira ou de outra contribuíram para que este percurso pudesse ser concluído.

“Só existem dois dias no ano que nada pode ser feito. Um se chama ontem e o outro se chama amanhã, portanto hoje é o dia certo para amar, acreditar, fazer e principalmente viver.”

— DALAI LAMA

Resumo

O amplificador Lock-in é uma excelente ferramenta para medir sinais em níveis inferiores aos níveis de ruído branco, até mesmo ruído intrínseco. Permite separar sinais em fase e em quadratura, fornecendo valiosas informações sobre diferentes grandezas físicas dos sinais medidos. Atualmente, no mercado, existem diversos fabricantes oferecendo tanto Lock-in analógicos quanto digitais. No entanto, nenhum com tecnologia nacional, tornando difícil a aquisição deste tipo de instrumento, devido a processos burocráticos e altos custos de importação. Esta dissertação tem como objetivo desenvolver um Amplificador Lock-in, baseado em microcontrolador, no modelo Open-Source Hardware, voltado para instrumentação científica. Para isto, a proposta é fazer a atualização de uma placa de aquisição de dados, desenvolvida anteriormente pelo grupo. Substituindo principalmente o microcontrolador para o dsPIC33FJ128GP706A. Implementando uma fonte simétrica de $\pm 5V$ a partir dos 5V da interface USB. Inserindo um “offset” no sinal de entrada permitindo medidas de tensões negativas. Além da instalação de outros recursos como, DDS, DACs, portas de comunicação e acesso a diferentes tensões da placa. Tornando este equipamento mais flexível para outras aplicações na instrumentação científica além de executar uma aplicação do Lock-in. Por fim, este trabalho descreve detalhadamente diferentes etapas, dentre elas, a simulação do Lock-in feita no computador e o desenvolvimento de dois protótipos, além da execução de experimento optoeletrônico para a medida de concentração de glicose, pela rotação do plano de polarização da luz, a fim de validar a aplicação do Amplificador Lock-in.

Palavras-Chave: *amplificador Lock-in, Microcontrolador, DMA, Open-Source Hardware, Instrumentação Científica, Aquisição de Dados*

Abstract

The Lock-in amplifier is an excellent tool to measure signals at levels below white noise levels, even intrinsic noise. It allows to separate signals in phase and quadrature, providing valuable information on different physical quantities of the measured signals. Currently in the market there are several manufacturers offering both analog and digital Lock-in. However, none with national brazilian technology, making difficult to acquire this type of instrument due to bureaucratic processes and high import costs. This dissertation aims to develop a Lock-in Amplifier, based on microcontroller, under Open-Source Hardware model, focused on scientific instrumentation. For this, the proposal is to upgrade a data acquisition board, previously developed by the group. Replacing mainly the microcontroller for the dsPIC33FJ128GP706A. Implementing a symmetric source $\pm 5V$ from 5V of USB port. By inserting an “offset” into the input signal allowing negative voltage measurements. In addition to the installation of other features such as DDS, DACs, communication ports and access to different board voltages. Making this equipment more flexible for other applications in scientific instrumentation beyond executing a Lock-in application. Finally, this work describes in detail different steps, among them, the simulation of the Lock-in made in the computer and the development of two prototypes. In addition to the execution of optoelectronic experiment for the measurement of glucose concentration, rotation of the polarization plane of the light, in order to validate the application of the Lock-in Amplifier.

Keywords: *Lock-in Amplifier, Microcontroller, DMA, Open-Source Hardware, Scientific Instrumentation, Data Acquisition.*

Lista de Figuras

FIGURA 1.1 – Amplificador Lock-in analógico comercial SRS Modelo SR530. . . .	20
FIGURA 1.2 – Amplificador Lock-in digital da Zurich Instruments, modelo UHFLI, que alcança frequência de 600 MHz.	21
FIGURA 1.3 – Simulando o efeito do detector sensível a fase. (a) Referência e sinal em fase produzem uma forma de onda totalmente positiva; (b) Referência e sinal defasados de 90 graus produzem uma forma de onda onde a média ao longo do tempo tende a zero (SYSTEMS, 1999).	23
FIGURA 1.4 – Diagrama em blocos da arquitetura de um Amplificador Lock-in (SYSTEMS, 1999).	27
FIGURA 1.5 – Componente em fase X, componente em quadratura Y e magnitude R.	28
FIGURA 2.1 – Diagrama em blocos do hardware de aquisição de dados tendo como base o microcontrolador dsPIC33FJ128GP706A da Microchip. . . .	33
FIGURA 2.2 – Circuito utilizando o conversor DC/DC RP-0505D para fornecimento da alimentação de +5V e -5V simétricos.	34
FIGURA 2.3 – esquemático da alimentação de 3,3V	34
FIGURA 2.4 – Modos de operação do ADC. Em a) a amostragem é feita no modo síncrono, lendo ao mesmo tempo todos os canais; Em b) a amostragem é feita um canal por vez sequencialmente. Fonte: (MICROCHIP, 2012)	35
FIGURA 2.5 – Projeto da placa no Altium Design: em a) modelo do circuito para a fabricação da placa; em b) simulação 3D mostrando uma perspectiva futura da placa.	36
FIGURA 2.6 – Foto do primeiro protótipo	37
FIGURA 2.7 – Esquemático do segundo RP-0505D instalado em paralelo.	37

FIGURA 2.8 – Opção de “offset” inserida na entrada do amplificador INA121.	38
FIGURA 2.9 – Microcontrolador PIC18F14K50, responsável pela conversão RS-232 para USB.	39
FIGURA 2.10 – Layout superior a) e inferior b) da placa desenhada utilizando o Altium.	39
FIGURA 2.11 – Visualização 3D da placa.	40
FIGURA 2.12 – Foto do segundo protótipo.	40
FIGURA 2.13 – Código para gerar o sinal referência.	42
FIGURA 2.14 – Código para gerar o sinal de interesse.	42
FIGURA 2.15 – Código simulando um PLL.	43
FIGURA 2.16 – Código simulando um Lock-in Digital.	44
FIGURA 2.17 – Sistema de medidas montado para caracterização do ADC.	44
FIGURA 2.18 – Diagrama proposto para o experimento utilizando a placa final do Javanês desenvolvida.	45
FIGURA 2.19 – Preparo das amostras no Laboratório de Produção de Materiais Avançados (LPMA) do CBPF.	46
FIGURA 2.20 – Sistema de medidas experimental de medida de glicose.	46
FIGURA 2.21 – Configuração do Clock do microcontrolador.	47
FIGURA 2.22 – Pausa para aguardar a mudança de Clock.	47
FIGURA 2.23 – Configuração do Timer do microcontrolador.	48
FIGURA 2.24 – Configuração da DMA.	49
FIGURA 2.25 – Configuração da interrupção da DMA.	50
FIGURA 3.1 – Resultado das medições do Lock-in no computador.	52
FIGURA 3.2 – Caracterização do ADC variando a frequência.	53
FIGURA 3.3 – Caracterização do ADC variando a Amplitude; a) Sinal com 1,9V , b) 1,2V, c) 1V; d) 600 mV; e) 200 mV; f) 100 mV.	54
FIGURA 3.4 – Sinal da amostra com ruído simulado.	55
FIGURA 3.5 – Sinais seno e cosseno gerados a partir do PLL.	55
FIGURA 3.6 – Resultado final da simulação com todos os parâmetros do Lock-in.	56

FIGURA 3.7 – Distribuição espectral do sinal medido, comparada com a análise de Fourier feita pelo Origin.	57
FIGURA 3.8 – Análise da resposta em frequência em dB.	58
FIGURA 3.9 – Em (a) temos a relação da potência óptica aplicada e a tensão de saída do sistema; Em (b) estabilidade ao longo do tempo.	59
FIGURA 3.10 – Análise da variação da potência óptica com a Lei de Malus.	60
FIGURA 3.11 – Resultado das medições utilizando o Lock-in sendo executado no microcontrolador.	61
FIGURA 3.12 – Distribuição espectral do sinal medido.	61
FIGURA 3.13 – Análise da resposta em frequência em dB.	62
FIGURA 3.14 – Em (a) temos a relação da potência óptica aplicada e a tensão de saída do sistema; Em (b) estabilidade ao longo do tempo.	63
FIGURA 3.15 – Análise da variação da potência óptica com a Lei de Malus	64
FIGURA 3.16 – Resultados obtidos utilizando o polarímetro.	65
FIGURA 3.17 – ângulo de rotação óptico em função da concentração amostra.	66
FIGURA D.1 – Esquema do tubo porta amostra para o polarímetro. Fonte (FAUTH, 2017)	94
FIGURA D.2 – Esquema de preparo das soluções de amostras. Fonte (FAUTH, 2017)	95

Lista de Tabelas

TABELA 2.1 – Tabela de custos da aquisição dos componentes do segundo protótipo.	41
TABELA 2.2 – Outros custos com a fabricação da placa.	41
TABELA 3.1 – Estabilidade da Potência Óptica.	59
TABELA 3.2 – Resposta em frequência.	62
TABELA 3.3 – Estabilidade da Potência Óptica.	63

Lista de Abreviaturas e Siglas

DC	Direct current
AC	Alternate current
Msp/s	Mega Samples per Second
USB	Universal Serial Bus
LAPS	Laboratório de Eletrônica e processamento de sinais do CBPF
IGBT	Insulated Gate Bipolar Transistor
ADC	Analog-to-digital
DMA	Direct access memory
CPU	Central processing unit
PCB	Printed circuit board
I/O	Input/Output
PLL	Phase-locked loop
DDS	Direct Digital Synthesis
DAC	Digital-to-analog
DSP	Digital Signal Processor
PSD	Phase Sensitive Detector
OSI	Open Source Initiative
ICSP	In-Circuit Serial Programming
DIP	Dual In-Line Package

Sumário

1	INTRODUÇÃO	16
1.1	Rotação do plano de polarização da luz	16
1.2	O Amplificador Lock-in	18
1.2.1	Lock-in Analógico	19
1.2.2	Lock-in Digital	20
1.2.3	Lock-in Digital Baseado em DSP	21
1.3	Detector Sensível a Fase	22
1.3.1	Princípio de funcionamento	22
1.4	Ruído	25
1.5	Estado da arte do Lock-in	26
1.6	Open-Source Hardware	29
1.7	Objetivos	31
2	METODOLOGIA	32
2.1	Desenvolvimento do hardware	32
2.1.1	Descrição do hardware	32
2.1.2	Alimentação simétrica	33
2.1.3	Tensão de alimentação do microcontrolador	34
2.1.4	ADC e DMA	35
2.2	Fabricação dos protótipos	36
2.2.1	Fabricação do primeiro protótipo	36
2.2.2	Fabricação do segundo protótipo	37
2.3	Simulando um Lock-in no computador	42

2.4	Caracterização e Validação	44
2.5	Desenvolvimento do código do Lock-in	47
3	RESULTADOS	52
3.1	Caracterização	52
3.2	Validação	54
3.2.1	Simulando um Lock-in no computador	54
3.2.2	Lock-in implementado no computador	56
3.2.3	Lock-in embarcado no microcontrolador	60
3.2.4	Aplicação em medida de glicose	64
4	DISCUSSÃO E CONCLUSÃO	67
5	PERSPECTIVAS	69
	REFERÊNCIAS	70
	APÊNDICE A – CÓDIGO SIMULAÇÃO NO COMPUTADOR	74
	APÊNDICE B – FIRMWARE DO MICROCONTROLADOR	76
	APÊNDICE C – DIAGRAMAS DO CIRCUITO FINAL	90
	APÊNDICE D – PREPARO DAS AMOSTRAS E PORTA AMOSTRA . .	94
	APÊNDICE E – APLICAÇÕES DO AMPLIFICADOR LOCK-IN	96

1 Introdução

O amplificador Lock-in é uma excelente ferramenta para medir sinais em níveis inferiores aos níveis de ruído branco¹, até mesmo ruído intrínseco. Permite separar sinais em fase e em quadratura, fornecendo valiosas informações sobre diferentes grandezas físicas associadas às componentes Real e Imaginária dos sinais medidos (SYSTEMS, 1999; MEADE, 1983).

Primeiramente este capítulo introduz os conceitos básicos da técnica de medida da rotação do plano de polarização da luz a partir de um sistema optoeletrônico como forma de aplicação física da técnica. Em seguida apresentaremos o principal objetivo deste trabalho, o Amplificador Lock-in.

Para isto, discutiremos os tipos conhecidos de Lock-in e as suas diferenças. Finalmente, apresenta, dentro da perspectiva da instrumentação científica com ênfase em Física, o objetivo dessa dissertação de mestrado.

1.1 Rotação do plano de polarização da luz

A partir dos estudos de Maxwell pode-se entender a luz como uma onda eletromagnética composta por campos elétricos e magnéticos oscilantes, perpendiculares entre si e transversais à direção de propagação da onda. A direção de oscilação da componente elétrica define a direção de polarização da onda.

Podemos descrever a luz não polarizada, de maneira simplificada, como sendo constituída por duas ondas linearmente polarizadas em direções perpendiculares, em que cada onda representa a soma de todas as componentes da luz em um eixo. Uma dessas direções pode ser convenientemente escolhida como sendo a direção de polarização, ou seja, a direção em que a absorção de luz pelo polarizador é praticamente nula (NUSSENZVEIG, 2018; FAUTH, 2017).

A Polarimetria é uma das técnicas para a medida da rotação do plano de polarização.

¹Ruído branco é um sinal aleatório com igual intensidade em diferentes frequências, o que lhe dá uma densidade espectral de potência constante (STEIN, 2012).

Através do polarímetro, é capaz de medir o ângulo de rotação do plano de polarização da luz quando esta passa através de um material. Este equipamento ainda é o mais simples e mais utilizado para se determinar a pureza óptica de um líquido, gás ou solução de um composto opticamente ativo (LIMA, 1997; FAUTH, 2017). Por exemplo, o açúcar refinado é composto de cerca de 99% de sacarose (CARLIN *et al.*, 2005), sendo assim, as usinas de açúcar utilizam o polarímetro até hoje para avaliar a quantidade de sacarose presente em uma amostra durante o processo de fabricação.

Conforme discutido anteriormente, a luz linearmente polarizada pode ser decomposta em duas componentes circularmente polarizadas: uma à direita (D) e outra à esquerda (L). Como os índices de refração dependem da concentração (C), temperatura (T) e do comprimento de onda (λ), obtemos a rotação óptica específica do açúcar a partir da equação 1.1, reescrevendo-a de uma maneira mais conveniente:

$$\theta_R = [a]_{\lambda}^T L \cdot C \quad (1.1)$$

Em que $[a]_{\lambda}^T$ é uma constante dado em grau, L em decímetros e C em g/mL. Podemos dizer que $[a]_{\lambda}^T L = \alpha$, e reescrever a equação 1.2:

$$\theta_R = \alpha L \cdot C \quad (1.2)$$

Em que:

- T é a temperatura em °C.
- λ é o comprimento de onda da luz do espectro de emissão, geralmente referenciada ao Sódio (598 nm).
- θ_R é a rotação do plano de polarização, medida em graus.
- L é o caminho óptico em decímetros.
- C é a concentração em gramas por mL.

A magnitude da rotação causada pela amostra depende do número de moléculas opticamente ativas que a luz encontra na amostra. Isto, por sua vez, depende da concentração da amostra, da temperatura, do comprimento do tubo (caminho óptico), e do comprimento de onda da fonte de luz do polarímetro.

Cada composto opticamente ativo tem uma rotação específica característica. A rotação óptica específica para a sacarose é de aproximadamente 67° quando utilizada sob fonte

luminosa de sódio (598 nm) a temperatura de 20°C. Atribuímos a divergência no resultado ao diferente comprimento de onda da luz utilizado no experimento.

Existem três gerações de instrumentos de polarimetria: a primeira referente aos polarímetros visuais, utilizando a luz do sol; a segunda referente aos fotopolarímetro utilizando luz contínua e sistemas eletroeletrônicos; e a terceira sendo os fotopolarímetros em conjunto com um sistema de detecção mais sensível, como por exemplo o sistema de detecção síncrona a partir de um amplificador Lock-in. Embora existam outros métodos de medida, o Lock-in é utilizado frequentemente em sistemas de medidas de precisão (CHEN *et al.*, 2012; CHOU *et al.*, 1997; FAUTH, 2017).

1.2 O Amplificador Lock-in

O amplificador Lock-in, desde a sua invenção no início da década de 1960 (BURDETT, 2005), também conhecido como, “Voltímetros de Frequência Seletiva” ou “Analisadores de Espectro de Único Canal” (CAPITANIO, 2008), se tornou um instrumento extremamente valioso e útil, largamente presente em laboratórios de física experimental (BOCHNER *et al.*, 2011). Ele pode ser utilizado em muitas aplicações, tais como: instrumento medidor de fase, unidade medidora de ruído, analisador de espectro, experimentos ópticos de baixo nível, medições de impedância complexa (D’AMICO *et al.*, 2010; INSTRUMENTS, 2000; CAPITANIO, 2008), espectroscopia (AUGULIS; ZIGMANTAS, 2011), pesquisa neurológica, contagem de fótons (RESTELLI *et al.*, 2005) entre outras (SCOTT, 2002). Existem também na literatura diversos projetos de amplificador Lock-in para fins específicos, como Lock-ins para sensores potenciométricos endereçáveis de Luz (LAPS) (FERRI *et al.*, 2001); Para detectar pequena quantidade de gás, através de um sensor resistivo (MARCELLIS *et al.*, 2007); Para tomografia de detecção por coerência óptica de sinal (XU *et al.*, 2008); Para a detecção de sinais de pequena amplitude magneticamente em ressonadores mecânicos estimulados (AZZOLINI *et al.*, 2008); Para sensores ópticos de espectroscopia (HU; CHO-DAVARAPU, 2010; SANTHANALAKSHMI, 2016) e para detectores em imagens de terahertz (XU, 2012). No Apêndice E encontra-se uma lista de outras aplicações conhecidas para o amplificador Lock-in.

O funcionamento do amplificador Lock-in é baseado no conceito do Detector Sensível a Fase (PSD)², método eficaz de recuperar sinal de pequena amplitude imerso em ruído (INSTRUMENTS, 2000; CAPITANIO, 2008). O detector PSD é capaz de alcançar amplificação em uma largura de banda estreita, reduzindo drasticamente o ruído que esteja fora da banda de interesse. Ou seja, incluindo apenas a frequência que ocorre o sinal e excluindo as frequências nas quais ocorrem ruído (BLAIR; SYDENHAM, 1975; SYSTEMS, 1999).

²PSD, do inglês Phase Sensitive Detector

Conceitualmente, o amplificador Lock-in tem como base a propriedade de ortogonalidade das funções senoidais. Quando as funções senoidais são multiplicadas e integradas durante uma duração significativa de tempo, o resultado tenderá a zero, a menos que as duas funções senoidais tenham a mesma frequência.

Os primeiros instrumentos utilizavam tecnologia analógica, com controles manuais. Mais tarde, os microprocessadores foram adicionados para fornecer novas operações, telas de saída digital e controle pelo computador (SONNAILLON; BONETTO, 2005). Mais recentemente, os detectores sensíveis à fase (analógico), sendo o coração do instrumento, foram substituídos por projetos que fazem uso do Processamento de Sinal Digital (DSP)³.

Atualmente no mercado existem diversos fabricantes oferecendo tanto Lock-in analógicos quanto digitais, no entanto, nenhum com tecnologia nacional. Isso torna difícil a aquisição deste tipo de instrumento, devido aos processos burocráticos e altos custos de importação (CASTILHO, 2014; MOSCATI *et al.*, 1982).

Segundo um texto de Eugênio Lerner, Coordenador de Ciências Exatas e da Terra da Universidade de São Paulo. *“É opinião unânime da comunidade científica brasileira a necessidade de se tomar ações no sentido de fortalecer ao máximo o desenvolvimento de instrumentação científica nacional. O desenvolvimento autônomo, dentro das instituições de pesquisa, de técnicas de automatização de controle e aquisição de dados é de suma importância para a atividade experimental e tem tido sérios obstáculos devido ao excessivo zelo de órgãos governamentais que controlam e coordenam a importação de componentes e instrumentos básicos. Existe ainda uma real necessidade de se importar equipamentos, componentes e certas matérias-primas de vários tipos para realizar pesquisas no Brasil”* (MOSCATI *et al.*, 1982).

1.2.1 Lock-in Analógico

Por um longo tempo os amplificadores Lock-in eram apenas analógicos (INSTRUMENTS, 2000). Com o avanço da tecnologia, a eletrônica digital foi introduzida utilizando microprocessadores, embora utilizados somente como suporte para algumas funções. Posteriormente, os filtros de saída foram implementados usando técnicas digitais, mas o PSD continuou a utilizar circuitos analógicos.

A Figura 1.1 mostra um Amplificador Lock-in analógico comercial da Stanford Research Systems, modelo SR530, que opera a uma frequência máxima de 100kHz e com uma reserva dinâmica⁴ de 80dB.

³DSP, do inglês Digital Signal Processor

⁴Muito usado em estudos de Lock-in, a reserva dinâmica é a razão entre a capacidade total do sinal



FIGURA 1.1 – Amplificador Lock-in analógico comercial SRS Modelo SR530.

Amplificadores Lock-in que usam um canal de processamento analógico são conhecidos como instrumentos analógicos, mesmo incluindo filtros de saída digitais (INSTRUMENTS, 2000). O termo “amplificador Lock-in digital” refere-se às unidades que utilizam um demodulador em software.

1.2.2 Lock-in Digital

O surgimento dos amplificadores Lock-in digitais não é recente, surgindo há mais de 30 anos (CAPITANIO, 2008). Estes amplificadores são normalmente descritos como “digitais”, ou instrumentos em DSP, que utilizam Processador Digital de Sinais (INSTRUMENTS, 2000). Isso porque o PSD é implementado em circuito digital, removendo grande quantidades de componentes do circuito (WENN, 2007). No entanto, na etapa de aquisição do sinal, os amplificadores Lock-in digitais ainda precisam ser implementados com tecnologia analógica. Para isso, existe o conversor ADC⁵, responsável em fazer o elo entre o sistema analógico e o digital, convertendo o sinal analógico de entrada para digital com determinada frequência de amostragem. A saída deste conversor é uma série de valores digitais representando a amplitude do sinal de entrada, alimentando assim o processador digital de sinais.

A maioria dos amplificadores Lock-in atuais são baseados em DSP. Ao longo dos últimos 20 anos, os amplificadores Lock-in digitais foram substituindo modelos analógicos, permitindo aos usuários realizarem medições até a frequência de 600 MHz (INSTRUMENTS, 2017). Além disso, o uso de circuitos digitais fornece uma série de benefícios adicionais, incluindo maior confiabilidade, resistência à temperatura e efeitos de envelhecimento, e a facilidade com que o sistema pode ser atualizado (WENN, 2007).

A Figura 1.2 mostra o Amplificador Lock-in digital UHFLI, da Zurich Instruments. Este equipamento cobre uma faixa de frequências desde o nível DC a 600 MHz e possui uma reserva dinâmica de 100dB, além de fornecer uma frequência maior de operação entre outros equipamentos comerciais.

e o nível de ruído. Diferente do SNR que é a relação entre o nível médio do sinal e o nível de ruído rms (ANDRETSKY *et al.*, 1999).

⁵ADC, do Inglês Analog Digital Converter



FIGURA 1.2 – Amplificador Lock-in digital da Zurich Instruments, modelo UHFLI, que alcança frequência de 600 MHz.

Os amplificadores Lock-in digitais de hoje superam os modelos analógicos em diversos parâmetros de desempenho relevantes, como faixa de frequência, ruído de entrada e estabilidade. O amplificador Lock-in digital também pode incluir múltiplos demoduladores, o que permite analisar um sinal com diferentes configurações de filtro ou várias frequências diferentes simultaneamente. Além disso, alguns modelos são controlados por computador e possuem interface gráfica do usuário (BURDETT, 2005; MEADE, 1983). Há também a vantagem da reserva dinâmica ser limitada apenas pela qualidade do conversor ADC (NOTE, 1992).

1.2.3 Lock-in Digital Baseado em DSP

A maioria dos Amplificadores Lock-in digitais comerciais são baseados em DSPs. Sendo os microprocessadores projetados para calcular, eficientemente, um grande número de operações matemáticas por segundo, esta e outras características o tornam uma opção conveniente para processar sinais digitalmente (SONNAILLON; BONETTO, 2005).

A incorporação de um microprocessador para monitorar e comandar as funções de um amplificador Lock-in, representa um avanço significativo no conceito evolutivo da técnica. O resultado final é um amplificador Lock-in autônomo, com a capacidade de realizar operações de mudanças dos seus próprios parâmetros via software (MEADE, 1983). Por exemplo, as rotinas do software embarcado permitem que o amplificador Lock-in ajuste a sensibilidade e a fase automaticamente para maximizar a saída de um determinado sinal. No que diz respeito ao usuário, o amplificador Lock-in agora funciona como um tipo especial de um microvoltímetro AC, que pode ler a amplitude e a fase de um sinal em resposta a um comando remoto.

Desta forma, o sistema é então menos propenso ao erro do usuário do que um sistema analógico, esse equipado com uma escala de ponteiro em que o perigo de negligenciar um

fator de escala está sempre presente (MEADE, 1983). Além disso, uma vez que todas as operações de comando são supervisionadas pelo microprocessador, o sistema pode inibir ou dar um aviso de combinações indesejáveis.

Em suma, a incorporação de um microprocessador tem um impacto cada vez maior à medida que a complexidade do sistema aumenta. É possível reduzir o nível desta complexidade operacional, convertendo em esforço de software (MEADE, 1983).

1.3 Detector Sensível a Fase

Como mencionado, o coração do amplificador Lock-in é o detector sensível a fase, também conhecido como demodulador.

O Detector Sensível a Fase (PSD) é extensamente utilizado na instrumentação científica para fornecer informação precisa sobre o sinal requerido, “imerso” em ruído, que pode ser de caráter aleatório ou determinístico.

A técnica utilizada pelo PSD consiste de multiplicar um sinal de entrada por um sinal de referência, podendo ser implementado de três maneiras: Multiplicador Analógico, Multiplicador Chaveado Digitalmente ou Multiplicador Digital.

Na multiplicação analógica, o PSD compreende um circuito eletrônico que multiplica o sinal aplicado e uma onda senoidal de mesma frequência que a aplicada no sinal de referência. Embora a técnica a princípio seja muito simples, na prática é difícil fabricar um multiplicador analógico que seja capaz de operar linearmente na presença de alto ruído ou outros sinais de interferência. A operação não linear resulta em pobre rejeição de ruído e assim limita a capacidade de recuperação de sinal.

O multiplicador chaveado digitalmente consiste de uma chave analógica de polaridade reversa controlada pela frequência de referência aplicada. A grande vantagem desta abordagem é que é muito mais fácil de fazer tal demodulador operar linearmente sobre uma faixa larga de sinais de entrada. Entretanto, o multiplicador de chaveamento não somente detecta sinais na frequência de referência aplicada, mas também suas harmônicas ímpares.

No multiplicador digital, o sinal de entrada é amplificado e então imediatamente digitalizado. Esta representação digital é então multiplicada por uma representação digital de uma onda senoidal na frequência de referência em um DSP.

1.3.1 Princípio de funcionamento

Três sinais são mostrados na Figura 1.3 (a). O primeiro é o sinal de entrada, um sinal senoidal livre de ruído. O segundo sinal corresponde ao sinal de referência. O terceiro

signal é o resultado da multiplicação dos sinais anteriores, identificado como “Saída do demodulador”.

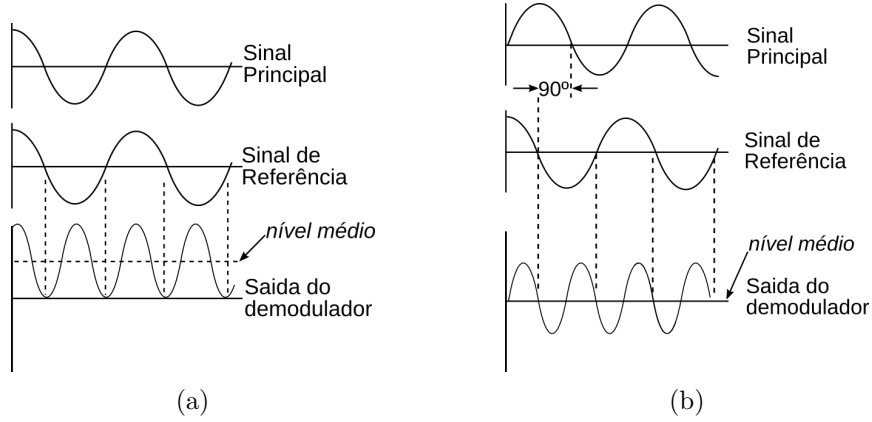


FIGURA 1.3 – Simulando o efeito do detector sensível a fase. (a) Referência e sinal em fase produzem uma forma de onda totalmente positiva; (b) Referência e sinal defasados de 90 graus produzem uma forma de onda onde a média ao longo do tempo tende a zero (SYSTEMS, 1999).

Quando não há mudança de fase relevante entre o sinal principal e a referência, a saída do demodulador tem a forma de um sinal senoidal, com o dobro da frequência de referência e com um nível médio de amplitude positivo.

A Figura 1.3 (b) mostra a mesma situação, exceto que a fase do sinal de entrada é agora atrasada em 90° com respeito à referência. Neste caso, pode ser observado que embora a saída ainda contenha um sinal com o dobro da frequência de referência, o nível médio ao longo do tempo tende a zero. Uma demonstração matemática destas conclusões pode ser vista a seguir.

Considerando um sinal senoidal livre de ruído, representado pela tensão V_{Sinal} , onde $\omega = 2\pi F$, e um sinal de referência representado por V_{Ref} , temos que:

$$V_{Sinal} = A \cos(\omega t) \quad (1.3)$$

$$V_{Ref} = B \cos(\omega t + \theta) \quad (1.4)$$

$$V_L = A \cos(\omega t) \cdot B \cos(\omega t + \theta) \quad (1.5)$$

Sendo:

A : Amplitude do sinal de interesse;

B : Amplitude do sinal de referência;

ω : Frequência angular dada em rad/s ;

t : Tempo (s);

F : Frequência do sinal em Hz ;

θ : Fase em graus.

O processo do detector sensível a fase consiste na multiplicação destas duas componentes, sendo a saída dada por V_L (INSTRUMENTS, 2000).

$$\begin{aligned}
 V_L &= A \cos(\omega t) \cdot B \cos(\omega t + \theta) \\
 &= AB \cos(\omega t) \cdot \cos(\omega t + \theta) \\
 &= AB \cdot (\cos(\omega t) \cos(\omega t) \cos(\theta) - \cos(\omega t) \sin(\omega t) \sin(\theta)) \\
 &= AB \cdot (\cos^2(\omega t) \cos(\theta) - \cos(\omega t) \sin(\omega t) \sin(\theta)) \\
 &= AB \cdot \left(\frac{1 + \cos(2\omega t)}{2} \cos(\theta) - \frac{1}{2} \sin(2\omega t) \sin(\theta) \right) \quad (1.6) \\
 &= \frac{AB}{2} \cdot (\cos(\theta) + \cos(2\omega t) \cos(\theta) - \sin(2\omega t) \sin(\theta)) \\
 &= \frac{AB}{2} \cdot (\cos(\theta) + \cos(2\omega t + \theta)) \\
 V_L &= \frac{AB}{2} \cos(\theta) + \frac{AB}{2} \cos(2\omega t + \theta)
 \end{aligned}$$

A saída será proporcional ao produto da amplitude do sinal principal e da referência, diretamente relacionadas ao ângulo de fase entre o sinal principal e a referência, que também será modulado ao dobro da frequência de referência (INSTRUMENTS, 2000). Assim, se a amplitude do sinal de referência for mantida em um valor fixo, e a fase de referência for ajustada para assegurar uma diferença de fase próximo de zero grau, a máxima amplitude do sinal principal poderá ser determinada, após passar por um filtro passa-baixa. A componente $2\omega t$ tenderá a zero.

A discussão acima é baseada no caso em que o sinal principal esteja livre de ruído. No entanto, em aplicações reais, o sinal principal está acompanhado por ruído que, por definição, não possui frequência fixa ou relação de fase com o sinal de referência. Neste caso, o ruído, ao ser multiplicado pelo sinal de referência no demodulador, não resultará em nenhuma contribuição no nível DC após o filtro passa-baixa. A Equação 1.7 mostra uma situação em que sinais com frequências diferente são multiplicados (ALONSO *et al.*,

2003).

$$\begin{aligned}
 V_L &= A \cos(\omega_{in} t) \cdot B \cos(\omega_{ref} t + \theta) \\
 &= A \cos(\omega_{in} t) \cdot B [\cos(\omega_{ref} t \cdot \cos(\theta) - \sin(\omega_{ref} t) \cdot \sin(\theta)] \\
 V_L &= \frac{AB}{2} \cos[(\omega_{in} + \omega_{ref})t + \theta] + \frac{AB}{2} \cos[(\omega_{in} - \omega_{ref})t + \theta]
 \end{aligned} \tag{1.7}$$

Obviamente, ruído em frequências muito próximas da referência, resultará na saída em frequências muito baixas. Isto ocorre como visto na Equação 1.7, devido ao fato de que o termo da diferença de fase na equação, ou seja, θ é aproximadamente zero. Ao passar por um filtro com longo tempo de integração, estes podem ser atenuados. Daí a combinação de um demodulador e um filtro passa-baixa, permitindo que os sinais sejam medidos mesmo quando acompanhado de ruído significativo (SYSTEMS, 1999).

1.4 Ruído

Ruído é qualquer variação no tempo de algum parâmetro físico. Que não produz informações úteis para o observador. No processamento de sinal, o ruído está presente durante a captura, armazenamento, transmissão, processamento e conversão (OTT; OTT, 1988).

Segundo um estudo relatado em (INSTRUMENTS, 2000), é possível obter uma comparação da técnica PSD com as técnicas tradicionais de aquisição de sinais. Em qualquer método de amplificação de sinal, a presença de ruído intrínseco do amplificador é inevitável e diretamente proporcional a frequência, especificado como $nV/\sqrt{Hz}$.

$$A_{saida} = S_{entrada} \cdot G_{ganho} \tag{1.8}$$

$$A_{ruído} = (R_{densidade} \cdot \sqrt{BW}) \cdot G_{ganho} \tag{1.9}$$

Sendo:

A_{saida} : Amplitude do sinal de saída em V;

$S_{entrada}$: Sinal de Entrada em V;

G_{ganho} : Ganho;

$A_{ruído}$: Amplitude do ruído de saída em V;

$R_{densidade}$: Densidade de ruído em $nV / \sqrt{Hz}$;

BW : Largura de banda do sinal em Hz;

Para ilustrar, suponha que se deseja amplificar um sinal de 10 nV em uma frequência de 10 kHz , aplicando um ganho de 1000 vezes e fixando a largura de banda em 100 kHz . Suponha ainda um amplificador com baixo ruído intrínseco, da ordem de $5 \text{ nV}/\sqrt{\text{Hz}}$. Substituindo nas Equações 1.8 e 1.9, teremos a seguinte saída :

$$A_{\text{saida}} = 10 \cdot 10^{-9} \cdot 1000 = 10 \text{ } \mu\text{V}$$

$$A_{\text{ruído}} = (5 \cdot 10^{-9} \cdot \sqrt{100 \cdot 10^3}) \cdot 1000 = 1,58 \text{ mV}$$

Deste modo, podemos dizer que o ruído é aproximadamente 158 vezes maior que o sinal. Claramente, é impossível medir o sinal de interesse na presença desse nível de ruído, a menos que seja feito algo para isolá-lo. Um meio de conseguir isso é o uso de um filtro antes do amplificador.

Supondo que o mesmo sinal primeiro passasse por um filtro passa-banda com largura de banda de 500Hz (uma especificação que é difícil de alcançar), qualquer sinal dentro dessa largura de banda de 500Hz será detectado. Aplicando a Equação 1.9, o ruído dentro dessa largura de banda será:

$$A_{\text{ruído}} = (5 \cdot 10^{-9} \cdot \sqrt{500}) \cdot 1000 = 111,80 \text{ } \mu\text{V}$$

Ainda assim, o ruído é aproximadamente 10 vezes maior que o sinal. Neste caso, para medir um sinal 10 vezes menor que o ruído, pode-se utilizar um amplificador Lock-in. Ele pode detectar o mesmo sinal, no entanto, com uma largura de banda de ruído na ordem de $0,125\text{Hz}$ (INSTRUMENTS, 2000).

1.5 Estado da arte do Lock-in

O amplificador Lock-in é essencialmente um correlator, no qual o sinal de saída é altamente dependente do grau de correlação que há entre o sinal de referência e o sinal de interesse (MEADE, 1982). Usando a teoria de Fourier, qualquer sinal de entrada acompanhado de ruído pode ser representado por uma soma de muitas ondas senoidais de diferentes amplitudes, fases e frequências (MEADE, 1983).

Basicamente, na entrada do amplificador Lock-in tem-se o sinal de interesse do experimento, já amplificado e imerso em ruído, e um sinal de referência que esteja na mesma fase e frequência do sinal que se deseja medir. Este sinal de referência pode estar na forma senoidal, quadrada ou qualquer outra forma desde que seja um sinal periódico. Este sinal servirá apenas como base para que o PLL⁶ (KUZNETSOV *et al.*, 2011) obtenha a fase e a

⁶PLL, Phase Locked Loop

frequência do sinal de referência. E, a partir disso, seja possível gerar um sinal senoidal já digitalizado.

Na sequência, tem-se o PSD, onde as componentes do sinal de interesse e a referência são multiplicados. Deste modo, segundo o princípio de ortogonalidade, estando apenas o sinal de interesse na frequência do sinal de referência a amplitude do sinal medido será máxima quando as fases dos sinais multiplicados coincidirem exatamente. Caso contrário, este valor será atenuado sendo levado à zero em caso de existir uma diferença de 90° .

Para solucionar isto, é introduzido um segundo PSD como mostra a Figura 1.4. Neste modelo, o Lock-in é chamado de “Lock-in em Fase Dupla”. Este segundo PSD é então alimentado com o mesmo sinal de entrada do primeiro, mas com o sinal de referência defasado de 90° . Ou seja, é realizado o produto ortogonal do sinal de entrada pelo sinal de referência em quadratura, o que produzirá a chamada componente em quadratura.

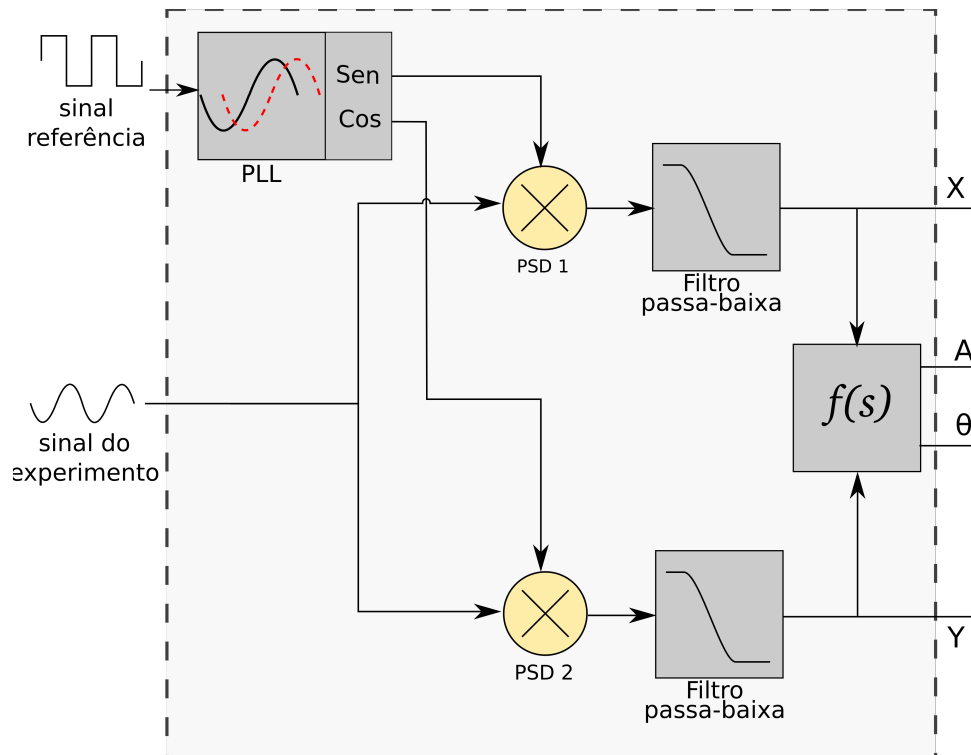


FIGURA 1.4 – Diagrama em blocos da arquitetura de um Amplificador Lock-in (SYSTEMS, 1999).

Em seguida, o resultado de ambos os produtos passam por filtros passa-baixa, obtendo assim uma componente em fase e a outra em quadratura (SONNAILLON; BONETTO, 2005). Obviamente que uma saída é proporcional ao $\cos(\theta)$ e a outra proporcional ao $\sin(\theta)$ como mostra a Figura 1.5.

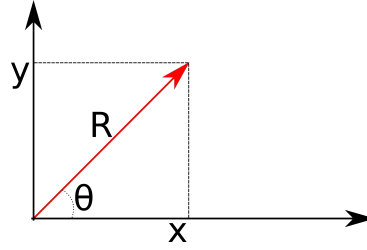


FIGURA 1.5 – Componente em fase X, componente em quadratura Y e magnitude R.

Matematicamente podemos dizer que para um sinal de referência senoidal $ref(t)$, e um sinal de entrada $V_{sinal}(t)$, a saída DC pode ser calculada a partir da seguinte expressão:

$$V(t) = \frac{1}{T} \int_{t-T}^t \sin(2\pi f_{ref} * s + \varphi) V_{sinal}(s) ds \quad (1.10)$$

onde φ é a diferença de fase do Lock-in (preferencialmente zero).

Se o tempo de integração (T) for grande suficiente para remover o ruído indesejável, a saída pode ser escrita por:

$$V = \frac{1}{2} V_{sinal} \cos(\theta) \quad (1.11)$$

Para um Lock-in de Dupla Fase o mesmo cálculo é feito, porém defasado de 90° . Obtendo assim:

$$X = \frac{1}{2} V_{sinal} \cos(\theta) , \text{ componente em fase}$$

$$Y = \frac{1}{2} V_{sinal} \sin(\theta) , \text{ componente em quadratura}$$

Onde V_{sinal} é a amplitude do sinal medido e θ é a diferença de fase entre o sinal de entrada e a referência.

Uma das vantagens do Lock-in de Dupla Fase é que, se a fase do sinal medido mudar em relação à referência e sua amplitude for a mesma, embora a saída de um dos PSD diminua, a diferença será compensada pela saída do outro PSD, dada pela Equação 1.12.

$$R = \sqrt{X^2 + Y^2} \quad (1.12)$$

Além disso, a fase pode ser obtida por:

$$\theta = \tan^{-1} \left(\frac{Y}{X} \right) \quad (1.13)$$

Com isso, é possível obter a amplitude (R) do sinal medido, a fase (θ) e as componentes X e Y com a máxima precisão sem a necessidade de efetuar a correção da diferença de fase.

1.6 Open-Source Hardware

Hardware livre (do inglês, Open source hardware), é um hardware eletrônico projetado e oferecido da mesma maneira que um software de código livre. O termo foi primeiramente empregado para refletir o lançamento irrestrito de informação sobre o projeto de hardware, tal como um diagrama, estrutura de produtos e dados de layout de uma placa de circuito impresso (GREGO, 2009).

Embora não exista uma definição clara e amplamente aceita de Hardware livre (GREGO, 2009), a iniciativa código aberto (OSI)⁷ mantém o que eles chamam de “Open Source Definition”, embora tenha sido criado sob a perspectiva de software. Uma definição razoável é encontrado na TAPR Open Hardware License (LICENSE, 2007):

“Hardware livre é um artefato físico, elétrico ou mecânico - cuja informação de design está disponível e acessível para o público de uma forma que permite que qualquer pessoa faça, modifique, distribua e use.”

O movimento de software livre teve enorme impacto na tecnologia de hoje. Vemos isso cada vez mais nos últimos anos. Ele ajudou a pesquisa acadêmica, mudou a forma como muitas empresas de tecnologia fazem negócios e mudou a nossa sociedade. Mais recentemente (embora não pela primeira vez), a ideia de hardware livre tem recebido muita atenção (RUBOW, 2008). Vários projetos de hardware livre surgiram ao longo dos anos. Alguns deles são:

Arduíno

O Arduíno (ARDUINO,) é uma plataforma de prototipagem de código livre para microcontroladores ATmega. Os projetos de hardware estão disponíveis no “Creative Commons Attribution-Share Alike 2.5 license”. Eles também fornecem bibliotecas de software e ferramentas sob LGPL e GPL.

Muitos sensores são criados na plataforma de desenvolvimento do Arduíno para o desenvolvimento de projetos de automatização e armazenamento de dados, de baixo custo para uso em projetos de pesquisa em diversas áreas da indústria, como por exemplo, projetos relacionados a agricultura e recursos naturais (FISHER; GOULD, 2012).

JOP: Java Optimized Processor

⁷OSI, The Open Source Initiative (OSI) is a California public benefit corporation

Iniciando como parte de uma tese de doutorado, o Java Optimized Processor (JOP,) é a implementação do “Java Virtual Machine” em hardware. O núcleo foi implementado em FPGAs. O tempo de execução é apropriado para aplicações em tempo real.

Portanto, o desenvolvimento de soluções pautadas no conceito Open-source Hardware tem um grande potencial na pesquisa científica, podendo munir pesquisadores com ferramentas flexíveis e de baixo custo para expandir suas capacidades de coleta de dados, automação e controle (FISHER; GOULD, 2012).

1.7 Objetivos

Esta pesquisa tem como objetivo principal o desenvolvimento de um Amplificador Lock-in digital, baseado no microcontrolador dsPIC33FJ128GP706A, no modelo Open-Source Hardware, voltado para instrumentação científica.

Dentro dessa perspectiva, a dissertação propõe desenvolver uma nova versão de uma placa de aquisição de dados denominada Javanês, criada pelo grupo no Laboratório de Eletrônica e Processamento de Sinais (LEPS). Isso inclui algumas melhorias no hardware, principalmente na substituição do microcontrolador central para um que tenha Acesso Direto a Memória (do inglês Direct Memory Access - DMA), obtendo maior velocidade na aquisição de dados em dois canais sincronizados com uma frequência máxima de 50kHz.

É importante ressaltar que esse projeto é uma continuação de outros projetos do grupo realizados a partir de 2013, com alterações propostas a fim de substituir um amplificador Lock-in comercial em um experimento optoeletrônico para medidas de rotação do plano de polarização da luz (no qual foi utilizado o Lock-in MFLI, da Zurich Instruments). Também a fim de medir a intensidade luminosa usando um fotodetector em conjunto com outros componentes ópticos (FAUTH, 2017), integrando assim os diversos trabalhos do grupo.

O capítulo 2 desta dissertação apresenta a metodologia e materiais utilizados neste trabalho; o capítulo 3 apresenta os resultados experimentais; o capítulo 4 apresenta a conclusão e o capítulo 5 apresenta as perspectivas futuras para este projeto.

2 Metodologia

2.1 Desenvolvimento do hardware

O objetivo deste capítulo, é mostrar o planejamento das mudanças necessárias no projeto do Javanês para atender as necessidades do objetivo principal do projeto.

2.1.1 Descrição do hardware

Como dito no capítulo anterior, para a concepção deste trabalho, optamos por dar continuidade a um projeto do grupo (LEPS) que teve início em 2011 por (FILHO, 2012). Foi desenvolvido um módulo eletrônico para a caracterização elétrica de EGFETs, baseado no microcontrolador PIC18F45K20¹, posteriormente denominado “Javanês”. Tal circuito foi projetado de modo a permitir a substituição deste microcontrolador em futuras aplicações. No entanto, ele foi projetado para ler até seis canais, utilizando um PGA MCP6S26 que, além de amplificador, opera como multiplexador. Esta e outras características tiveram de ser alteradas para que fosse possível implementar um amplificador Lock-in.

Embora existam outras formas possíveis de implementar um Lock-in em um microcontrolador, é necessário considerar qual a melhor arquitetura do processador. Neste caso, o microcontrolador escolhido foi o dsPIC33FJ128GP706A ².

Esse microcontrolador tem uma velocidade de processamento de 40 MIPS, capaz de trabalhar a uma taxa de conversão de 1 Msps. Isto é suficiente para fazer a aquisição de 2 canais de ADC sincronizados com uma taxa de amostragem de 500kHz em cada canal; Possui acesso a periféricos de comunicação como: UART, necessário para comunicação com o PC e SPI, necessário para comunicação com outros componentes da placa. Possui também acesso direto à memória (DMA)³ imprescindível neste projeto para obter maior taxa de transferência de dados (WENN, 2007).

A Figura 2.1 mostra um diagrama em blocos dos recursos propostos para a atualização

¹PIC18F45K20 - Microcontrolador de 16-bit da Microchip

²dsPIC33FJ128GP706A - Componente da família de dsPIC de 16-bits da Microchip

³DMA, Direct Memory Access

da placa Javanês, tendo como base o microcontrolador dsPIC33FJ128GP706A.

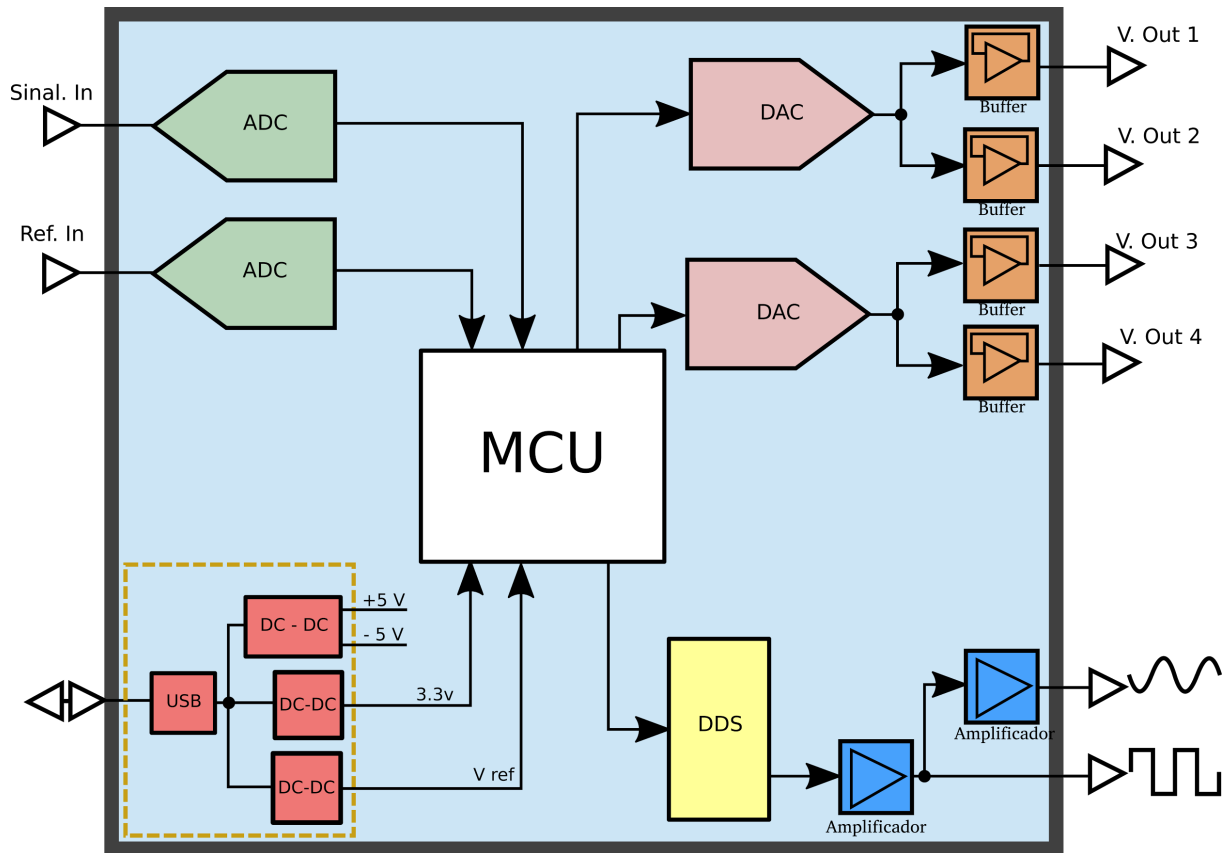


FIGURA 2.1 – Diagrama em blocos do hardware de aquisição de dados tendo como base o microcontrolador dsPIC33FJ128GP706A da Microchip.

2.1.2 Alimentação simétrica

Para que seja possível utilizar a alimentação de 5V da porta USB, foi necessário inserir um componente que fornecesse uma tensão de $\pm 5V$ simétricos para a alimentação negativa dos amplificadores. Para isso, foi inserido no projeto o RP-0505D um conversor DC/DC baseado em IGBT (*Insulated Gate Bipolar Transistor*), capaz de fornecer duas alimentações simétricas de +5V e -5V a partir da interface USB, que tem uma alimentação de 5V, como pode ser visto na Figura 2.2.

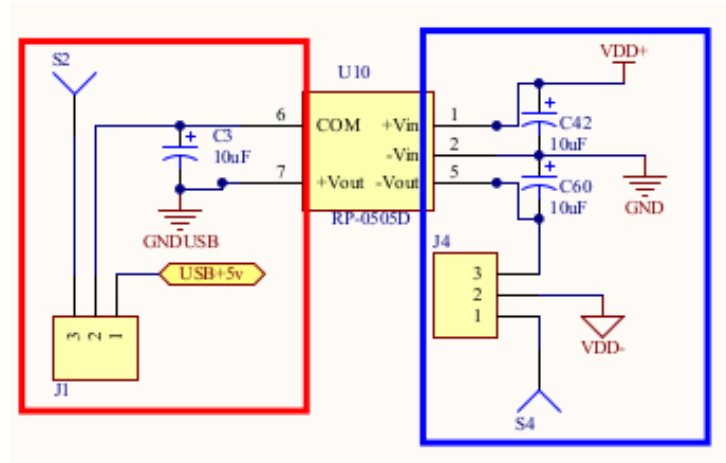


FIGURA 2.2 – Circuito utilizando o conversor DC/DC RP-0505D para fornecimento da alimentação de +5V e -5V simétricos.

2.1.3 Tensão de alimentação do microcontrolador

O microcontrolador dsPIC33FJ128GP706A necessita de uma alimentação de 3,3V. Para isso, foi necessário incluir no projeto o regulador ajustável LM317 para fornecer os 3,3V ao microcontrolador, como mostra a Figura 2.3. O cálculo para definir a tensão desejada é mostrado na Equação 2.1.

$$V_o = V_{REF} \left(1 + \frac{R2}{R1} \right) + (I_{ADJ} * R2) \quad (2.1)$$

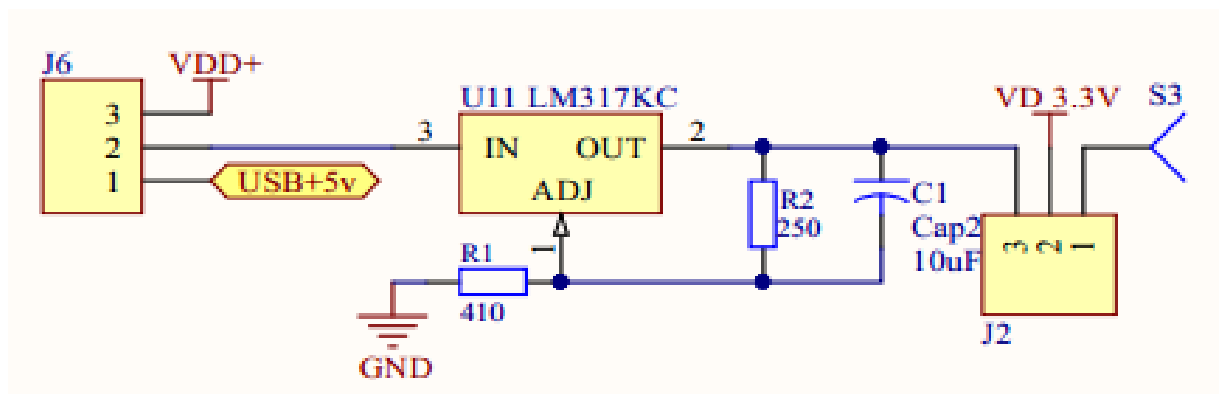


FIGURA 2.3 – esquemático da alimentação de 3,3V

2.1.4 ADC e DMA

A escolha do microcontrolador dsPIC33FJ128GP706A tem como principal razão a velocidade de conversão de até 1Msps, suficiente para a aquisição em 50kHz. O ADC possui a configuração de 10 e 12 bits de resolução. No entanto, por características desse microcontrolador somente a opção de 10 bits permite o modo síncrono (MICROCHIP, 2012). Deste modo para atender os requisitos do projeto foi necessário configurar o ADC no modo síncrono e com 10 bits de resolução.

Embora a maioria dos microcontroladores com ADC possam medir valores sequencialmente, o ADC deste microcontrolador pode ler até 4 canais de uma só vez, como mostra a Figura 2.4. Isto é importante para garantir o sincronismo dos canais, por exemplo, para uma aplicação de um Lock-in de fase única.

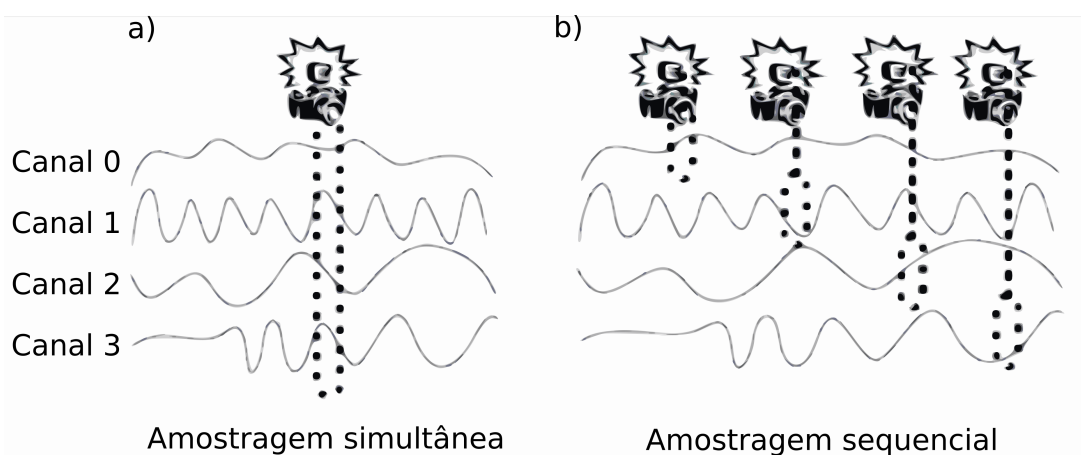


FIGURA 2.4 – Modos de operação do ADC. Em a) a amostragem é feita no modo síncrono, lendo ao mesmo tempo todos os canais; Em b) a amostragem é feita um canal por vez sequencialmente. Fonte: (MICROCHIP, 2012)

Outra característica decisiva na escolha do microcontrolador é o suporte à DMA. Este recurso é importante para realizar tarefas de forma independente do processador principal, ou seja, a transferência de dados entre a CPU e outro periférico sem a assistência direta da CPU (MICROCHIP, 2012). Em outras palavras, não interrompendo qualquer rotina que esteja eventualmente sendo executada por outra instrução. Como dito no capítulo anterior, o recurso de DMA é imprescindível para obter uma maior taxa de transferência de dados na aquisição.

2.2 Fabricação dos protótipos

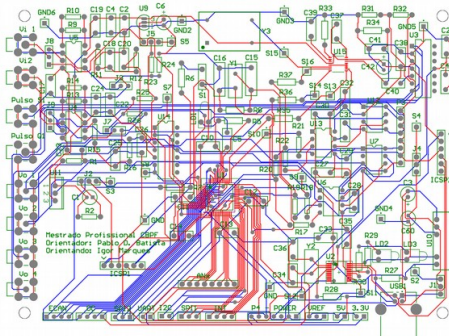
O objetivo deste tópico é mostrar a atualização da placa de aquisição de dados (Java-nês) e suas etapas de protótipos. Duas placas eletrônicas foram desenvolvidas durante o projeto.

2.2.1 Fabricação do primeiro protótipo

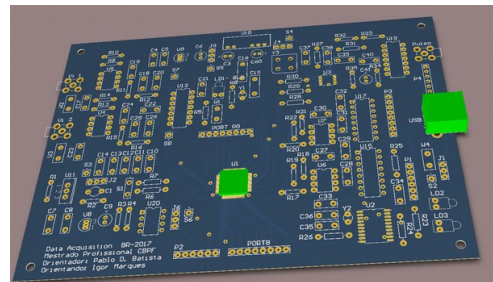
A primeira concepção da placa teve como principal modificação a substituição do microcontrolador principal, além de outras implementações sugeridas. Dentre elas, houve a inclusão de uma fonte simétrica de $\pm 5V$, duas entradas de ADC com amplificadores e diversas saídas, tornando a placa de aquisição de dados mais flexível para outros experimentos de bancada.

A Figura 2.5 mostra o modelo para a fabricação do primeiro protótipo da placa PCB⁴ desenvolvida no programa Altium Design⁵. Devido à complexidade do circuito, as placas foram desenhadas em duas camadas. Na Figura 2.5 a) as linhas vermelha e azul representam, respectivamente, as trilhas superior e inferior da placa.

Para melhor análise de como ficaria a disposição dos componentes na placa, foi desenvolvido um modelo 3D, no próprio programa Altium, como mostra a Figura 2.5 b).



(a)



(b)

FIGURA 2.5 – Projeto da placa no Altium Design: em a) modelo do circuito para a fabricação da placa; em b) simulação 3D mostrando uma perspectiva futura da placa.

Após a conclusão do primeiro circuito, optou-se por enviar o projeto para fabricação por empresa especializada. A soldagem dos componentes SMD também foi realizada por empresa, obtendo assim uma qualidade superior no menor tempo de fabricação. Alguns componentes foram soldados no CBPF. Finalmente, é possível visualizar na Figura 2.6 uma foto da placa montada.

⁴PCB - Printed Circuit Board

⁵Altium - É um software de criação de placas de circuito impresso.

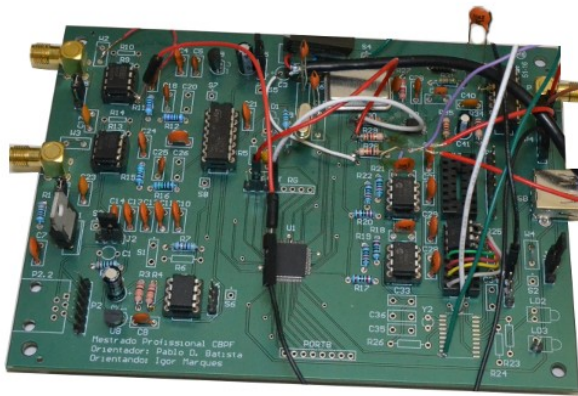


FIGURA 2.6 – Foto do primeiro protótipo

2.2.2 Fabricação do segundo protótipo

Após testes feitos com o primeiro protótipo, foi identificada uma deficiência no circuito da fonte simétrica de $\pm 5V$. Segundo o fabricante, o limite de corrente do conversor DC/DC é de 100mA, o suficiente para a alimentação de todos os componentes da placa. No entanto, notou-se que com uma corrente de 90mA o conversor eleva a temperatura acima do indicado pelo fabricante. Sendo assim, foi decidido incluir uma opção, através de “jumpers”⁶ para alimentar apenas os amplificadores que utilizam alimentação simétrica, permanecendo os outros componentes diretamente ligados à alimentação da interface USB. Também foi inserido no projeto do segundo protótipo a opção de instalação de um segundo RP-0505D em paralelo, como pode ser visto na Figura 2.7, a fim de aumentar o limite da corrente, caso necessário em aplicações futuras.

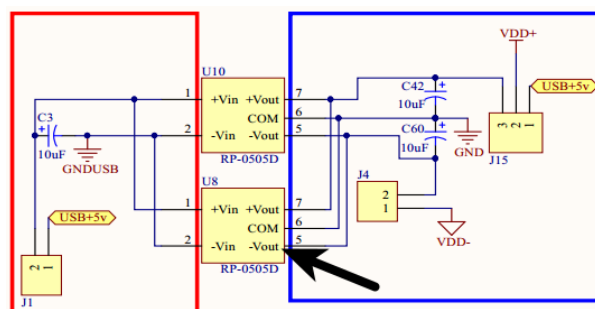


FIGURA 2.7 – Esquemático do segundo RP-0505D instalado em paralelo.

Notou-se também um erro conceitual com a configuração da tensão de referência negativa no microcontrolador. Pensou-se que o valor de referência negativa poderia ser de $-2,48V$, simétricos aos $+2,48V$. No entanto, o microcontrolador não permite tensões negativas em seus pinos de I/O inviabilizando assim este projeto, pelo qual é necessário

⁶Jumpers - É um dispositivo usado para fechar ou abrir parte de um circuito eletrônico.

que o sinal de entrada seja simétrico, ou seja, capaz de ler valores positivos e negativos a partir do amplificador INA121.

Para solucionar isto, utilizou-se o pino 5 do amplificador INA121 que permite que seja aplicada uma tensão de “offset” deslocando a faixa do nível de medida para cima. Deste modo, no segundo protótipo, foi inserido esta opção de “offset” nos sinais de entrada em cada amplificador INA121, como pode ser visto na Figura 2.8. Deste modo, as tensões positivas e negativas podem ser convertidas pelo ADC do microcontrolador.

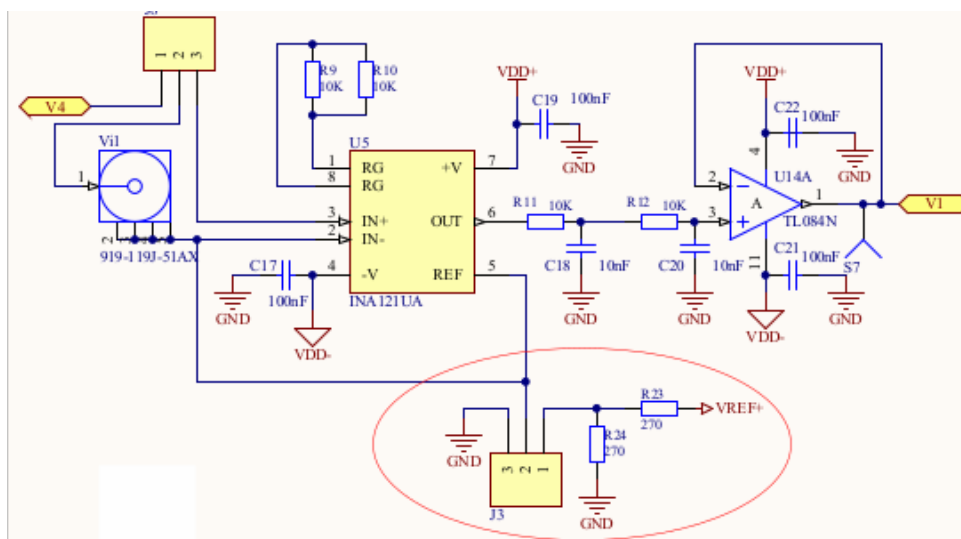


FIGURA 2.8 – Opção de “offset” inserida na entrada do amplificador INA121.

Como pode ser visto na Figura 2.9, a placa de aquisição possui um microcontrolador PIC18F14K50 secundário, responsável por estabelecer a conversão entre os padrões USB e RS-232, disponibilizando uma comunicação eficiente e atual com um computador através de uma porta padrão USB 2.0. A gravação do firmware deste microcontrolador pode ser feito no próprio circuito, sem a necessidade de remoção do componente, através do conector ICSP⁷, utilizando o gravador Pickit3 da Microchip (SMOLNIKAR; MOHORCIC, 2008).

⁷ ICSP - In-Circuit Serial Programming

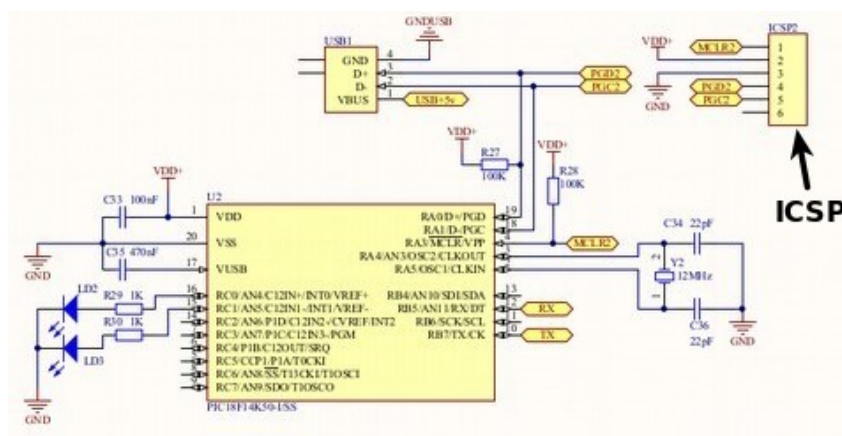


FIGURA 2.9 – Microcontrolador PIC18F14K50, responsável pela conversão RS-232 para USB.

Outra modificação sugerida neste segundo protótipo foi a inclusão de pinos de acesso à outras portas do microcontrolador bem como a reorganização de todas as portas de entrada e saída, dispostas nas bordas da placa. O resultado pode ser visto na Figura 2.10. Os esquemáticos completos deste segundo protótipo encontram-se no Apêndice C.

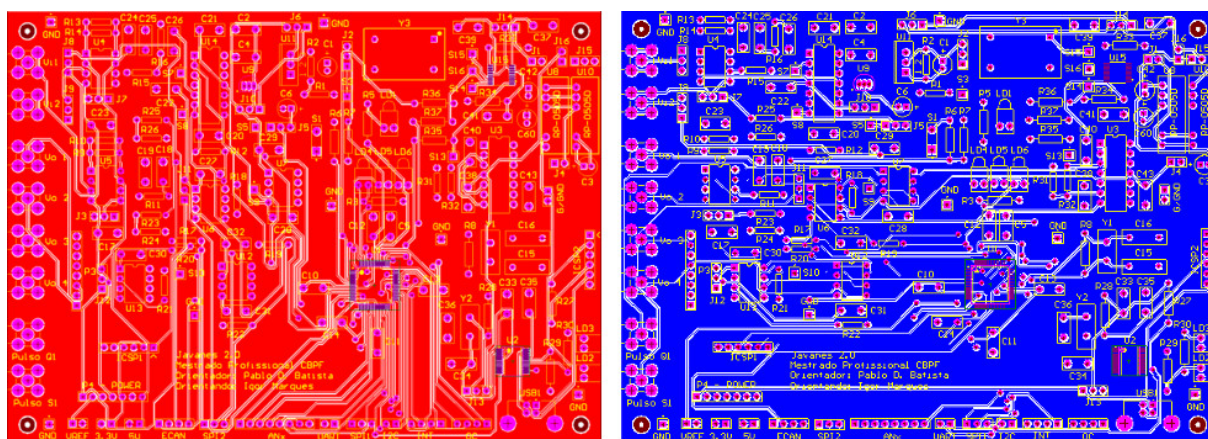


FIGURA 2.10 – Layout superior a) e inferior b) da placa desenhada utilizando o Altium.

Tanto no primeiro protótipo, quanto no segundo, optamos pela utilização do encapsulamento DIP⁸ para todos os componentes, exceto o DDS e o PIC18F14K50. Essa opção tem como objetivo facilitar a montagem da placa sem a necessidade de ferramentas e técnicas dedicadas a outros tipos de encapsulamento, como o SMD. Por outro lado, nada impede que estes componentes possam ser convertidos para o encapsulamento SMD no futuro, visando a miniaturização do circuito.

A Figura 2.11 mostra uma visualização em 3D de como será a placa após sua fabricação. Este recurso está disponível na grande maioria das ferramentas de CAD-CAE para projetos de placa de circuito impresso.

⁸DIP - Dual In-Line Package.

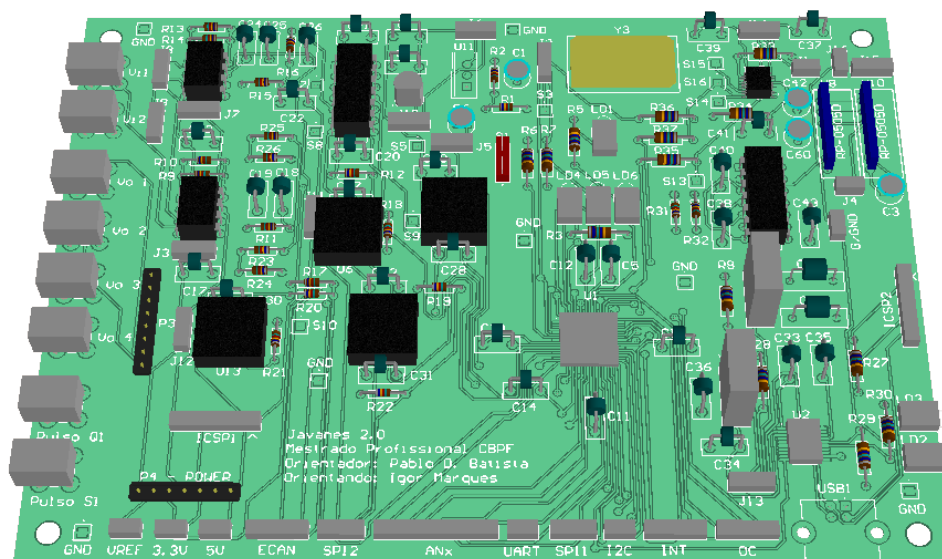


FIGURA 2.11 – Visualização 3D da placa.

Finalmente, após concluir todas as alterações necessárias no primeiro protótipo, o projeto também foi enviado para fabricação profissional. Posteriormente, a montagem dos componentes foi feita no CBPF. A Figura 2.12 apresenta uma foto da placa final.

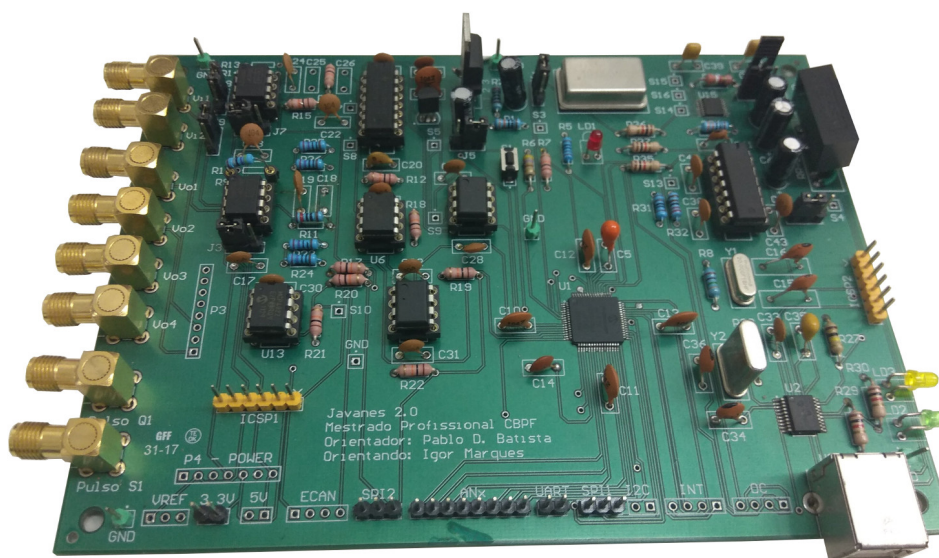


FIGURA 2.12 – Foto do segundo protótipo.

A lista dos principais componentes necessários para a montagem de uma placa, já com os custos de importação, pode ser vista na Tabela 2.1. Na Tabela 2.2 estão os custos relativos a fabricação da placa e soldagem dos componentes SMD.

TABELA 2.1 – Tabela de custos da aquisição dos componentes do segundo protótipo.

Item	Descrição	Qt.	Valor Unitário (R\$)	Valor total (R\$)
1	CI INA121P	2	22,82	45,64
2	CI L272M	2	1,58	3,16
3	Conector SMA Crimpar	8	12,88	103,04
4	Cristal 12MHz	1	3,45	3,45
5	DSPIC33FJ128GP706A	1	18,84	18,84
6	MCP1525	1	2,39	2,39
7	LM317	1	2,32	2,32
8	PIC18F14K50	1	7,32	7,32
9	TL084	2	1,50	3,00
10	RP-0505D	2	35,29	70,58
11	Cristal 20MHz	1	1,04	1,04
12	MCP4822	2	9,08	18,16
13	AD9835	1	52,00	52,00
				330,94

TABELA 2.2 – Outros custos com a fabricação da placa.

Descrição	Qt.	Valor Unitário (R\$)
Fabricação da placa	1	198,00
Soldagem dos CI SMD	1	117,00
		315,00

2.3 Simulando um Lock-in no computador

Para simular um Lock-in no computador, foi desenvolvido um programa em linguagem C, a fim de investigar seu comportamento matemático e transcrever seu código para um microcontrolador de 16 bits. Os resultados desta simulação são salvos em arquivo de texto, contendo uma sequência de valores do sinal simulado, juntamente com as etapas dos calculados caracterizando as operações do Lock-in, resultando então em um modelo matemático do instrumento. Para simular um sinal de referência no código, foi utilizado uma bloco que se repete “N” vezes onde N é o tamanho da janela de observação. Deste modo, foi criado um vetor denominado `Ref[ i ]`, como mostra a Figura 2.13.

```

1  for(i=0 ; i < N ; i++){
2      t[i] = (double)i / FA;
3      //sinal senoidal
4      Ref_in[i] = sin( 2.0 * PI * Fsinal * t[i] );
5      //sinal quadrado
6      //if(sin( 2.0 * PI * Fsinal * t[i] + 16) >= 0 ) Ref_in[i] = 2 ; else
          Ref_in[i] = -2 ;
7  }
```

FIGURA 2.13 – Código para gerar o sinal referência.

Da mesma forma, foi criado um novo bloco de repetição que se repete “N” vezes para criar um vetor denominado `V_sig[ i ]` com uma função senoidal simulando o sinal de interesse. Em seguida adicionamos outra função contendo parâmetros randômicos, simulando assim um ruído no sinal de interesse `V_sig[ i ]`, como mostra a Figura 2.14.

```

1  for(i=0 ; i < N ; i++){
2      t[i] = (double)i / FA;
3      //V_sig0[i] = A * sin( 2.0 * PI * Fsinal * t[i] + 16);
4      V_sigO[i] = A * sin( 2.0 * PI * Fsinal * t[i] );
5  }
6  // ruído
7  for(i=0 ; i < N ; i++){
8      V_sig[i]=V_sigO[i]+(8*( rand()%3 ) * sin(2.0*PI*Fsinal*(rand()%40 ) * t[i]
          ]+rand()%90));
9  }
```

FIGURA 2.14 – Código para gerar o sinal de interesse.

Para simular um sistema de aquisição independente ignoramos o conhecimento dos parâmetros usados para gerar os sinais anteriores e implementamos um PLL (phase locked

loop) (KUZNETSOV *et al.*, 2011), a fim de encontrar a frequência e a fase do sinal de referência, definindo a variável Fsig, como é mostrado na Figura 2.15. Em seguida, foi possível criar um novo vetor contendo um sinal senoidal em fase e com a mesma frequência do sinal de referência. Deste modo foi possível simular como seria a leitura de sinais externos onde não conhecemos os seus aspectos. Este método para simular um PLL foi implementado de forma simplificada visto que o Lock-in que está sendo implementado terá seu funcionamento por blocos de dados calculados e exibidos.

```
1  // PLL Digital
2  // encontrar o período
3  for(i=1 ; i < N ; i++){
4      if(Ref_in[i] > 0 && Ref_in[i-1] < 0 && start > -1 ){
5          end = i;
6          count++;
7      }
8      if(Ref_in[i] > 0 && Ref_in[i-1] < 0 && start == -1)
9          start = i;
10 }
11 pontos = ( (end - start) / count ) ;
12 Fsig = ( FA / pontos );
13
14 // gerando um seno com a frequência e fase encontrada
15 for(i=0 ; i < N ; i++){
16     V_ref[((int)start+i-1)] = sin( 2.0 * PI * Fsig * t[i] ) ;
17 }
```

FIGURA 2.15 – Código simulando um PLL.

Por fim, o código na Figura 2.16 mostra os cálculos digitais simulando o filtro passa-baixa, as componentes em fase e em quadratura, X e Y, respectivamente, e por fim a magnitude do sinal medido e a fase do Lock-in.

```
1 // Lock-in Digital
2 X = 0 ;
3 Y = 0 ;
4 for(i=start ; i < N ; i++){
5     X += V_sig[i] * V_ref[i]; // Sig * Vref
6     Y += V_sig[i] * V_ref[(int)(i+(pontos/4))]; // Sig * Verf (90)
7 }
8 X = X / (N - start);
9 Y = Y / (N - start);
10 mag = 2 * (sqrt (X*X + Y*Y) );
11 P = atan(Y/X);
```

FIGURA 2.16 – Código simulando um Lock-in Digital.

2.4 Caracterização e Validação

Para caracterizar o ADC com dois canais sincronizados, foi utilizado um gerador de sinal de dois canais, modelo AFG3052C da Tektronix. Ambos os canais gerando um sinal senoidal em fase, com frequência de 50 kHz e amplitude de 1 Vpp. Em seguida, foi mantido a frequência e a fase dos sinais, e variado apenas a amplitude do segundo canal. Este aparato experimental pode ser visto na Figura 2.17.

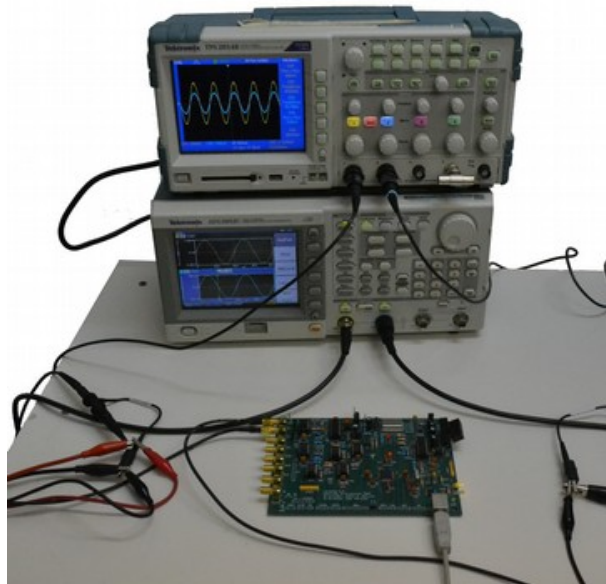


FIGURA 2.17 – Sistema de medidas montado para caracterização do ADC.

Para testar a aplicação em um experimento optoeletrônico, apresentamos o sistema experimental, como mostra a Figura 2.18. A parte óptica é composta por um laser, um fotodetector, dois polarizadores lineares e uma lente convergente. O detector é um fotodi-

odo de Silício (SM05PD2A) capaz de medir sinais contínuos ou pulsados de até (1GHz). Além disso, uma lente biconvexa de distância focal 20 mm (LB1212) está acoplada ao sistema de medida, permitindo concentrar o feixe de luz no centro do detector. Finalmente, são utilizados dois polarizadores ópticos lineares compostos por: uma lente (LPVISE100-A) de diâmetro 25,4mm e um analisador (LPVISE200-A) de diâmetro 50,8mm, ambos fabricados pela Thorlabs. Este analisador conta com um micrômetro acoplado ao sistema giratório permitindo excursionar 14° com uma resolução de $0,028^\circ$. Para pulsar o feixe laser em 5kHz, foi utilizado um gerador de sinal modelo AFG3052C da Tektronix, fornecendo um sinal quadrado com tensão máxima de 1.8V. O gerador de sinal AFG3052C também é responsável por fornecer a tensão de referência para o Lock-in.

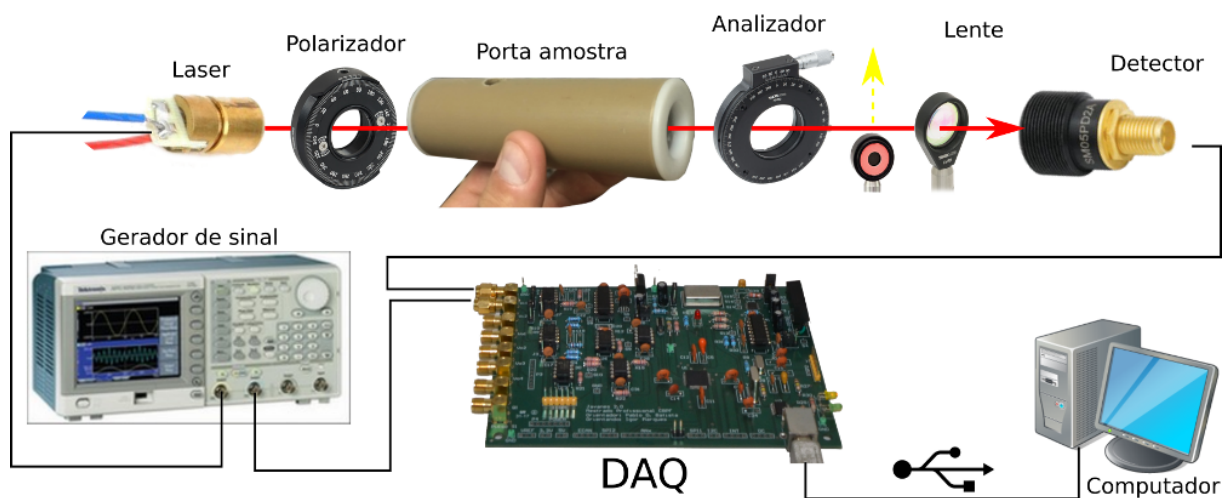
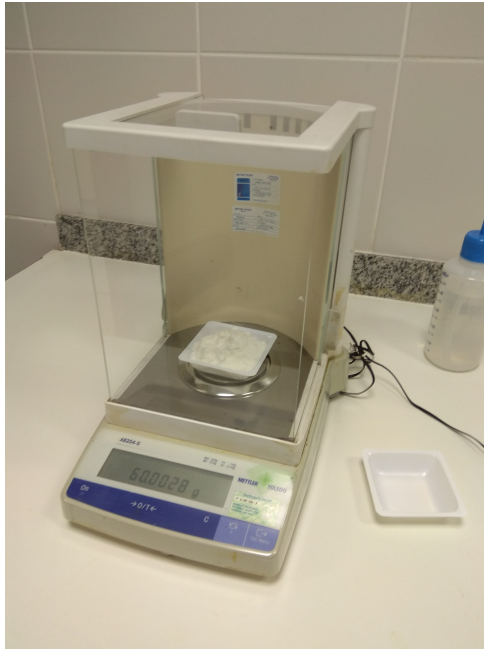
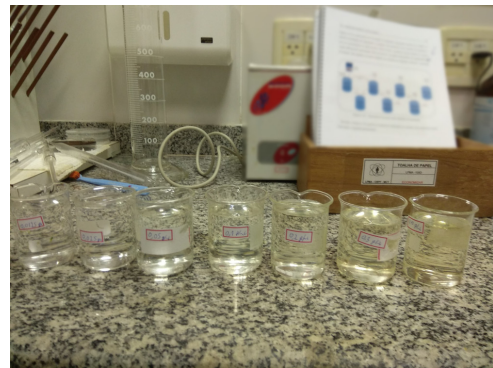


FIGURA 2.18 – Diagrama proposto para o experimento utilizando a placa final do Javanês desenvolvida.

O preparo das amostras e porta amostra foi igualmente reproduzido baseado no trabalho de (FAUTH, 2017). A Figura 2.19 mostra a preparação das amostras no Laboratório de Produção de Materiais Avançados (LPMA) do CBPF. As amostras líquidas utilizadas são soluções de água com açúcar em diferentes concentrações: 400; 300; 200; 100; 50; 25 e 12,5mg/mL. A síntese deste processo, bem como o preparo do porta amostra, estão disponíveis no Apêndice D.



(a)



(b)

FIGURA 2.19 – Preparo das amostras no Laboratório de Produção de Materiais Avançados (LPMA) do CBPF.

Seguindo os trabalhos anteriores do grupo, em medidas de polarização, foi reduzido ao máximo de detecção a potência óptica da luz do laser, e aplicado um ganho de 1000 vezes no amplificador operacional na entrada da placa. Com isto, foi possível definir a máxima potência óptica em $200nW$ medindo $350\mu V$ do ADC.

A Figura 2.20 mostra o aparato experimental montando na bancada

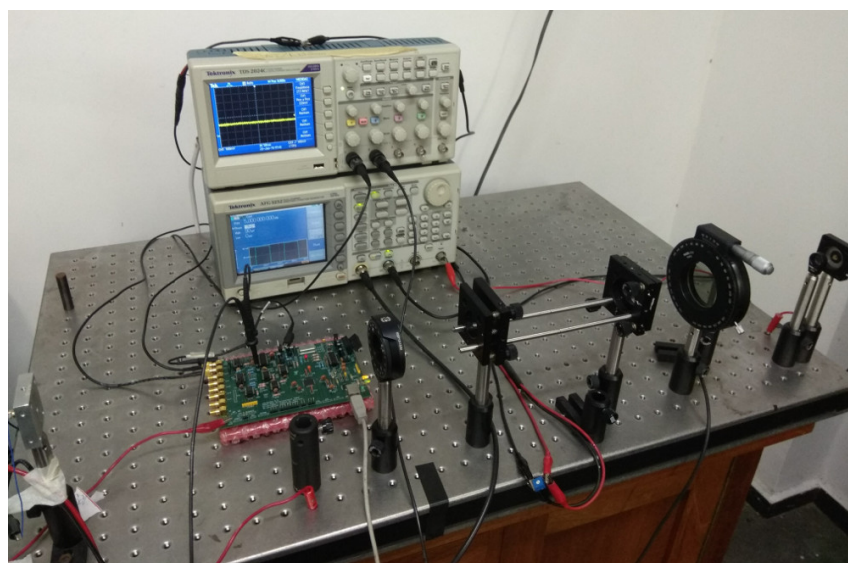


FIGURA 2.20 – Sistema de medidas experimental de medida de glicose.

2.5 Desenvolvimento do código do Lock-in

O objetivo desta seção é mostrar a programação do microcontrolador e as alterações necessárias no programa escrito na linguagem C para ser executado no micro-controlador. Para isto, foi configurado o ADC, o Clock do micro-controlador, o acesso a DMA e o Timer.

Iniciando pela configuração do Clock do micro-controlador, é necessário definir algumas instruções para informar ao compilador que será utilizado o Clock externo, ligado no nos pinos OSC1 e OSC2, como pode ser visto na figura 2.21.

```

1 //1- Seleciona o oscilador
2 _FOSCSEL(FNOSC_PRIPLL & IESO_OFF);
3 //2- Configura a troca para o oscilador
4 _FOSC( POSCMD_HS & OSCIOFNC_OFF & FCKSMCSECMD );
5 //Watchdog Timer Off
6 _FWDTP(FWDTPEN_OFF); frame

```

FIGURA 2.21 – Configuração do Clock do microcontrolador.

Para que o microcontrolador tenha uma velocidade de 40MHz, é preciso definir as configurações do PLL. Além disso uma pausa é inserida no código para garantir que o programa inicie apenas quando o clock externo for chamado. Como mostra a Figura 2.22.

```

1 // espera a troca para o clock principal 0b011
2 while (OSCCONbits.COSC != 0b011);
3
4 // Espera o PLL ser ativado
5 while (OSCCONbits.LOCK != 1) {}; frame

```

FIGURA 2.22 – Pausa para aguardar a mudança de Clock.

Para uma correta conversão do analógico para o digital, tanto o tempo de amostragem quanto o tempo de conversão, devem ser definidos para garantir um número mínimo de ciclos do clock. O tempo de conversão para o modo de 10-bit pode ser visto na Equação 2.2.

$$\begin{aligned}
 T_{AD} &= 76 \text{ ns} \quad (\text{mínimo de ciclos do clock}) \\
 T_{SAMP} &= 2 * T_{AD} \\
 T_{CONV} &= 12 * T_{AD}
 \end{aligned}
 \tag{2.2}$$

Onde T_{AD} é o tempo total do ADC, T_{SAMP} é o tempo de amostragem em cada canal e T_{CONV} é o tempo total de conversão do ADC.

O microcontrolador dsPIC33FJ128GP706A suporta a aquisição simultânea de dois ou quatro canais. Em seguida inicia-se o processo de conversão de cada canal de forma sequencial. Como as informações das entradas analógicas foram capturadas no mesmo instante, isso garante a simultaneidade entre os canais independente do tempo que cada canal levará para completar o ciclo de conversão.

Neste caso, o tempo total, de amostragem e conversão, é multiplicado pelo número de canais lidos “M”, como mostra a Equação 2.3.

$$T_{SIM} = M \cdot T_{CONV} \quad (2.3)$$

Onde: T_{SIM} é tempo total de amostragem e conversão, multiplicado pelo número de canais lidos simultaneamente.

Definido o tempo mínimo necessário para a conversão total, o próximo passo é configurar o registrador “Timer3” do microcontrolador.

O ADC do microcontrolador precisa de um Clock para definir a frequência de amostragem da aquisição. As opções de clock deste microcontrolador são: Contador interno, Timer3 ou Timer5 e Controle por PWM. Como mostra a Figura 2.23, o ADC foi configurado para utilizar o Timer3 e definir o tempo entre as amostragens.

```

1  void initTmr3(void)
2  {
3      IPC2bits.T3IP = 0x01;      // Set Timer3 Interrupt Priority Level
4      IFS0bits.T3IF = 0;        // Clear Timer3 Interrupt Flag
5      IEC0bits.T3IE = 0;        // 1 = Enable Timer3 interrupt
6
7      PR3 = (int) ( FCY/ (FA) );
8
9      TMR3HLD = 0x0000;
10     TMR3     = 0x0000;
11     TMR2     = 0x0000;
12
13     T2CONbits.TCS = 0;          // Internal clock (FOSC /2)
14     T2CONbits.TCKPS = 0b00;
15     T3CONbits.TON = 1;
16 }frame

```

FIGURA 2.23 – Configuração do Timer do microcontrolador.

Basicamente os temporizadores do microcontrolador funcionam como contadores e comparadores de ciclos do clock. Em cada ciclo do clock, um registrador é incrementado. Em seguida comparado com o valor de outro registrado inicialmente definido pelo usuário.

Quando os valores desses dois registradores são iguais, o “Timer” dispara uma trigger para o ADC. Em seguida zera o valor do registrador e começa o processo novamente.

A configuração da DMA se divide em duas partes. A primeira consiste na definição de como a DMA irá se comportar, como mostra a Figura 2.24. A segunda consiste na programação da interrupção gerada pela DMA. Em outras palavras, o que de fato será executado após a DMA ser preenchida.

```

1  void initDma0(void)
2  {
3      DMA0CONbits.AMODE = 0; // = 0 Acesso indireto a DMA with Post-
        Increment mode
4      DMA0CONbits.MODE = 2; // = 2 Continuous, Modo Ping-Pong
5      DMA0CONbits.HALF = 0; //
6      DMA0CONbits.SIZE = 0; // word
7      DMA0CONbits.DIR = 0; // = 0 Read from peripheral address, write to
        DMA RAM address
8
9      DMA0PAD=(volatile unsigned int)&ADC1BUF0;
10     DMA0CNT = (DMA_BUFF_SIZE - 1); // (SAMP_BUFF_SIZE*NUMCHS2SCAN
        )-1;
11     DMA0REQ = 13; // Seleciona ADC1 como fonte para DMA
12
13     DMA0STA = __builtin_dmaoffset(&BufferA1);
14     DMA0STB = __builtin_dmaoffset(&BufferB1);
15
16     IFS0bits.DMA0IF = 0; // Limpa flag
17     IEC0bits.DMA0IE = 1; // Habilita interrupt de DMA m
18     DMA0CONbits.CHEN= 1; // Habilita DMA
19
20     IFS0bits.DMA0IF = 0;
21 }frame

```

FIGURA 2.24 – Configuração da DMA.

Como mostra a Figura 2.25, o controlador DMA transfere dados entre os registradores dos periféricos e o espaço de memória SRAM. No entanto, esta memória tem o formato duplo ou (DPSRAM), possibilitando funcionar no modo Ping-Pong. Desta forma, permite que a DMA funcione através de seus próprios endereços, sem impactar na operação da CPU. Esta arquitetura elimina a necessidade de roubar um ciclo da CPU, quando uma transferência de maior prioridade, é solicitada.

```

1  void __attribute__((interrupt , no_auto_psv)) _DMA0Interrupt(void)
2  {
3      static int DmaBuffer = 0;
4      int i;
5
6      if(DmaBuffer == 0)
7      {
8          if (FLAG_CPU_READ == 0 )
9          {
10             FLAG_DMA_WRITE = 1;
11             for ( i = 0; i < (DMA_BUFF_SIZE/NUMCHS2SCAN) ; i++)
12             {
13                 buffer_adc1[i1] = BufferA1[i*NUMCHS2SCAN];
14                 buffer_adc2[i1] = BufferA1[i*NUMCHS2SCAN+1];
15                 i1++;
16                 if( i1 >= SAMPLES)
17                 {
18                     i1 = 0;
19                     IEC0bits.DMA0IE = 0;
20                 }
21             }
22         }
23     }
24     else
25     {
26         if (FLAG_CPU_READ == 0 )
27         {
28             FLAG_DMA_WRITE = 1;
29             for ( i =0; i < (DMA_BUFF_SIZE/NUMCHS2SCAN) ; i ++ )
30             {
31                 buffer_adc1[i1] = BufferB1[i*NUMCHS2SCAN];
32                 buffer_adc2[i1] = BufferB1[i*NUMCHS2SCAN+1];
33                 i1++;
34                 if( i1 >= SAMPLES)
35                 {
36                     i1 = 0;
37                     IEC0bits.DMA0IE = 0;
38                 }
39             }
40         }
41     }
42     DmaBuffer ^= 1;
43     IFS0bits.DMA0IF = 0; //Limpa flag de interrupt
44     FLAG_DMA_WRITE = 0;
45 }frame

```

FIGURA 2.25 – Configuração da interrupção da DMA.

A implementação do código do Lock-in no microcontrolador, bem como o código completo do firmware escrito na linguagem C encontram-se no Apêndice B.

3 Resultados

Nesta seção serão apresentados os resultados obtidos no processo de caracterização dos dois canais de ADC sincronizados, utilizando um gerador de sinal diretamente conectado ao hardware desenvolvido, abordando os aspectos de frequência e amplitude dos sinais gerados. Neste sentido será discutido principalmente o sincronismo dos dois canais de ADC.

Por se tratar de um projeto de instrumentação científica, este trabalho possui como principal resultado o próprio dispositivo desenvolvido com a evolução em dois protótipos.

3.1 Caracterização

A Figura 3.1 apresenta a tela do software de Lock-in desenvolvido pelo grupo LEPS, sendo executado no computador. Através dos dados obtidos em tempo real pela porta USB, foi possível obter os resultados da detecção da amplitude e fase do sinal, gerado pelo gerador de sinais. Na parte inferior da tela é mostrado o valor da amplitude do sinal desejado, identificado pela letra “R”. A fase do sinal em relação a referência, identificado pela letra “P”, e a frequência do sinal pela letra “F”. Esse experimento mostra que o sistema está operando de maneira satisfatória como uma placa de aquisição de dados, com dois canais sincronizados.

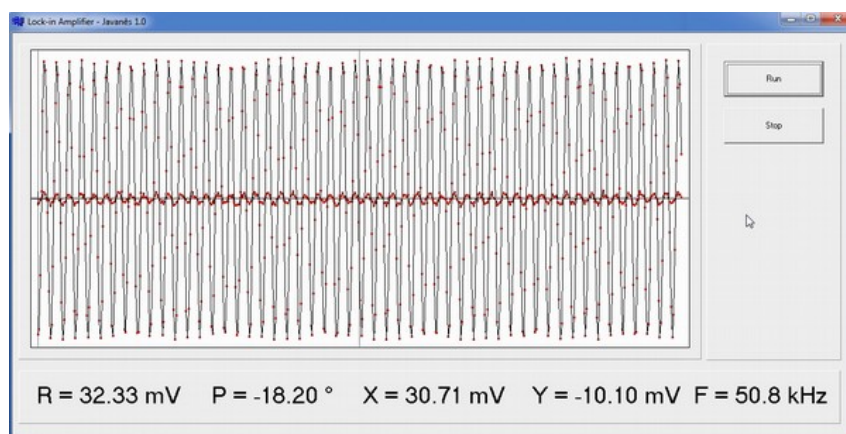


FIGURA 3.1 – Resultado das medições do Lock-in no computador.

É possível notar que a frequência de 50 kHz é o limite para uma aquisição coerente, pois a medida em que a frequência aumenta a definição diminui por causa do erro de digitalização. Para estudar este comportamento, foram feitas medidas da resposta em frequência, Nos quais foi mantida a mesma amplitude e a fase nos dois canais, variando apenas a frequência, como mostra a Figura 3.2 .

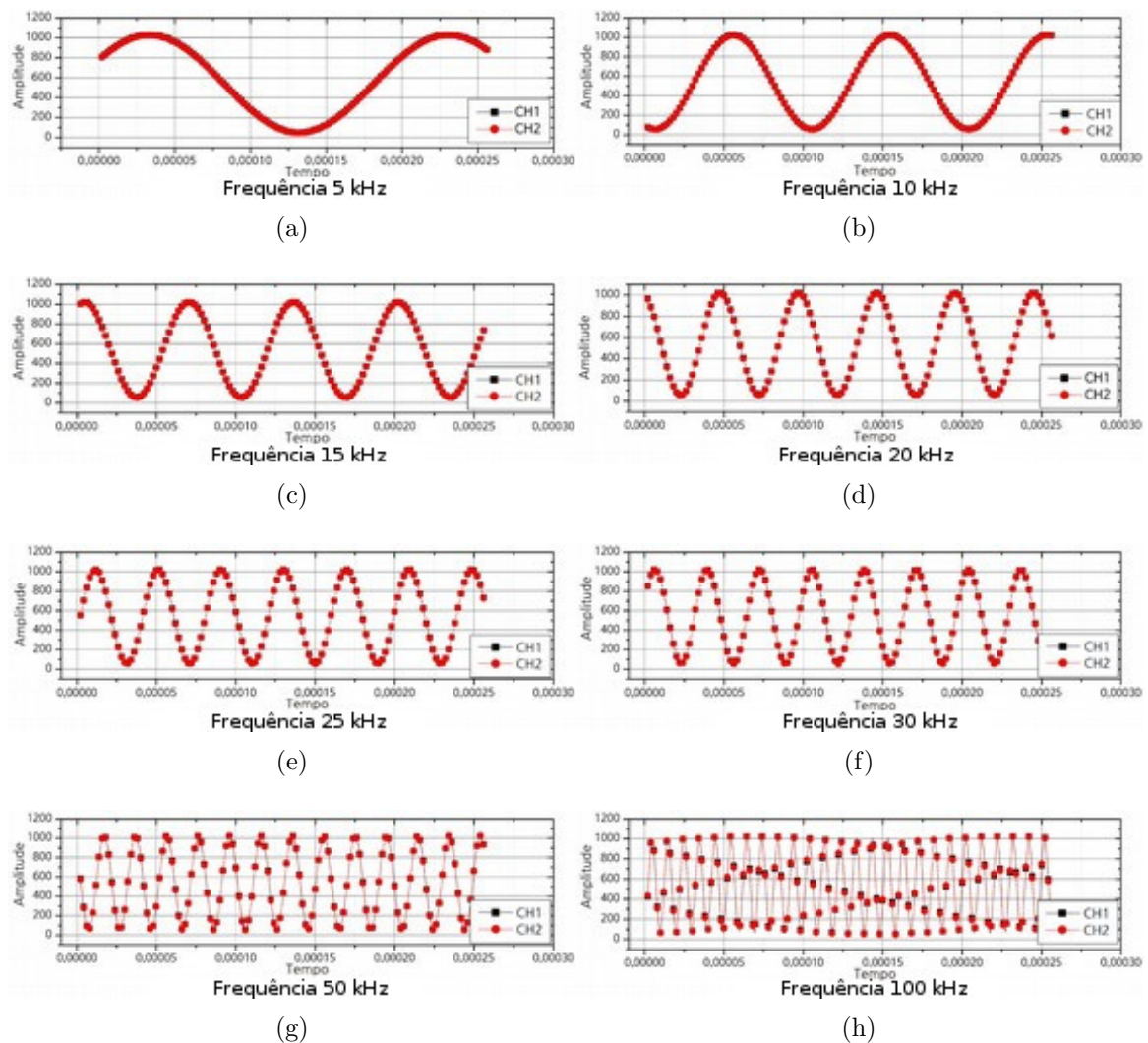


FIGURA 3.2 – Caracterização do ADC variando a frequência.

Como pode ser visto na Figura 3.2, a medida em que a frequência do sinal aumenta, a quantidade de pontos medidos por período diminui. Aumentando assim o erro de amostragem.

Em outro experimento no qual a amplitude é variada até o seu limite de detecção, pode ser notado também o erro de quantização como mostra a Figura 3.3.

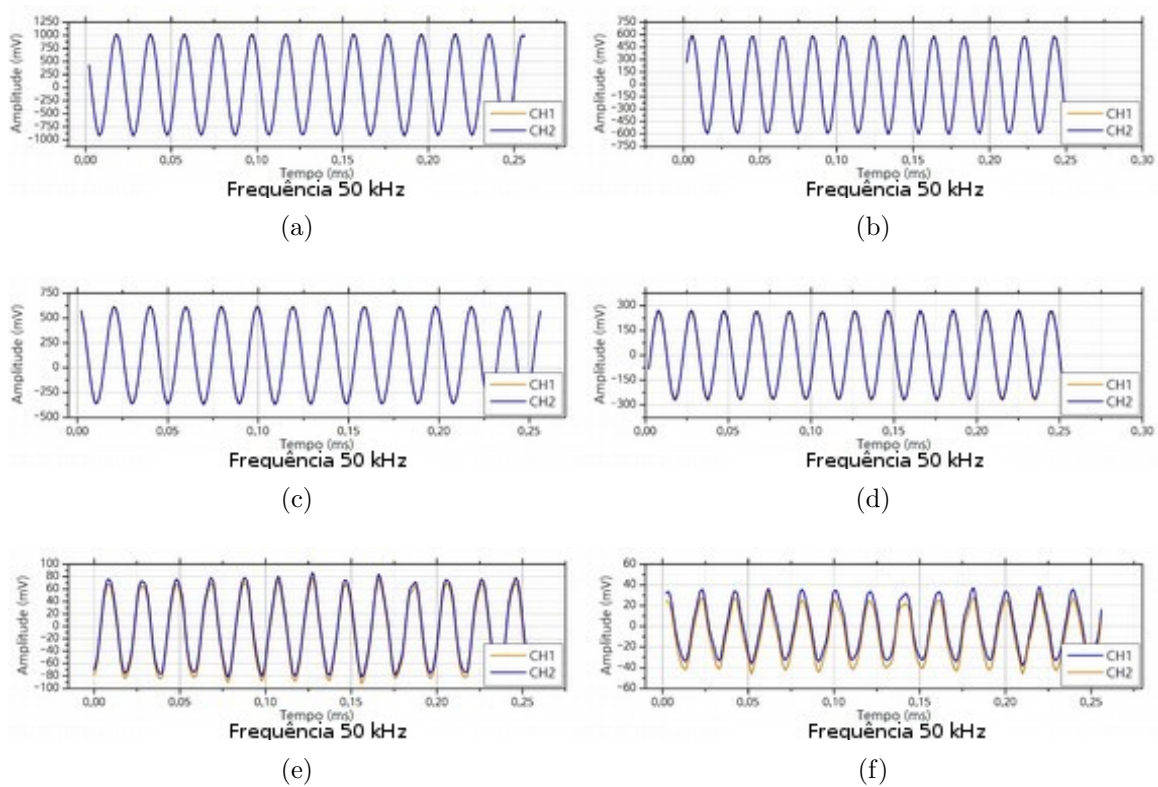


FIGURA 3.3 – Caracterização do ADC variando a Amplitude; a) Sinal com 1,9V , b) 1,2V, c) 1V; d) 600 mV; e) 200 mV; f) 100 mV.

Com o ADC configurado em 10 bits de resolução, tem-se 1024 níveis de quantização. De modo que, para uma tensão de referência de 2,48V, temos 2mV de resolução. Como pode-se notar na Figura 3.3(f), com um sinal com amplitude de 100mV pico a pico tem-se apenas 50 níveis digitais para quantizar este sinal, afetando assim sua representação digital.

3.2 Validação

3.2.1 Simulando um Lock-in no computador

Para simular um sinal ruidoso, foi criado com um bloco que se repete “N” vezes com uma função senoidal. Em seguida foi adicionado uma outra função com parâmetros randômicos. O resultado pode ser visto na Figura 3.4.

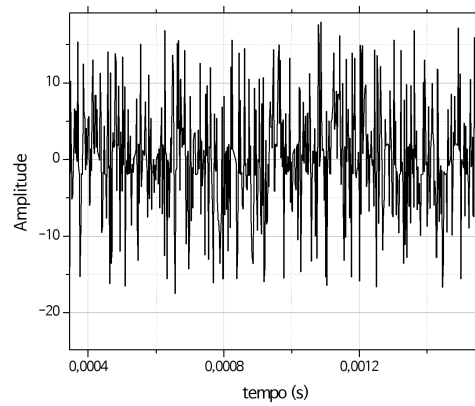


FIGURA 3.4 – Sinal da amostra com ruído simulado.

Após o programa de simulação detectar a frequência e a fase do sinal de referência, uma nova função é chamada para gerar dois novos sinais, um sinal seno de mesma fase do sinal de referência e um outro defasado de 90 graus, como pode ser visto na Figura 3.5.

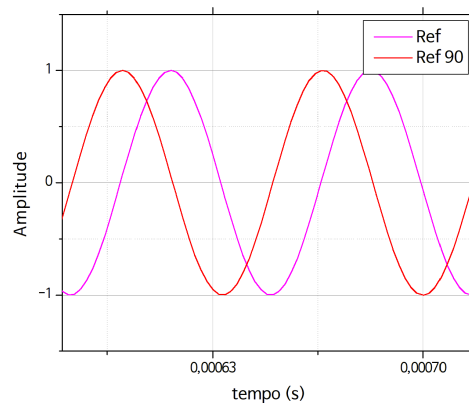


FIGURA 3.5 – Sinais seno e cosseno gerados a partir do PLL.

Com o objetivo de estudar o funcionamento matemático do Lock-in, a Figura 3.6 mostra claramente todas as etapas sobrepostas no mesmo instante. A curva tracejada representa o sinal de interesse sem ruído. As curvas laranja e roxa são as duas funções geradas a partir da referência, uma estando na mesma fase que o sinal principal e a outra defasada de 90 graus. Por fim, o objetivo final deste gráfico é mostrar os resultados das multiplicações dos sinais. A curva azul representa a multiplicação do sinal principal pela referência e a curva verde a multiplicação do sinal principal pela referência em quadratura.

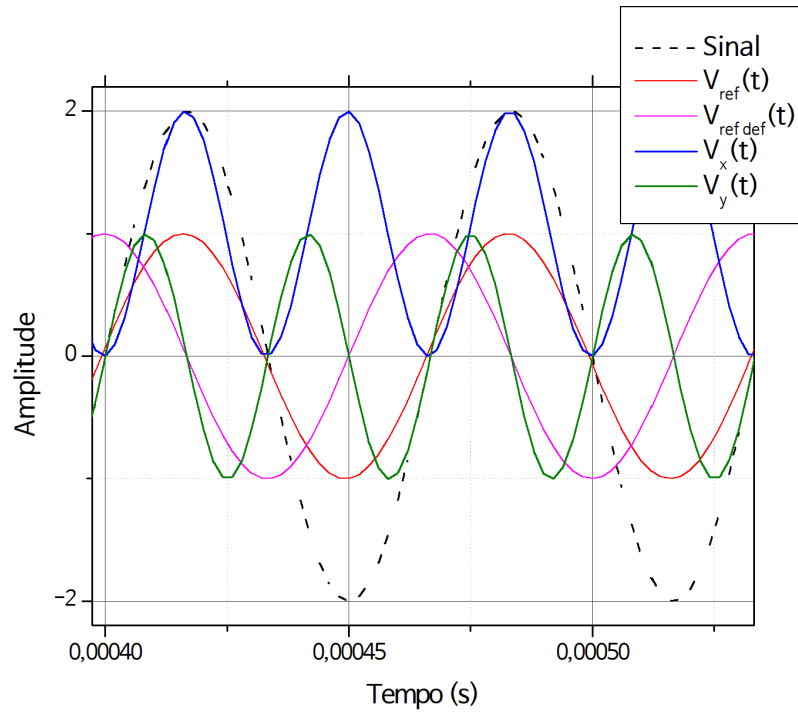


FIGURA 3.6 – Resultado final da simulação com todos os parâmetros do Lock-in.

Observando a curva azul $V_x(t)$, é possível notar que ao aplicarmos um somatório neste sinal, o resultado obtido será o valor RMS da metade da amplitude do sinal de interesse, ou seja, $(2 * \sqrt{X^2 + Y^2})$.

3.2.2 Lock-in implementado no computador

Com o objetivo de testar a sensibilidade do Lock-in e fornecer parâmetros comparativos, quando executado no computador, foi gerado um sinal com amplitude fixa variando apenas a frequência do sinal de referência. Com isso obtêm-se a resposta em frequência medida pela Lock-in.

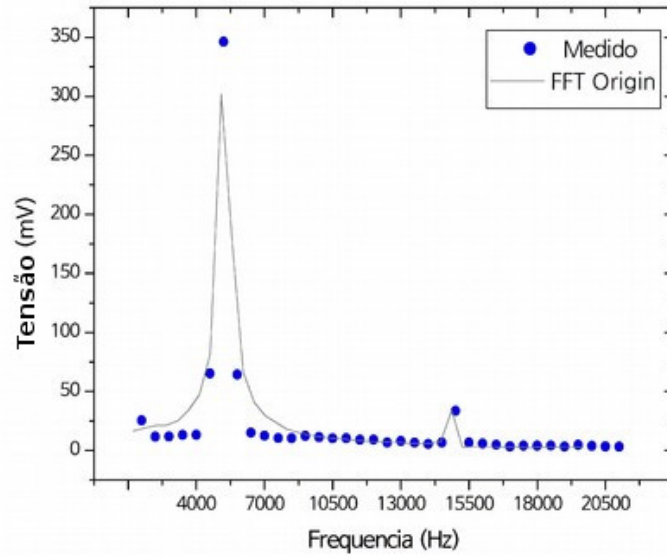


FIGURA 3.7 – Distribuição espectral do sinal medido, comparada com a análise de Fourier feita pelo Origin.

A partir do gráfico na Figura 3.7, pode ser observado uma relação coerente entre a medida feita pela implementação do Lock-in, representado pelos pontos, com o cálculo matemático da transformada de Fourier do sinal principal, obtido através de uma função matemática no programa "Origin Lab".

Dando prosseguimento ao trabalho, é apresentado na Figura 3.8 a análise da resposta em frequência dada em dB. Os dados são os mesmos ao anterior, apenas os valores foram convertidos para dB utilizando a Equação 3.1.

$$dBV = 20 * \log \left(\frac{V_0}{V} \right) \quad (3.1)$$

Em que V_0 é a tensão de referência.

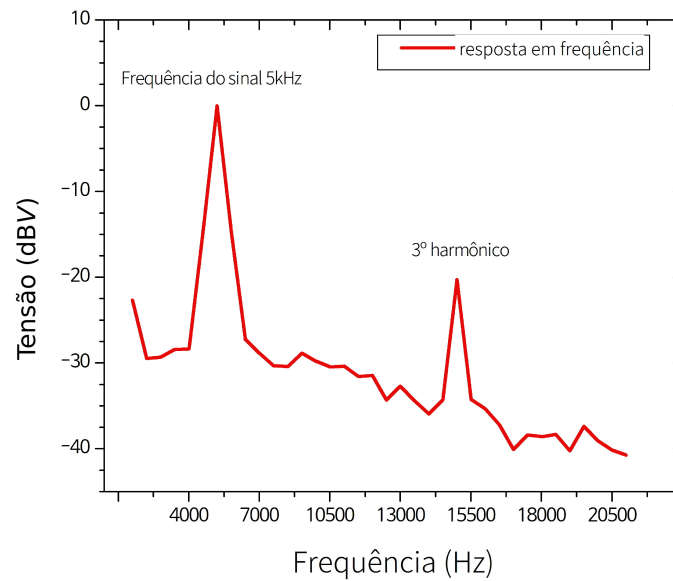


FIGURA 3.8 – Análise da resposta em frequência em dB.

Observando a Figura 3.8 é possível notar que a faixa de ruído das outras frequências é 30 vezes menor que o sinal principal tendo apenas no terceiro harmônico, um pico 10 vezes menor que sinal principal. Após o terceiro harmônico, o ruído cai para 100 vezes menor que o sinal principal.

A Figura 3.9 (a) apresenta a resposta dos fotodetectores em função da potência óptica e em (b) uma representação da estabilidade ao longo do tempo. O procedimento consiste em variar a potência óptica do laser, em intervalos de $25\mu\text{W}$. Para isto foi utilizado um detector auxiliar do modelo (S120C) da ThorLab para monitorar a grandeza. Em seguida recolhe-se o detector auxiliar a fim de obter a medida pelo detector (SM05PD2A) ligado diretamente a entrada do Javanês.

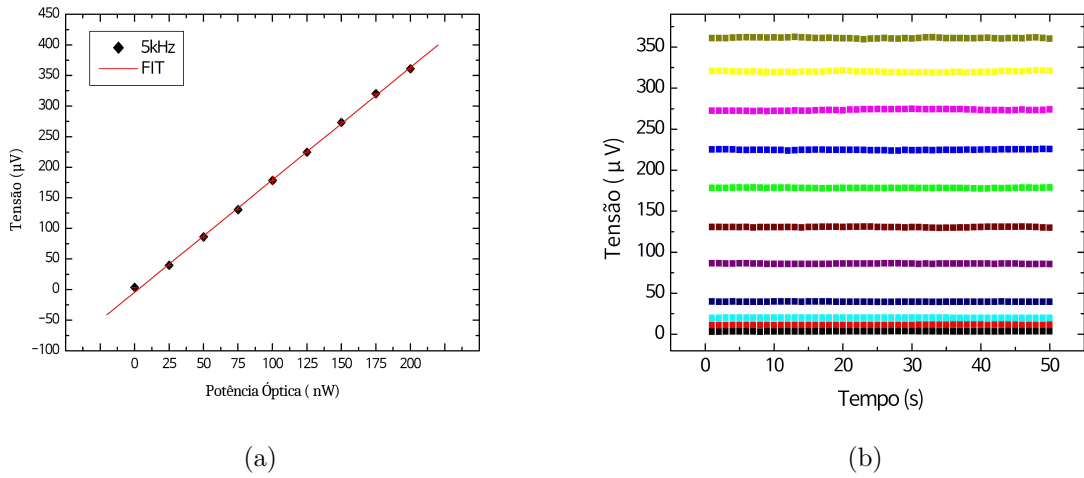


FIGURA 3.9 – Em (a) temos a relação da potência óptica aplicada e a tensão de saída do sistema; Em (b) estabilidade ao longo do tempo.

TABELA 3.1 – Estabilidade da Potência Óptica.

Potência μW	Tensão μV	Erro
0	3,39167	0,16343
25	39,54379	0,01631
50	85,96761	0,02462
75	130,50357	0,04596
100	178,28444	0,04462
125	224,74533	0,06743
150	273,32169	0,11719
175	320,08288	0,0951
200	361,09036	0,08916

Como esperado, observa-se um comportamento linear em quase toda a região de potência óptica investigada, havendo uma pequena perda da linearidade abaixo de $25 \mu W$. No entanto, permanecendo estável com um erro médio de 0,047 observado na Tabela 3.1.

Tendo em vista o objetivo de validar este trabalho, foi utilizado um aparato de medida optoeletrônica para verificar a Lei de Malus (KAHR; CLABORN, 2008). A Figura 3.10 mostra a resposta do detector óptico em função do ângulo do analisador. O procedimento foi feito variando o ângulo de 0° a 360° em intervalos de 10 graus.

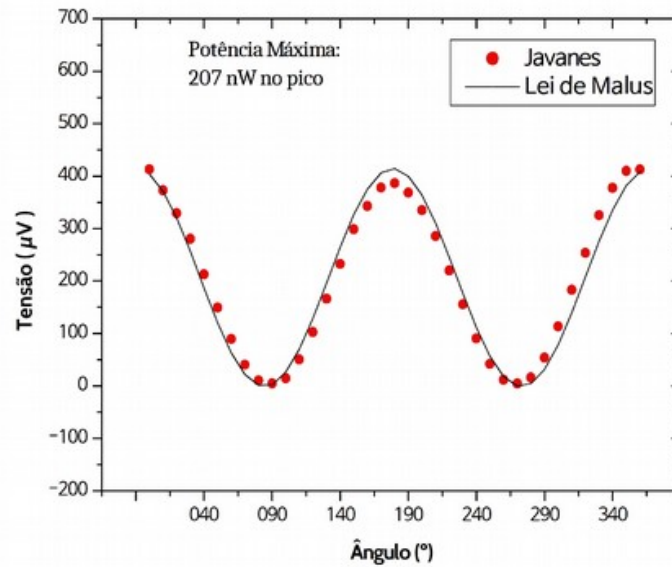


FIGURA 3.10 – Análise da variação da potência óptica com a Lei de Malus.

O resultado deste experimento mostra coerência entre as medidas, representadas pelos pontos vermelhos, e a Lei de Malus, definida por uma função cossenoidal quadrática, representada na figura como a curva preta.

3.2.3 Lock-in embarcado no microcontrolador

A Figura 3.11 apresenta a interface Web desenvolvida para visualizar os dados recebidos pela porta USB da placa do Javanês em tempo real. É possível obter os valores instantâneos da detecção da amplitude e fase do sinal medido, bem como os valores das componentes X e Y do Lock-in.

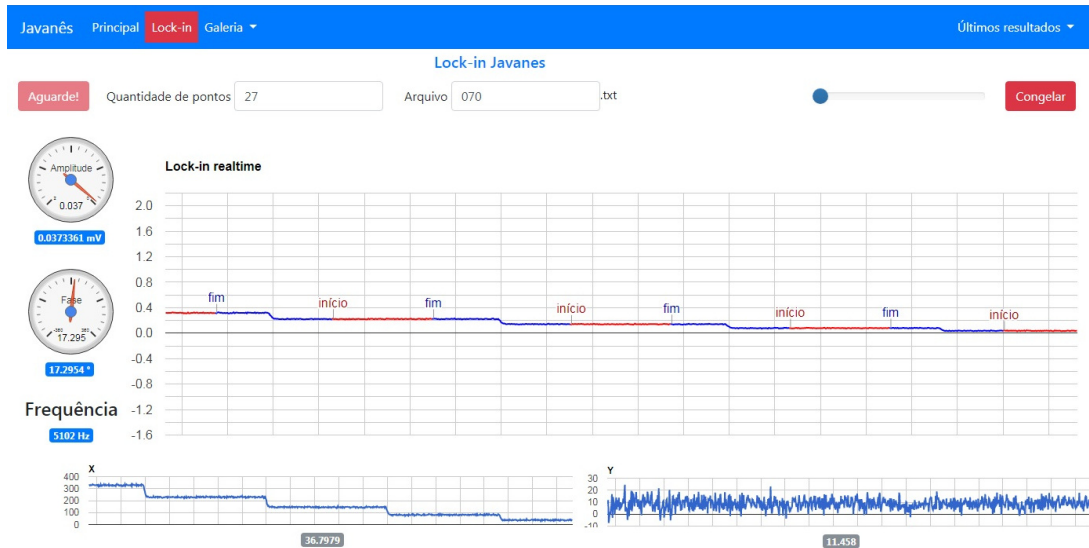


FIGURA 3.11 – Resultado das medições utilizando o Lock-in sendo executado no micro-controlador.

Para comparar o desempenho do Lock-in sendo executado diretamente no micro-controlador com os resultados obtidos quando foi executado no computador, definiu-se os mesmos parâmetros. No entanto, para as frequências próximas a $5kHz$, foram obtidos pontos com intervalos menores, de $100Hz$ em $100Hz$, como pode ser visto na Figura 3.12.

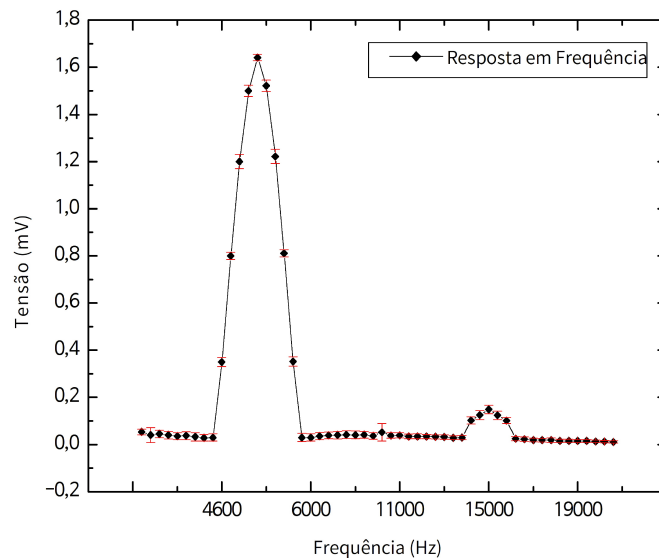


FIGURA 3.12 – Distribuição espectral do sinal medido.

A partir do gráfico na Figura 3.12, podemos observar sutilmente o segundo harmônico, diferentemente do experimento anterior no qual apenas o terceiro harmônico foi percebido. Para uma melhor análise, os resultados deste gráfico foram convertidos para dB, e os dados

apresentados na Figura 3.13.

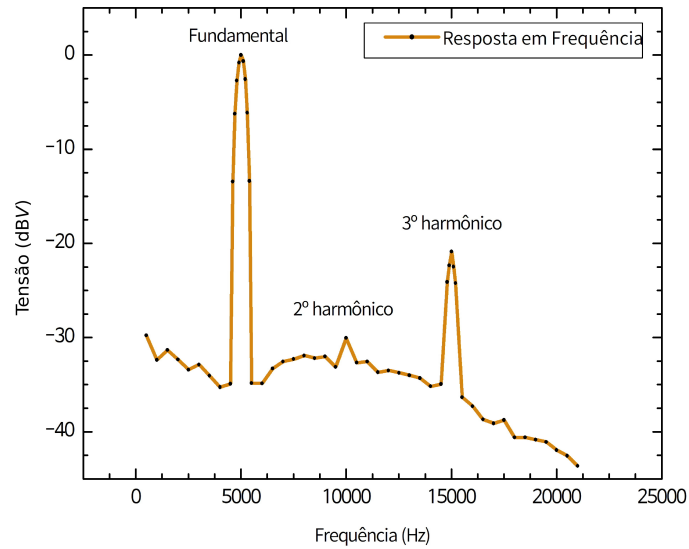


FIGURA 3.13 – Análise da resposta em frequência em dB.

TABELA 3.2 – Resposta em frequência.

Frequência (Hz)	Amplitude (dB)
4500	-34,89602
4600	-13,42282
4700	-6,24238
4800	-2,72056
4900	-0,78236
5000	-0,00000
5100	-0,65755
5200	-2,56524
5300	-6,12211
5400	-13,36014
5500	-34,83013

Observando a Figura 3.13 podemos notar que a reserva dinâmica deste Lock-in implementando no micro-controlador é de 20 dB, o suficiente para aquisição de sinais da ordem de μV , levando em consideração que o ganho aplicado na entrada do ADC é de 1000 vezes.

Repetindo o experimento anterior, a potência óptica do laser foi variada, em intervalos de $25\mu\text{W}$, e comparado com a grandeza obtida pelo detector auxiliar (S120C) da

ThorLab. Em seguida recolhe-se o detector auxiliar a fim de obter a medida pelo detector (SM05PD2A).

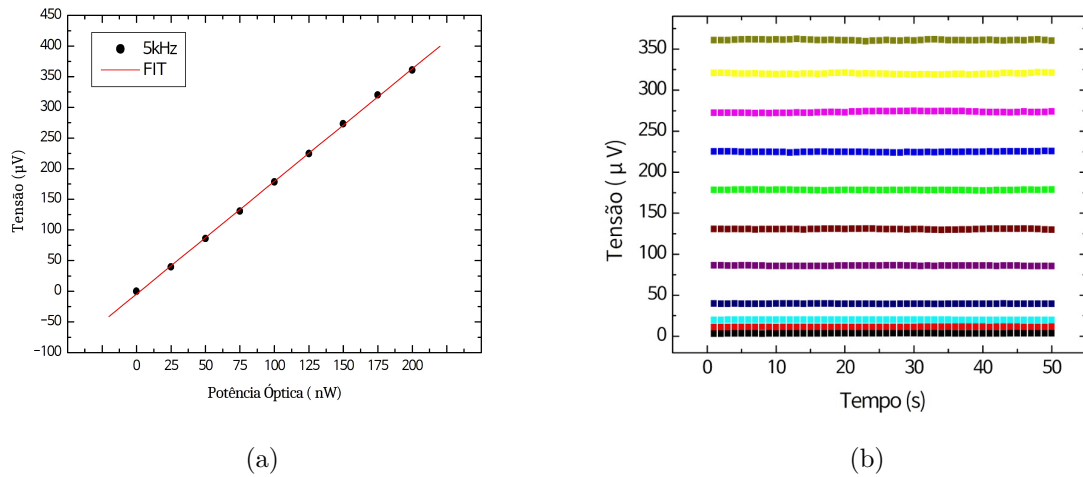


FIGURA 3.14 – Em (a) temos a relação da potência óptica aplicada e a tensão de saída do sistema; Em (b) estabilidade ao longo do tempo.

TABELA 3.3 – Estabilidade da Potência Óptica.

Potência μW	Tensão μV	Desvio (yEr)	Erro
25	39,54379	0,11529	0,01631
50	85,96761	0,1741	0,02462
75	130,50357	0,32499	0,04596
100	178,28444	0,31553	0,04462
125	224,74533	0,47681	0,06743
150	273,32169	0,82867	0,11719
175	320,08288	0,67246	0,0951
200	361,09036	0,63047	0,08916

Como esperado, obtivemos resposta linear e estabilidade em toda faixa de frequência analisada, como mostra a Figura 3.14. Além disso, a potência óptica se manteve estável durante todo o experimento podendo ser observado na Figura 3.14 (b) e melhor detalhando no quadro 3.3. Sendo assim, foi adotada arbitrariamente a frequência de 5 kHz para os todos os próximos experimentos.

Finalmente, para uma validação final utilizando o Lock-in sendo executado diretamente no microcontrolador, o mesmo experimento de medida optoeletrônica foi feito, a fim de verificar a lei de Malus. A Figura 3.15 apresenta a tensão em função do ângulo, na medida em que o ângulo do analisador gira de 0° a 360° , com intervalos de 5° .

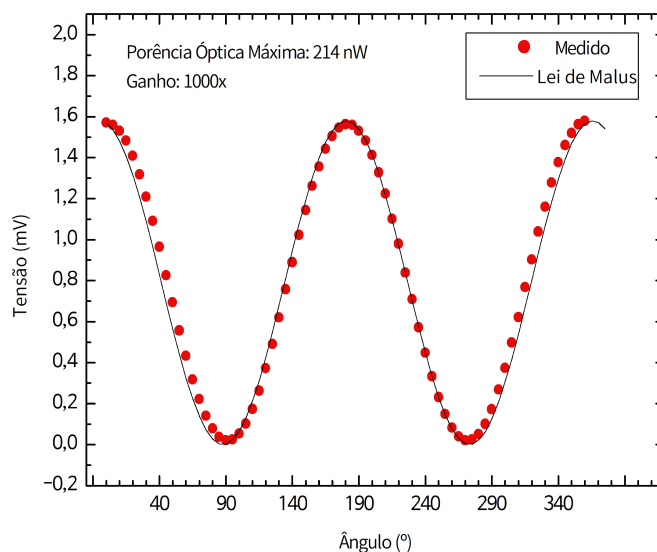


FIGURA 3.15 – Análise da variação da potência óptica com a Lei de Malus

Nota-se uma clara melhora na concordância entre as medidas, representada pelos pontos vermelhos, com a Lei de Malus representada pelo ajuste cossenoidal quadrático.

Para efeito de comparação, podemos dizer que o desempenho destas medidas foram muito semelhantes, tanto com o Lock-in executado no computador, quanto para o Lock-in executado diretamente no microcontrolador.

3.2.4 Aplicação em medida de glicose

Após a caracterização do sistema, prosseguimos para a aplicação em medidas de glicose.

Quando não há rotação do plano de polarização, a luz é bloqueada no segundo polarizador. Já na presença de substâncias opticamente ativas, como o açúcar, o desvio causado no plano de polarização faz com que uma parte da luz atravesse o segundo polarizador e chegue ao detector. O açúcar refinado é composto de mais de 99% de sacarose, uma conhecida, talvez a mais comum, substância opticamente ativa (CARLIN *et al.*, 2005).

A intensidade luminosa que passa pelo segundo polarizador está diretamente relacionada com a concentração de açúcar na amostra.

A Figura 3.16 apresenta o comportamento da potência óptica para sete soluções de água com açúcar.

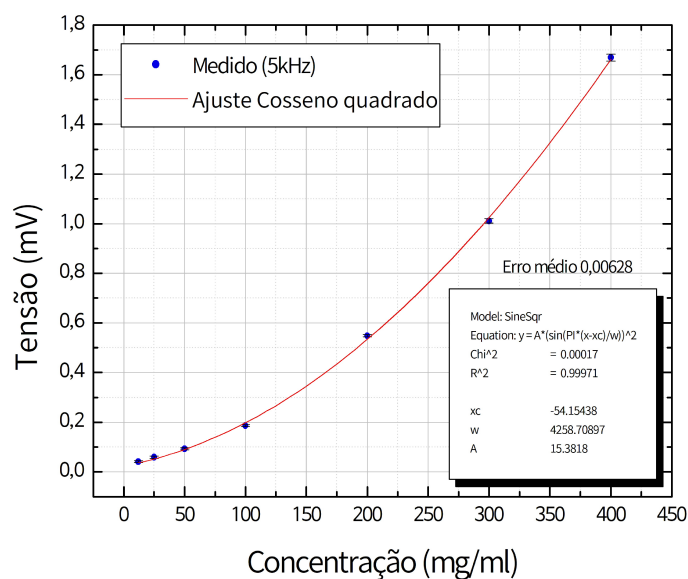


FIGURA 3.16 – Resultados obtidos utilizando o polarímetro.

Analisando o gráfico 3.16, não observamos uma relação linear entre tensão e concentração da solução, mas sim uma relação cossenoidal quadrática por consequência da lei de Malus.

Para descobrir o ângulo de desvio causado pela amostra, podemos girar o segundo polarizador até obter a mínima intensidade luminosa e assim descobrir o ângulo de desvio causado pela amostra. Este ângulo também é proporcional à concentração de açúcar na solução. Desta maneira, podemos obter informação da concentração de açúcar ao determinar o ângulo de desvio, como mostra a Figura 3.17.

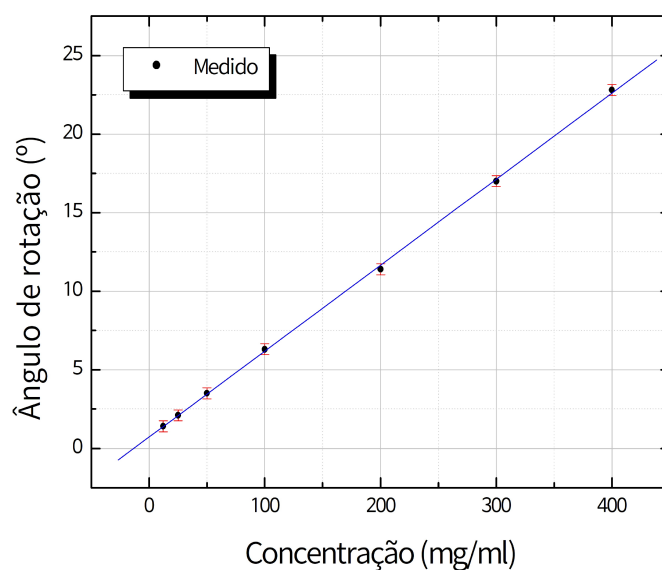


FIGURA 3.17 – ângulo de rotação óptico em função da concentração amostra.

Como esperado, foi observada uma relação linear do ângulo de rotação do plano de polarização em função da concentração da solução. A partir da relação linear do gráfico da Figura 3.17, podemos obter uma sensibilidade de aproximadamente $0,035^\circ$ para cada 1 mg/mL.

A partir da análise dos resultados obtidos pode-se dizer que obtivemos êxito na aplicação proposta, apresentando desempenho e sensibilidade coerentes com a literatura. Sendo assim, o sistema pode funcionar como um polarímetro AC para investigar concentração de substâncias opticamente ativas.

4 Discussão e conclusão

Neste projeto, o objetivo era desenvolver um Amplificador Lock-in digital, baseado no micro-controlador dsPIC33FJ128GP706A, voltado para instrumentação científica. Para isto, começou-se por estudar os principais conceitos básicos de um Amplificador Lock-in. Em seguida, prosseguiu-se para um estudo do estado da arte, o que permitiu perceber os produtos existentes no mercado.

O desenvolvimento deste instrumento foi realizado por partes, tendo como passo inicial a modificação de um placa de aquisição de dados já existente, desenvolvida pelo grupo no Laboratório de Eletrônica e Processamento de Sinais (LEPS). Esta decisão foi crucial no aspecto de economia no tempo de desenvolvimento, visto que, grande parte do circuito elétrico foi aproveitado. Como resultado foram desenvolvidos dois protótipos no decorrer do projeto.

Sendo um poderoso controlador de sinal digital e com suporte DMA, o microcontrolador dsPIC33FJ128GP706A atendeu as expectativas iniciais desse projeto, permitindo uma taxa de amostragem de 1MHz em dois canais ADC simultâneos.

A descoberta do conversor RP-0505D foi de grande importância, permitindo utilizar somente a alimentação de 5V da porta USB e fornecer duas alimentações simétricas de +5V e -5V aos amplificadores. A limitação de corrente de 100 mA deste componente foi superada no desenvolvimento do segundo protótipo, onde foram inseridas chaves para selecionar os componente que seriam alimentados por ele. Além de inserir no circuito da placa, a posição de um segundo RP-0505D em paralelo, aumentando assim a corrente fornecida.

Outras alterações também foram feitas no segundo protótipo, como a implementação de “offset” nos sinais de entrada, permitindo que tensões negativas fossem medidas. Essa mudança foi extremamente necessária para a aplicação do Lock-in, bem como a inclusão de pinos de acesso as outras portas do microcontrolador. Por fim, a reorganização de todas as portas de entrada e saída para facilitar as conexões.

A instalação de outros recursos na placa, tais como DDS, DACs, portas de comunicação, e acesso a diferentes tensões como 3.3v, +5V e -5V, tornou este equipamento mais flexível para outras aplicações na instrumentação científica.

O desenvolvimento de um programa em linguagem “C”, simulando um Lock-in no computador com o objetivo de investigar o comportamento matemático do Lock-in, facilitou também a transcrição do código para o micro-controlador de 16 bits. Poucas alterações foram necessárias.

Nos teste de sensibilidade na implementação do Lock-in foi observado um ruído 30 vezes menor que o sinal principal e uma reserva dinâmica de 20 dB.

Para validar a aplicação do Amplificador Lock-in, foi repetido o experimento optoeletrônico para medidas de rotação do plano de polarização da luz, baseado no trabalho de (FAUTH, 2017), substituindo o amplificador Lock-in comercial MFLI, da Zurich Instruments.

Em seguida, foram repetidos os mesmos experimentos optoeletrônico, onde foi possível definir uma potência óptica máxima de 200 nW medindo 350 μV , sendo aplicado um ganho de 1000 vezes no amplificador operacional. Como esperado, obtivemos respostas lineares e estabilidade em todas faixas de frequência analisadas. Além disso, a potência óptica se manteve estável durante todo o experimento. No teste feito analisando a Lei de Malus, os resultados se mostram coerentes com a função cossenoidal quadrático.

Também nos experimentos optoeletrônicos para medida de concentração de glicose, foi possível observar uma relação linear entre a tensão e concentração da solução. Apresentando baixo erro, e uma resolução angular de $0,035^\circ$ para cada 1mg/mL .

Tendo em conta que se trata de um projeto que não envolve a produção de produto em série, e sim a produção de protótipos experimentais, este projeto contribui com a pesquisa científica brasileira no aspecto de reduzir a importação de tecnologia na área de instrumentação científica (CASTILHO, 2014; MOSCATI *et al.*, 1982). É possível com este instrumento implementar um Amplificador Lock-in embarcado no microcontrolador da placa (CHEN *et al.*, 2017). Para diferentes aplicações de aquisição de dados (CASTILHO, 2014; MOSCATI *et al.*, 1982).

Finalmente, este trabalho foi desenvolvido tendo como base o paradigma do Open-Source hardware, munindo a pesquisa científica com ferramentas flexíveis e de baixo custo, visando expandir sua capacidade de coleta de dados. Portanto, cabe ressaltar que todos os arquivos deste trabalho estão disponíveis para que possam ser utilizados e aprimorados pela comunidade científica em <http://javan.es.carneiro.eng.br>.

5 Perspectivas

Este projeto foi uma excelente prova de conceito. No entanto, necessita de algumas melhorias, como a implementação de uma versão otimizada com CORDIC (VOLDER, 1959; YANG *et al.*, 2009) no código do Lock-in embarcado no microcontrolador de modo a otimizar o tempo de execução dos cálculos de seno e cosseno.

Outras considerações futuras passam pela utilização do DDS interno da placa, para fornecer o sinal quadrado ao laser, melhorando ainda mais o sincronismo da referência interna.

Outra sugestão para aperfeiçoamento deste projeto é a substituição do microcontrolador pelo modelo dsPIC33EP256MU806, o qual tem suporte a comunicação USB, eliminando a necessidade de um segundo microcontrolador na placa.

Referências

ALONSO, R.; VILLUENDAS, F.; BORJA, J.; BARRAGAN, L.; SALINAS, I. Low-cost, digital lock-in module with external reference for coating glass transmission/reflection spectrophotometer. **Measurement Science and Technology**, IOP Publishing, v. 14, n. 5, p. 551, 2003.

ANDRETRY, P.; LINDNER, M. W.; HERRMANN, J. M.; SCHULTZ, A.; KONZOG, M.; KIESEWETTER, F.; HAEUSLER, G. Optical coherence tomography by spectral radar: dynamic range estimation and in-vivo measurements of skin. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Optical and Imaging Techniques for Biomonitoring IV**. [S.l.], 1999. v. 3567, p. 78–88.

ARDUINO. **Arduino project home page**. [S.l.]. Disponível em: <<http://www.arduino.cc>>.

AUGULIS, R.; ZIGMANTAS, D. Two-dimensional electronic spectroscopy with double modulation lock-in detection: enhancement of sensitivity and noise resistance. **Optics express**, Optical Society of America, v. 19, n. 14, p. 13126–13133, 2011.

AZZOLINI, C.; MAGNANINI, A.; TONELLI, M.; CHIORBOLI, G.; MORANDI, C. Integrated lock-in amplifier for contactless interface to magnetically stimulated mechanical resonators. In: IEEE. **Design and Technology of Integrated Systems in Nanoscale Era, 2008. DTIS 2008. 3rd International Conference on**. [S.l.], 2008. p. 1–6.

BLAIR, D.; SYDENHAM, P. Phase sensitive detection as a means to recover signals buried in noise. **Journal of Physics E: Scientific Instruments**, IOP Publishing, v. 8, n. 8, p. 621, 1975.

BOCHNER, M.; ALBUQUERQUE, M. P.; ALFONSO, J. L. G.; RESENDE, L. C.; NUNES, R. A. A. Introdução ao amplificador lock-in e prototipação em hardware reconfigurável. **NOTAS TÉCNICAS**, v. 1, n. 1, 2011.

BURDETT, R. Amplitude modulated signals: The lock-in amplifier. **Handbook of Measuring System Design**, Wiley Online Library, 2005.

CAPITANIO, R. A. **Implementação de um Filtro Sintonizado Tipo Lock-in, Utilizando Processador Digital de Sinais (DSP) na Aquisição de Sinais Fototérmicos para Análise de Perda Fotoquímica em Plantas**. Dissertação (Mestrado), 2008.

- CARLIN, N.; SZANTO, E.; SEALE, W.; JORGE, F.; SOUZA, F.; BECHTOLD, I.; GASQUES, L. Birrefringência em placas de onda e atividade óptica de uma solução de açúcar. **Revista Brasileira de Ensino de Física**, SciELO Brasil, v. 27, n. 3, p. 349–355, 2005.
- CASTILHO, B. V. Desenvolvimento de instrumentação óptica e infravermelha no Brasil (1980-2013). **História da Astronomia no Brasil**. Recife: MAST/MCTI, Cepe Editora e Secretaria de Ciência e Tecnologia de Pernambuco, Recife, v. 2, p. 196–225, 2014.
- CHEN, K.-H.; CHU, Y.-C.; CHEN, J.-H. Applying the phase difference property of polarization angle for measuring the concentration of solutions. **Optics & Laser Technology**, Elsevier, v. 44, n. 1, p. 251–254, 2012.
- CHEN, X.; CHANG, J.; WANG, F.; WANG, Z.; WEI, W.; LIU, Y.; QIN, Z. A portable analog lock-in amplifier for accurate phase measurement and application in high-precision optical oxygen concentration detection. **Photonic Sensors**, Springer, v. 7, n. 1, p. 27–36, 2017.
- CHOU, C.; HUANG, Y.-C.; FENG, C.-M.; CHANG, M. Amplitude sensitive optical heterodyne and phase lock-in technique on small optical rotation angle detection of chiral liquid. **Japanese journal of applied physics**, IOP Publishing, v. 36, n. 1R, p. 356, 1997.
- D'AMICO, A.; MARCELLIS, A. D.; CARLO, C. D.; NATALE, C. D.; FERRI, G.; MARTINELLI, E.; PAOLESSE, R.; STORNELLI, V. Low-voltage low-power integrated analog lock-in amplifier for gas sensor applications. **Sensors and Actuators B: Chemical**, Elsevier, v. 144, n. 2, p. 400–406, 2010.
- FAUTH, L. H. de A. **Desenvolvimento de um sistema optoeletrônico para medida da rotação do plano de polarização da luz**. Dissertação (Mestrado) — Centro Brasileiro de Pesquisas Físicas, Rio de Janeiro, RJ, 2017.
- FERRI, G.; LAURENTIIS, P. D.; D'AMICO, A.; NATALE, C. D. A low-voltage integrated CMOS analog lock-in amplifier prototype for gas applications. **Sensors and Actuators A: Physical**, Elsevier, v. 92, n. 1, p. 263–272, 2001.
- FILHO, L. R. F. F. Módulos eletrônicos para a caracterização elétrica de efgets como sensores de pH. **Centro Brasileiro de Pesquisas Físicas, Rio de Janeiro**, 2012.
- FISHER, D. K.; GOULD, P. J. Open-source hardware is a low-cost alternative for scientific instrumentation and research. **Modern Instrumentation**, Scientific Research Publishing, v. 1, n. 02, p. 8, 2012.
- GREGO, M. O hardware em código aberto. **Info Exame**, Março 2009.
- HU, A.; CHODAVARAPU, V. P. CMOS optoelectronic lock-in amplifier with integrated phototransistor array. **IEEE Transactions on Biomedical Circuits and Systems**, IEEE, v. 4, n. 5, p. 274–280, 2010.
- INSTRUMENTS, P. **Specifying Lock-in Amplifiers**. [S.l.], 2000. v. 1002.

INSTRUMENTS, P. **What is a Lock-in Amplifier**: Technical note. [S.l.], 2000. (TN 1000).

INSTRUMENTS, Z. Hf2li lock-in amplifier. **HF2 User Manual, Revision**, v. 15296, 2017.

JOP. **A Java Optimized Processor for Embedded Real-Time Systems**. [S.l.]. Disponível em: <<http://www.jopdesign.com/thesis/thesis.pdf>>.

KAHR, B.; CLABORN, K. The lives of malus and his bicentennial law. **ChemPhysChem**, Wiley Online Library, v. 9, n. 1, p. 43–58, 2008.

KUZNETSOV, N.; LEONOV, G.; YULDASHEV, M.; YULDASHEV, R. Analytical methods for computation of phase-detector characteristics and pll design. In: IEEE. **Signals, Circuits and Systems (ISSCS), 2011 10th International Symposium on**. [S.l.], 2011. p. 1–4.

LICENSE, T. O. H. **The TAPR Open Hardware License**. 2007. http://www.tapr.org/TAPR_Open_Hardware_License_v1.0.txt.

LIMA, V. L. E. Os fármacos e a quiralidade: uma breve abordagem. **Química Nova**, v. 20, n. 6, p. 657–663, 1997.

MARCELLIS, A. D.; FERRI, G.; PATRIZI, M.; STORNELLI, V.; D'AMICO, A.; NATALE, C. D.; MARTINELLI, E.; ALIMELLI, A.; PAOLESSE, R. An integrated analog lock-in amplifier for low-voltage low-frequency sensor interface. In: IEEE. **Advances in Sensors and Interface, 2007. IWASI 2007. 2nd International Workshop on**. [S.l.], 2007. p. 1–5.

MEADE, M. Advances in lock-in amplifiers. **Journal of Physics E: Scientific Instruments**, IOP Publishing, v. 15, n. 4, p. 395, 1982.

MEADE, M. L. **Lock-in amplifiers: principles and applications**. [S.l.]: Mike Meade, 1983.

MICROCHIP. **Analog-to-Digital Converter (ADC)**. [S.l.]: Microchip, 2012. <http://ww1.microchip.com/downloads/en/DeviceDoc/70183D.pdf>. Section 16.

MOSCATI, G.; LERNER, E. *et al.* Ciências exatas e da terra. **Avaliação & Perspectivas**, 1, CNPq/Coordenação Editorial, p. 17–52, 1982.

NOTE, A. 3. about lock-in amplifiers. **Stanford Research Systems, Scientific and Engineering Instruments**, v. 1993, p. 129, 1992.

NUSSENZVEIG, H. M. **Curso de Física Básica: fluidos, oscilações e ondas, calor**. [S.l.]: Editora Blucher, 2018.

OTT, H. W.; OTT, H. W. **Noise reduction techniques in electronic systems**. [S.l.]: Wiley New York, 1988.

RESTELLI, A.; ABBIATI, R.; GERACI, A. Digital field programmable gate array-based lock-in amplifier for high-performance photon counting applications. **Review of scientific instruments**, AIP, v. 76, n. 9, p. 093112, 2005.

RUBOW, E. **Open source hardware**. [S.l.], 2008.

SANTHANALAKSHMI, M. A cmos lock-in amplifier for low-power biomedical applications. **International Journal of Biomedical Engineering and Technology**, Inderscience Publishers (IEL), v. 20, n. 1, p. 12–32, 2016.

SCOTT, J. Introduction to lock-in amplifiers. **DL Instruments Technical Notes IAN47**, 2002.

SMOLNIKAR, M.; MOHORCIC, M. A framework for developing a microchip pic microcontroller based applications. **WSEAS Transactions on Advances in Engineering Education**, v. 5, n. 2, p. 83–91, 2008.

SONNAILLON, M. O.; BONETTO, F. J. A low-cost, high-performance, digital signal processor-based lock-in amplifier capable of measuring multiple frequency sweeps simultaneously. **Review of Scientific Instruments**, AIP, v. 76, n. 2, p. 024703, 2005.

STEIN, M. L. **Interpolation of spatial data: some theory for kriging**. [S.l.]: Springer Science & Business Media, 2012.

SYSTEMS, S. S. R. **DSP Lock-In Amplifier model SR830**. [S.l.], 1999.

VOLDER, J. E. The cordic trigonometric computing technique. **IRE Transactions on electronic computers**, IEEE, n. 3, p. 330–334, 1959.

WENN, D. Implementing digital lock-in amplifiers using the dspic® dsc. **Application Note**, 2007.

XU, R. Design of a lock-in amplifier for terahertz detector and imager arrays on monolithic cmos. **Intel Science Talent Search 2013 Competition**, 2012.

XU, W.; MATHINE, D. L.; BARTON, J. K. Analog cmos design for optical coherence tomography signal detection and processing. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 55, n. 2, p. 485–489, 2008.

YANG, Z.; JIN, X.; MA, H.; JIN, Z. Cordic algorithm based digital detection technique applied in resonator fiber optic gyroscope. **Optical Fiber Technology**, Elsevier, v. 15, n. 3, p. 328–331, 2009.

Apêndice A - CÓDIGO

SIMULAÇÃO NO COMPUTADOR

```
1  for(i=0 ; i < N ; i++){
2      t[i] = (double)i / FA;
3      //sinal senoidal
4      Ref_in[i] = sin( 2.0 * PI * Fsinal * t[i] );
5      //sinal quadrado
6      //if(sin( 2.0 * PI * Fsinal * t[i] + 16) >= 0 ) Ref_in[i] = 2 ; else
7          Ref_in[i] = -2 ;
8
9  }
10
11  for(i=0 ; i < N ; i++){
12      t[i] = (double)i / FA;
13      //V_sig0[i] = A * sin( 2.0 * PI * Fsinal * t[i] + 16);
14      V_sigO[i] = A * sin( 2.0 * PI * Fsinal * t[i] );
15  }
16  // ruído
17  for(i=0 ; i < N ; i++){
18      V_sig[i]=V_sigO[i]+(8*( rand()%3 )* sin(2.0*PI*Fsinal*(rand()%40 )*t[i]
19          ]+rand()%90));
20  }
21
22  // PLL Digital
23  // encontrar o período
24  for(i=1 ; i < N ; i++){
25      if(Ref_in[i] > 0 && Ref_in[i-1] < 0 && start > -1 ){
26          end = i;
27          count++;
28      }
29  }
30  if(Ref_in[i] > 0 && Ref_in[i-1] < 0 && start == -1)
31      start = i;
32
33  pontos = ( (end - start) / count ) ;
34  Fsig = ( FA / pontos );
35
36  // gerando um seno com a frequência e fase encontrada
```

```
33  for(i=0 ; i < N ; i++){
34      V_ref[((int)start+i-1)] = sin( 2.0 * PI * Fsig * t[i]) ;
35  }
36
37  // Lock-in Digital
38  X = 0 ;
39  Y = 0 ;
40  for(i=start ; i < N ; i++){
41      X += V_sig[i] * V_ref[i];    // Sig * Vref
42      Y += V_sig[i] * V_ref[(int)(i+(pontos/4))];    // Sig * Verf (90)
43  }
44
45  X = X / (N - start);
46  Y = Y / (N - start);
47  mag = 2 * (sqrt (X*X + Y*Y) );
```

Apêndice B - FIRMWARE DO MICROCONTROLADOR

```
1  #include <xc.h>
2  #include <math.h>
3  #include "main.h"
4  /*****
5   * File:    main.c
6   * Author: Marques-carneiro , Igor
7   *
8   * Created on 24 de Agosto de 2017
9   *****/
10 //1- Selecciona o oscilador
11 _FOSCSEL(FNOSC_PRIPLL & IESO_OFF);
12 //2- Configura a troca para o oscilador
13 _FOSC( POSCMD_HS & OSCIOFNC_OFF & FCKSMCSECMD );
14 //Watchdog Timer Off
15 _FWDT(FWDTEN_OFF);
16
17 int  buffer_adc1 [SAMPLES] , i1 =0;
18 int  buffer_adc2 [SAMPLES] , i2 =0;
19
20 /*SAMP_BUFF_SIZE*/
21 int  BufferA1 [DMA_BUFF_SIZE] __attribute__((space(dma), aligned (
    DMA_BUFF_SIZE)));
22 int  BufferB1 [DMA_BUFF_SIZE] __attribute__((space(dma), aligned (
    DMA_BUFF_SIZE)));
23 int  tempRX2;
24 unsigned char Gain_DAC_A  = 0x30;
25 unsigned char Gain_DAC_B  = 0xB0;
26
27 /* * * * * *
28  *   PLL
29  * * * * * */
30
31 void initPLL(void)
32 {
```



```

33 // osc interno = FIN = 7.37 MHz
34 // osc cristal = FIN = 20 MHz
35 CLKDIVbits.DOZEN = 0;
36 CLKDIVbits.PLLPRE = 0b000;
37 CLKDIVbits.PLLPOST = 0b000;
38 PLLFBDbits.PLLDIV = 14;
39
40 OSCCONbits.CLKLOCK = 0; // Clock and PLL selections are not
    locked, configurations may be modified
41 OSCCONbits.LPOSCEN = 0; // Disable secondary oscillator
42 OSCCONbits.NOSC = 0b011; // Primary oscillator (XT, HS, EC)
    with PLL
43 OSCCONbits.OSWEN = 1; // Request oscillator switch to selection
    specified by NOSC<2:0> bits
44
45 // espera a troca para o clock principal 0b011
46 while (OSCCONbits.COSC != 0b011);
47
48 // Espera o PLL ser ativado
49 while(OSCCONbits.LOCK != 1) {};
50 }
51
52 /* * * * * * * * * * * * * *
53 * ADC1 *
54 * * * * * * * * * * * * */
55 // igor
56 void initAdc1(void)
57 {
58 AD1CON1bits.FORM = 0; //0 = ADC insigned int / 1 = int
59 AD1CON1bits.SSRC = 2; //Fonte de clock para amostragem: Timer
60 AD1CON1bits.ASAM = 1; //ADC Sample Control: Amostra inicia
    imediatamente
61 AD1CON1bits.AD12B = 0; //10-bit
62 AD1CON2bits.CSCNA = 0; //Escolhe entradas para Scan
63 AD1CON2bits.CHPS = 1; //Converte CH0 do ADC1
64 AD1CON2bits.ALTS = 0; // 1 = Enable Alternate Input Selection
65 // This bit is only used when the SSRC<2:0> bits (ADxCON1<7:5>) = 111.
66 AD1CON3bits.ADRC = 0; // 0 = Clock derivado do principal
67 AD1CON3bits.SAMC = 1; // sempre 1 para simultaneo
68 AD1CON3bits.ADCS = 1; //This bit is not used if ADxCON3<15> (ADRC) =
    1
69 //This bit only used if ADxCON1<7:5> (SSRC<2:0>) = 111
70 AD1CON1bits.ADDMABM = 0; //If this bit is set, DMA buffers are written
    in the order of conversion.
71 AD1CON2bits.SMPI = (NUM_CHS2SCAN-1); //2 entradas avaliadas
72 AD1CON4bits.DMABL = 7; //Cada buffer com 128 palavras
73 AD1CON1bits.SIMSAM = 1;

```

```

74 // AD1CON2bits.BUFM = 1; //Always starts filling buffer at address 0
    x0
75 AD1CON2bits.VCFG = 1; //0; // External VREF+ External VREF-
76 AD1CHS0bits.CH0SA = 4; // Select AN4 for CH0 +ve input
77 AD1CHS0bits.CH0NA = 0; // Select VREF- for CH0 -ve input
78 AD1CHS0bits.CH0SB = 4; // Select AN4 for CH0 +ve input
79 AD1CHS0bits.CH0NB = 0; // Select VREF- for CH0 -ve input
80 /* trecho onde selecionamos os canais e as entradas */
81 AD1CHS123bits.CH123SA = 1 ; // = CH1 positive input is AN3, CH2
    positive input is AN4, CH3 positive input is AN5
82 AD1CHS123bits.CH123NA = 0 ; // 0x = CH1, CH2, CH3 negative input is
    VREFL
83 AD1CHS123bits.CH123SB = 1 ; // = CH1 positive input is AN3, CH2
    positive input is AN4, CH3 positive input is AN5
84 AD1CHS123bits.CH123NB = 0 ; // 0x = CH1, CH2, CH3 negative input is
    VREFL
85 // ou seja , CH1 para AN# e CH2 para AN4
86 // CH1 -> AN3
87 // CH2 -> AN4
88 AD1CSSH = 0x0000;
89 AD1CSSL = 0x0000;
90 AD1CSSLbits.CSS3 = 1; //Scan em AN3
91 AD1CSSLbits.CSS4 = 1; //Scan em AN4
92
93 AD1PCFGL=0xFFFF;
94 AD1PCFGH=0xFFFF;
95 AD1PCFGLbits.PCFG3 = 0; //AN3 como entrada
96 AD1PCFGLbits.PCFG4 = 0; // set AN4 as Analog Input
97
98 IFS0bits.AD1IF = 0; //Limpa flag
99 IEC0bits.AD1IE = 0; //Desliga interrupt
100 AD1CON1bits.ADON = 1; //Liga CAD
101 }
102
103 /* * * * * * * * * * * * * * *
104 * Timer3 *
105 * * * * * * * * * * * * */
106 void initTmr3(void)
107 {
108 IPC2bits.T3IP = 0x01; // Set Timer3 Interrupt Priority Level
109 IFS0bits.T3IF = 0; // Clear Timer3 Interrupt Flag
110 IEC0bits.T3IE = 0; // 1 = Enable Timer3 interrupt
111
112 PR3 = (int) ( FCY/ (FA) );
113
114 TMR3HLD = 0x0000;
115 TMR3 = 0x0000;

```

```

116   TMR2      = 0x0000;
117
118   T2CONbits.TCS = 0;           // Internal clock (FOSC /2)
119   T2CONbits.TCKPS = 0b00;
120   T3CONbits.TON = 1;
121 }
122 /* * * * * * * * * * * * * * *
123  *           DMA                      *
124  * * * * * * * * * * * * * */
125 void initDma0(void)
126 {
127     DMA0CONbits.AMODE = 0; // = 0 Acesso indireto a DMA with Post-
        Increment mode
128     DMA0CONbits.MODE = 2; // = 2 Continuous, Modo Ping-Pong
129     DMA0CONbits.HALF = 0; //
130     DMA0CONbits.SIZE = 0; // word
131     DMA0CONbits.DIR = 0; // = 0 Read from peripheral address, write to
        DMA RAM address
132
133     DMA0PAD=(volatile unsigned int)&ADC1BUF0;
134     DMA0CNT = (DMA_BUFF_SIZE - 1);           //(SAMP_BUFF_SIZE*NUMCHS2SCAN
        )-1;
135     DMA0REQ = 13;           //Seleciona ADC1 como fonte para DMA
136
137     DMA0STA = __builtin_dmaoffset(&BufferA1);
138     DMA0STB = __builtin_dmaoffset(&BufferB1);
139
140     IFS0bits.DMA0IF = 0; //Limpa flag
141     IEC0bits.DMA0IE = 1; //Habilita interrupt de DMA m
142     DMA0CONbits.CHEN= 1; //Habilita DMA
143
144     IFS0bits.DMA0IF = 0;
145 }
146
147
148 /* * * * * * * * * * * * * * *
149  *           IOs                      *
150  * * * * * * * * * * * * * */
151 void initIOs(void)
152 {
153     TRISDbits.TRISD0=0; //Configura I/Os para Chip Enable SPI DAC1
154     TRISDbits.TRISD1=0; //Configura I/Os para Chip Enable SPI DAC2
155     TRISDbits.TRISD2=0; //Configura I/Os para Chip Enable SPI DDS
156     TRISDbits.TRISD4=0; //Configura I/Os
157     TRISDbits.TRISD5=0;
158     TRISDbits.TRISD6=0; //Configura I/Os
159     TRISDbits.TRISD7=0;

```

```

160  TRISGbits.TRISG0=0;  //Configura I/Os
161  TRISGbits.TRISG1=0;
162  TRISGbits.TRISG2=0;  //Configura I/Os
163  TRISGbits.TRISG3=0;
164  TRISGbits.TRISG6=0;
165  TRISGbits.TRISG7=1;
166  TRISGbits.TRISG8=0;
167      TRISBbits.TRISB0 = 1;          // The port pins that are desired as
      analog inputs must have
168  TRISBbits.TRISB1 = 1;  // their corresponding TRIS bit set (input).
169  }
170
171  unsigned char FLAG_DMA_WRITE = 0;
172  unsigned char FLAG_CPU_READ  = 0;
173
174  /* * * * * * * * * * * * * * *
175   *      DMA                      *
176   * * * * * * * * * * * * * */
177  void __attribute__((interrupt , no_auto_psv)) _DMA0Interrupt(void)
178  {
179      static int DmaBuffer = 0;
180      int i;
181
182      if(DmaBuffer == 0)
183      {
184          if (FLAG_CPU_READ == 0 )
185          {
186              FLAG_DMA_WRITE = 1;
187              for ( i = 0; i < (DMA_BUFF_SIZE/NUM_CHS2SCAN) ; i++)
188              {
189                  buffer_adc1[i1] = BufferA1[i*NUM_CHS2SCAN];
190                  buffer_adc2[i1] = BufferA1[i*NUM_CHS2SCAN+1];
191                  i1++;
192                  if( i1 >= SAMPLES)
193                  {
194                      i1 = 0;
195                      IEC0bits.DMA0IE = 0;
196                  }
197              }
198          }
199      }
200      else
201      {
202          if (FLAG_CPU_READ == 0 )
203          {
204              FLAG_DMA_WRITE = 1;
205              for ( i =0; i < (DMA_BUFF_SIZE/NUM_CHS2SCAN) ; i ++ )

```

```

206     {
207         buffer_adc1[i1] = BufferB1[i*NUMCHS2SCAN];
208         buffer_adc2[i1] = BufferB1[i*NUMCHS2SCAN+1];
209         i1++;
210         if( i1 >= SAMPLES)
211         {
212             i1 = 0;
213             IEC0bits.DMA0IE = 0;
214         }
215     }
216 }
217 }
218 DmaBuffer ^= 1;
219 IFS0bits.DMA0IF = 0; //Limpa flag de interrupt
220 FLAG_DMA_WRITE = 0;
221 }
222
223 /* * * * * * * * * * * * * * *
224  *      delay                      *
225  * * * * * * * * * * * * * */
226 void Delay_ms(unsigned int N)
227 {
228     unsigned int j,i;
229     while(N--)
230         for(j=0;j < MILLISEC;j++);
231 }
232
233 void InitUART2(void)
234 {
235     U2MODE = 0x0000;
236     U2MODEbits.UARTEN = 0;
237     // U2BRG = (FCY/(16*UART2.BAUD))-1
238     // U2BRG = (37M/(16*9600))-1
239     // U2BRG = 240
240     U2BRG = 42; // 42 = 57600 e 259 = 9600
241     //U2MODEbits.RTSMD = 1;
242     U2STA = 0x0000;
243     IPC7 = 0x4400;
244     IFS1bits.U2TXIF = 0; // Clear the Transmit Interrupt Flag
245     IEC1bits.U2TXIE = 0; // Enable Transmit Interrupts
246     IFS1bits.U2RXIF = 0; // Clear the Recieve Interrupt Flag
247     IEC1bits.U2RXIE = 0; // Enable Recieve Interrupts
248     U2MODEbits.UARTEN = 1;
249     U2STAbits.UTXEN = 1;
250     Delay_ms(10);
251 }
252

```

```

253  /* * * * * * * * * * * * * *
254  *          envio de CHR          *
255  * * * * * * * * * * * * */
256  void txCHR(unsigned char dado)
257  {
258      while (!U2STAbits.TRMT);
259      U2TXREG=dado;
260      while (!U2STAbits.TRMT);
261      IFS1bits.U2TXIF = 0;
262  }
263
264  void lockin_pc ()
265  {
266      int cont;
267      char dado;
268      for ( cont = 0; cont < SAMPLES; cont++)
269      {
270          buffer_adc1[cont] = 0;
271          buffer_adc2[cont] = 0;
272      }
273
274      FLAG_DMA_WRITE = 0;
275      FLAG_CPU_READ  = 0;
276      LED_TESTE =0;
277
278      while (1)                //Loop principal
279      {
280          while(!IFS1bits.U2RXIF);
281          LED_TESTE = ~LED_TESTE;
282          dado = U2RXREG;
283          IFS1bits.U2RXIF = 0;
284          txCHR(dado);
285          IEC0bits.DMA0IE = 0;
286
287          for(cont=0; cont < SAMPLES; cont++)
288          {
289              dado = buffer_adc1[cont];
290              txCHR(dado);
291
292              dado = (buffer_adc1[cont]>>8);
293              txCHR(dado);
294          }
295
296          for(cont=0; cont < SAMPLES; cont++)
297          {
298              dado = buffer_adc2[cont];
299              txCHR(dado);

```

```
300
301         dado = (buffer_adc2[cont]>>8);
302         txCHR(dado);
303     }
304
305     txCHR('F');
306     txCHR('I');
307     txCHR('M');
308     txCHR(13);
309     txCHR(10);
310
311     il = 0;
312     IFS0bits.DMA0IF = 0; //Limpa flag
313     IEC0bits.DMA0IE = 1; //Habilita interrupt de DMA
314     DMA0CONbits.CHEN = 1; //Habilita DMA
315     FLAG_CPU_READ = 0;
316 }
317 }
318
319
320 void lockin_igor()
321 {
322     int cont;
323     char dado;
324     int V_ref[SAMPLES];
325     int pontos=0;
326     char txt[50];
327     double t[SAMPLES];
328     int X = 0;
329     int Y = 0;
330     int mag = 0;
331     int P = 0;
332
333     for ( cont = 0; cont < SAMPLES; cont++)
334     {
335         buffer_adc1[cont] = 0;
336         buffer_adc2[cont] = 0;
337         V_ref[cont] = 0;
338     }
339     FLAG_DMA_WRITE = 0;
340     FLAG_CPU_READ = 0;
341     LED_TESTE = 0;
342
343     while (1) //Loop principal
344     {
345         while(!IFS1bits.U2RXIF);
346         LED_TESTE = ~LED_TESTE;
```

```

347         dado = U2RXREG;
348         IFS1bits.U2RXIF = 0;
349         while(il != 0 );
350         IEC0bits.DMA0IE = 0;
351         // simulando um sinal de entrada em kHz
352         //simular_sinal(buffer_adc1 , 20, 0); // sinal medido
353         //simular_sinal(buffer_adc2 , 20, 0); // referencia in
354
355         adc_shift(buffer_adc1);
356         adc_shift(buffer_adc2);
357
358         pontos = adc_pll(buffer_adc2 , V_ref);
359
360         lockin(buffer_adc1 , V_ref , pontos);
361
362         il = 0;
363         IFS0bits.DMA0IF = 0; //Limpa flag
364         IEC0bits.DMA0IE = 1; //Habilita interrupt de DMA
365         DMA0CONbits.CHEN = 1; //Habilita DMA
366         FLAG_CPU_READ = 0;
367     }
368 }
369
370 /* * * * * * * * * * * * * *
371  *      Principal      *
372  * * * * * * * * * * * * */
373 int main (void)
374 {
375     int cont;
376     char dado;
377
378     initPLL (); //Inicializa PLL
379     initAdc1 (); //Inicializa ADC
380     initDma0 (); //Inicializa DMA
381     initTmr3 (); //Inializa Timer3
382     initIOs (); //Inicializa IOs
383
384     InitUART2(); //Inicializa UART
385     Delay_ms(2);
386
387     while (1) //Loop principal
388     {
389         while(!IFS1bits.U2RXIF);
390         dado = U2RXREG;
391         IFS1bits.U2RXIF = 0;
392         if(dado == 'I')
393             lockin_igor();

```



```

394         else if(dado == 'A')
395             lockin_pc();
396
397     }
398
399 }
```

```

1  #include <xc.h>
2  #include "lockin_dspic.h"
3  /*****
4   * File:    main.h
5   * Author: Marques-carneiro , Igor
6   *
7   * Created on 24 de Agosto de 2017
8   *****/
9  // Frequencia final de operação do microcontrolador
10 #define FCY (unsigned long int) 40000000
11
12 /* Boud rate do UART */
13 #define UART2BAUD (long int) 57600
14
15 /* Frequencia de amostragem do ADC */
16 #define FA (long int) 250000 // Hz
17
18 /* Informacoes do DDS */
19 #define CLOCK 10000000 // clock do DDS
20 #define DDS_q 4294967296 // 2^32 (Coeficiente frequencia DDS)
21 #define F_DDS 1000000 // 1MHz (frequencia do sinal de saida do DDS )
22
23 #define MILLISEC FCY/10000
24
25 /* Funcoes de delay */
26 #define __delay_ms(d) { __delay32( (unsigned long) (((unsigned long long
    ) d)*(FCY)/1000ULL)); }
27 #define __delay_us(d) { __delay32( (unsigned long) (((unsigned long long
    ) d)*(FCY)/1000000ULL)); }
28
29 /* * * * * *
30  *   Configuração de variáveis   *
31  * * * * * */
32 #define SAMPLES 1024
33 #define PI 3.14159265358979323846
34
35 /* * * * * *
36  *   Configuração de I/Os   *
37  * * * * * */
38 #define LED_TESTE PORTDbits.RD7
```

```

39
40 #define PORT_SPLCS_DAC1 PORTDbits.RD0
41 #define PORT_SPLCS_DAC2 PORTDbits.RD1
42 #define PORT_SPLCS_DDS PORTDbits.RD2
43
44 /* * * * * *
45  * Definição de constantes/variáveis *
46  * * * * *
47 #define DMA_BUFF_SIZE 128
48 #define NUMCHS2SCAN 2

```

```

1 #include "xc.h"
2 /*****
3  * File:   lockin_dspic.c
4  * Author: Marques-carneiro, Igor
5  *
6  * Created on 24 de Agosto de 2017
7  *****/
8
9 #include "main.h"
10
11 int simular_sinal(int *b[], int setf, int p)
12 {
13     int cont;
14     for ( cont = p; cont < SAMPLES; cont++)
15         b[cont] = (int) (500.0 * (1 + sin(2.0 * PI * (double)(
16     setf * 1000 ) * (double)((double)cont / FA)))) ;
17
18 int gerar_sin_lut(int *b[], int setf, int p)
19 {
20     int cont;
21     for ( cont = 0; cont < SAMPLES; cont++)
22         b[p+cont] = (int) (10.0 * (sin(2.0 * PI * (double)setf * (
23     double)((double)cont / FA)))) ;
24
25     if(p != 0 ){
26         for(cont=p ; cont >= 0 ; cont--)
27             b[p-cont] = (int) (10.0 * ( sin( (2.0 * PI * (double)
28     setf * (double)((double)cont / FA)) + PI ) ) ) ;
29     }
30
31 int adc_shift(int *buffer[])
32 {
33     int i;
34     int max, min;

```

```

34
35     max = 0;
36     min = 32767;
37     for(i=1 ; i < SAMPLES ; i++){
38 if( buffer[i] > max)
39     max = buffer[i];
40         if( buffer[i] < min )
41     min = buffer[i];
42     }
43
44 for(i=0 ; i < SAMPLES ; i++)
45     buffer[i] -= ( max + min) / 4 ;
46 }
47
48 int adc_pll(int *buffer[] , int *V_ref[])
49 {
50     int i;
51     int count = 0;
52     int start = 0;
53     int end = -1;
54     int temp, tempold = 0;
55     char txt[100];
56     int pontos;
57
58     // encontrar o periodo
59     for(i=1 ; i < SAMPLES ; i++){
60         temp = buffer[i];
61
62         if( ( temp & 0x8000 ) != 0x8000){
63             if( ( tempold & 0x8000 ) == 0x8000){
64                 if(!start){
65                     start = i;
66                 }else{
67                     end = i;
68                     count++;
69                 }
70             }
71         }
72         tempold = temp;
73     }
74
75     pontos = (int)( (double)(end - start) / (double)count ) ;
76     gerar_sin_lut(V_ref, (int)( ( FA / (double)( ((double)(end -
77         start) / (double)count) ) ) ), start);
78     return pontos;
79 }

```

```

80 int lockin(int *V_sig , int *V_ref , int pontos )
81 {
82     int i;
83     double temp_x = 0;
84     double temp_y = 0;
85     double temp_f = 0;
86     char txt[100];
87     double mag;
88     double P;
89     unsigned char *dado;
90
91     // Lock-in Digital
92     for(i=0 ; i < (SAMPLES - (pontos/4)) ; i++){
93         temp_x = temp_x + (double)(V_sig[i] * V_ref[i]);
94         temp_y = temp_y + (double)(V_sig[i] * V_ref[i+(pontos/4)]); //
95         // Sig * Verf (90)
96     }
97     temp_x /= (double)(SAMPLES);
98     temp_y /= (double)(SAMPLES);
99
100     mag =( (2.0/10.0) * (sqrt((( temp_x * temp_x ) + (temp_y * temp_y) )
101         ))) );
102     P = atan2(temp_x,temp_y);
103
104     // modo de envio bytes
105     dado = (unsigned char *) &mag;
106     txCHR(*dado++);
107     txCHR(*dado++);
108     txCHR(*dado++);
109     txCHR(*dado);
110
111     dado = (unsigned char *) &P;
112     txCHR(*dado++);
113     txCHR(*dado++);
114     txCHR(*dado++);
115     txCHR(*dado);
116
117     dado = (unsigned char *) &temp_x;
118     txCHR(*dado++);
119     txCHR(*dado++);
120     txCHR(*dado++);
121     txCHR(*dado);
122
123     dado = (unsigned char *) &temp_y;
124     txCHR(*dado++);
125     txCHR(*dado++);
126     txCHR(*dado++);
127     txCHR(*dado);

```

```
125     txCHR(*dado++);
126     txCHR(*dado);
127
128     temp_f = ( FA / pontos);
129     dado = (unsigned char *)&temp_f;
130     txCHR(*dado++);
131     txCHR(*dado++);
132     txCHR(*dado++);
133     txCHR(*dado);
134
135 }
```

```
1  #include <xc.h>
2  /*****
3   * File:    lockin_dspic.h
4   * Author: Marques-carneiro , Igor
5   *
6   * Created on 24 de Agosto de 2017
7   *****/
8  #include <math.h>
9  #include "main.h"
10 #include <stdio.h>
11
12 int simular_sinal(int *, int ,int);
13 int gerar_sin_lut(int *, int ,int);
14 int adc_shift(int *);
15 int adc_pll(int *, int *);
16 int lockin(int * , int * , int );
```

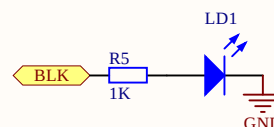
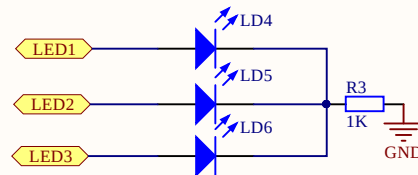
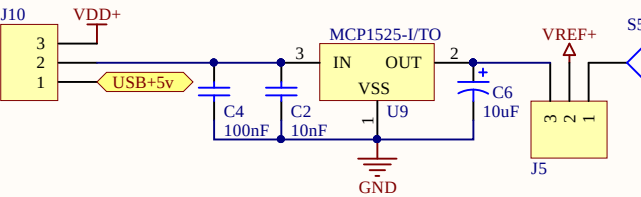
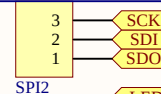
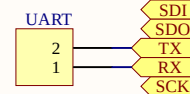
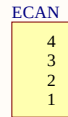
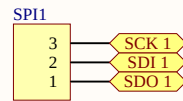
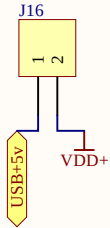
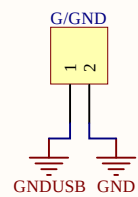
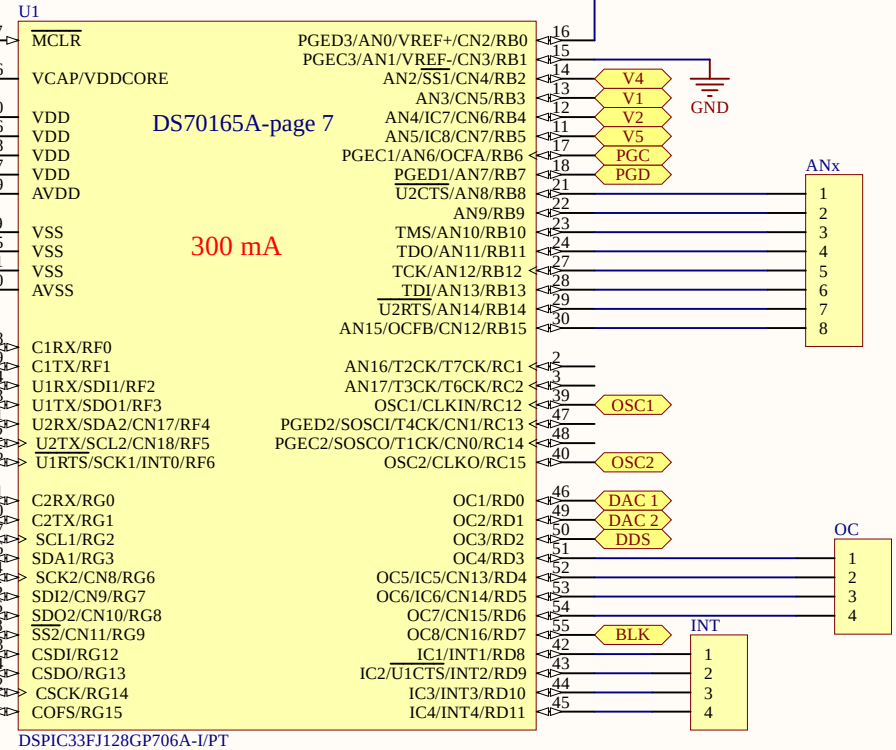
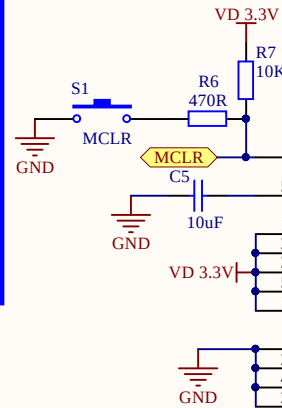
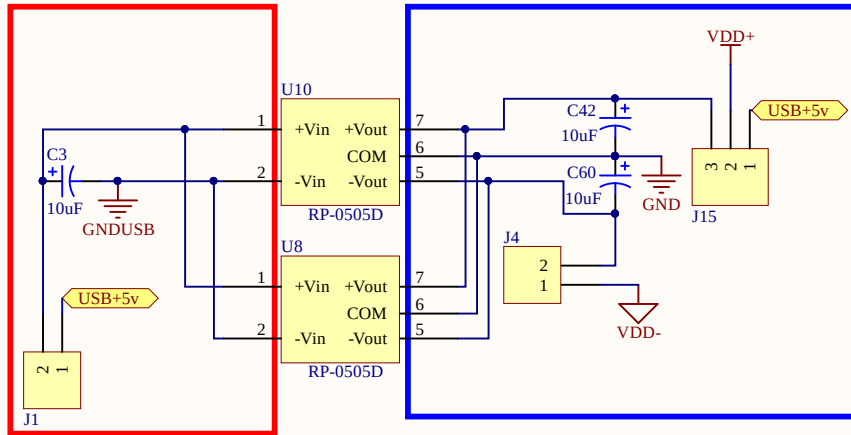
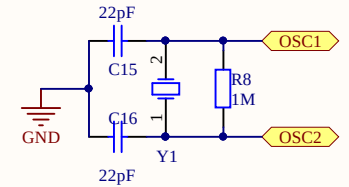
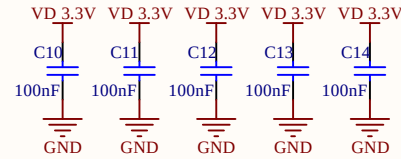
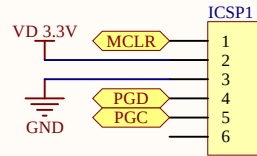
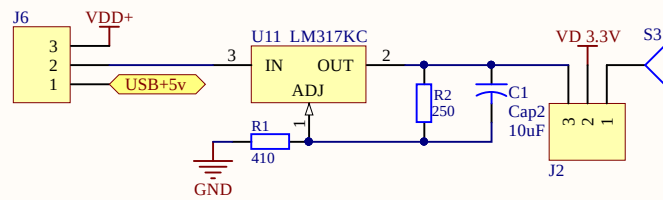
Apêndice C - Diagramas do circuito final

1

2

3

4



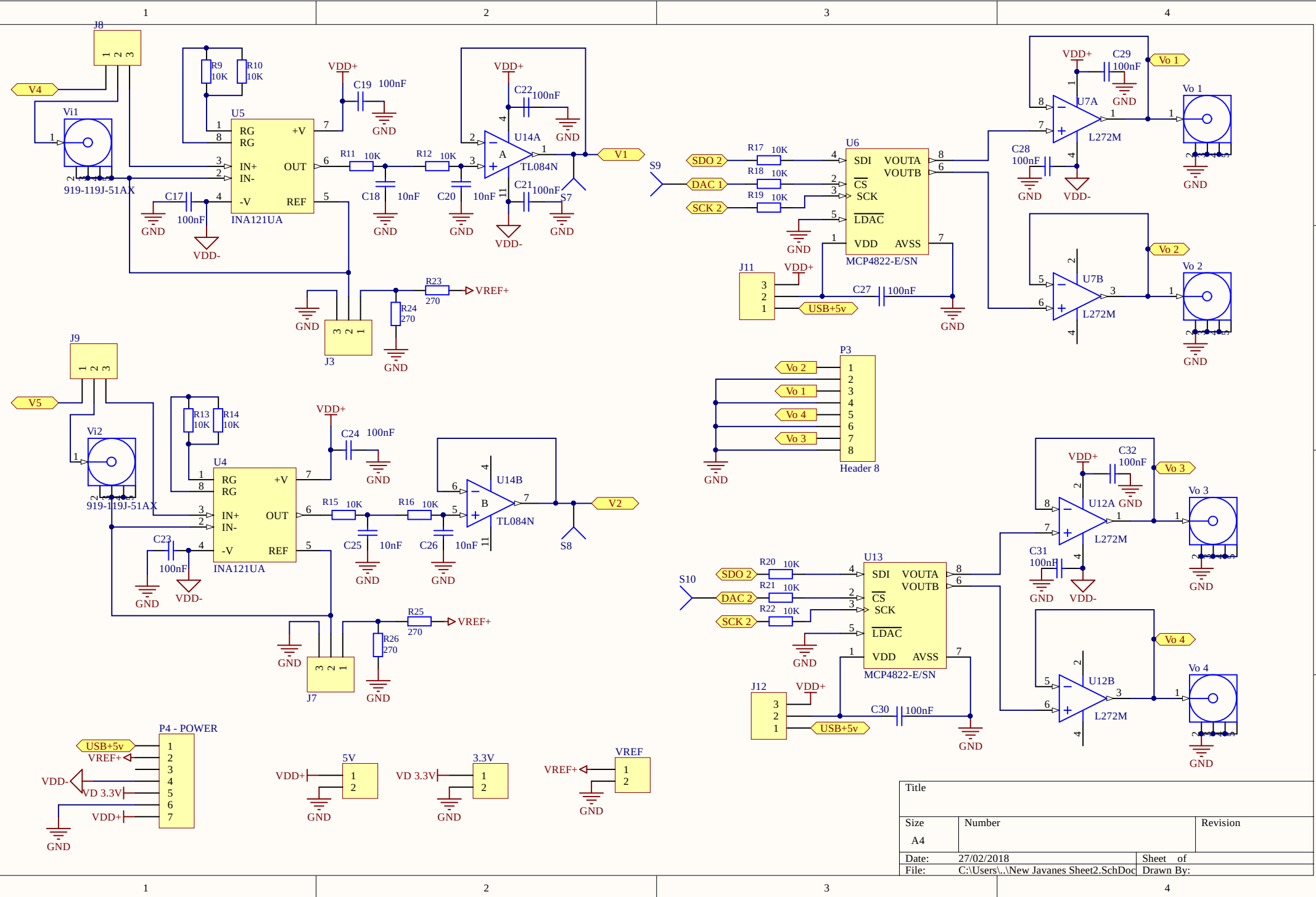
Title		
Size	Number	Revision
A4		
Date:	27/02/2018	Sheet of
File:	C:\Users\...\New Javanes Sheet1.SchDoc	Drawn By:

1

2

3

4



Apêndice D - Preparo das amostras e porta amostra

Síntese do trabalho de (FAUTH, 2017) referente ao Preparo das amostras e porta amostra.

Para o preparo do porta amostra, foi utilizado um tubo de PVC comercial com aproximadamente uma polegada de diâmetro, 10 cm de comprimento e com uma tampa em cada extremidade. Cada tampa, na verdade, é um suporte para janela de quartzo temperado, como apresentado na Figura D.1.



FIGURA D.1 – Esquema do tubo porta amostra para o polarímetro. Fonte (FAUTH, 2017)

O corpo do suporte de lente foi fabricado em Technyl com um O-Ring para garantir

a vedação, permitindo serem acoplados e removidos facilmente em tubos de diferentes comprimentos. As lentes de quartzo temperado foram coladas ao suporte com cola de silicone. Após a montagem das tampas no tubo, um furo no cano permite o envase da solução com o auxílio de uma seringa e sua retirada pode ser feita removendo o suporte de uma das extremidades.

As amostras líquidas utilizadas são soluções de água com açúcar em diferentes concentrações: 400; 300; 200; 100; 50; 25 e 12,5 mg/mL preparadas conforme o esquema da Figura D.2 no Laboratório de Produção de Materiais Avançados (LPMA) do CBPF. O método da diluição foi empregado no preparo das amostras com o intuito de diminuir o erro na concentração de cada amostra. Desta forma, prepara-se uma solução de concentração 400 mg/mL misturando 120 g de açúcar com água deionizada até atingir 300 mL de solução. Em seguida, as outras soluções são obtidas por diluição conforme o esquema.

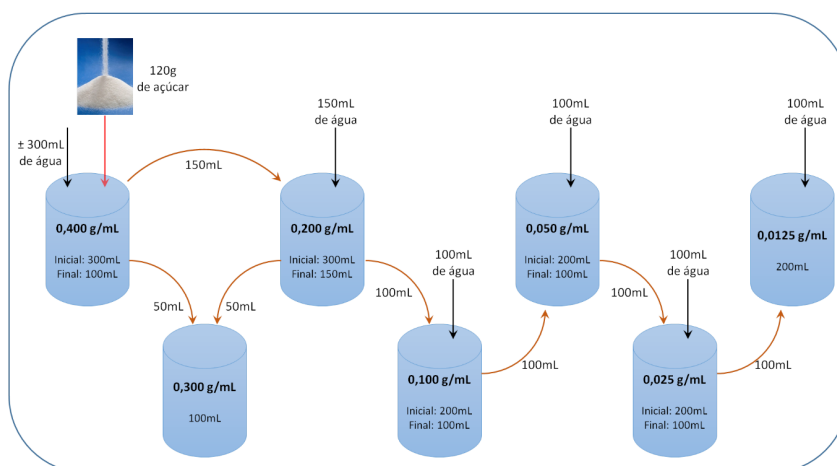


FIGURA D.2 – Esquema de preparo das soluções de amostras. Fonte (FAUTH, 2017)

Apêndice E - APLICAÇÕES DO AMPLIFICADOR Lock-in

A lista a seguir mostra algumas aplicações conhecidas para o amplificadores lock-in. Esta compilação foi feita pela equipe no EG&G Brookdeal ¹.

- Espectroscopia de absorção
- Pontes de A.C.
- Padrões de antena
- Espectroscopia astronômica
- Absorção atômica
- Resposta em frequência do amplificador de áudio
- Audiometria
- Espectroscopia Auger
- Calibração do filtro
- Espectroscopia de fluorescência
- Medições de resposta de frequência
- Medições de mudança de frequência
- Efeito Hall: frequência única
- Efeito Hall: dupla frequência
- Espectroscopia de infra-vermelho (próximo e distante)
- Interferometria
- Medições de resposta de estímulo bio-médico
- Gráfico de Bode
- Estabilização de Klystron
- Microfônica de cócleas
- Medições de rejeição de modo comum
- Medições de impedância complexa
- Contato com medidas potenciais
- Crosstalk em cabos, amplificadores, etc.
- C-V conspirando
- Interferometria de cubo
- Efeito De Haas Van Alphen
- Densitometria
- Compensação de detecção
- Medições de deslocamento
- Medições Doppler
- Medições Óptica de feixe duplo
- Pesquisa a laser
- Medição da ondulação da linha em fontes de alimentação de amplificador
- Medições de campo magnético
- Magnetometria
- Estudos de magnetoresívia
- Avaliação de Marx
- Espectroscopia de massa
- Calibração do microfone
- Reflexões de microondas, atenuação
- Espectroscopia de microondas
- Medição do conteúdo de umidade

¹EG&G, Conhecido como Edgerton, Germeshausen e Grier Inc. É empresa prestadora de serviços para a defesa nacional dos Estados Unidos, fornecendo serviços de gestão e tecnologia.

- (CG)
- Espectroscopia de feixe molecular
- Espectroscopia R.M.N.
- Espectroscopia NEM,
- Parcelas de Nyquist
- Teste de falhas de corrente de Foucault
- Mudança de borda em GaAs
- Eletroquímica
- Electroluminescência
- Espectroscopia de emissão
- Espectroscopia E.P.R./e.s.r.
- Medição de ganho do amplificador operacional
- Medições de derivado óptico
- Fotometria
- Pesquisa de física de plasma
- Pyrometria
- Medições de tensão-deformação
- Controle de temperatura
- Medição de temperatura
- Medições de torque
- Radiometria
- Espectroscopia Raman
- Medições Ratiométricas
- Termometria de resistência
- R.F. medições
- Espectroscopia ultravioleta
- Espectroscopia visível
- Segundo som
- Medições sísmicas
- Pesquisa de semicondutores
- Compensação de origem
- Espectrofotometria
- Medição de tensão
- Medições do sinal de Whistler
- Medidas da função de trabalho
- Módulo de Young
- Efeito Zeeman