

JEFERSON LEANDRO NOGUEIRA PELISON

MEMSCLE – MEDIDOR MULTISENSORIAL CAPACITIVO PARA O  
USO EM LÍNGUA ELETRÔNICA

São Paulo  
2012

## AGRADECIMENTOS

Durante toda a minha vida universitária e mais especificamente na criação deste trabalho final existiram pessoas e instituições que contribuíram com a minha formação. Nestes agradecimentos, por não ser possível citar todos que contribuíram com a minha formação citarei apenas aqueles mais diretamente envolvidos na criação deste projeto.

Agradeço toda a sociedade paulista e a Universidade de São Paulo, mais especificamente a Escola Politécnica por financiar os meus estudos e tornar possível este projeto. Agradeço a POLI-USP por ceder os laboratórios e equipamentos necessários para tal.

O meu muito obrigado ao professor Dr. Fernando Josepetti Fonseca pela paciência, pela orientação por acreditar no meu trabalho. Igualmente os meus agradecimentos a toda a equipe do GEM (Grupo de Eletrônica Molecular) por acreditar que o meu trabalho funcionaria e por toda a ajuda nos laboratórios etc. Agradeço ao Dr. Guilherme Braga por ceder a língua eletrônica para os testes e ao Mestre Roberto Cavallari por me ajudar com técnicas de dissertação para teses.

Agradeço também o professor Dr. Sergio Takeo Kofuji por todas as dicas nas aulas de projeto de formatura. Também lembro aqui o Sr. Jair do laboratório pelo auxílio na soldagem de CIs com encapsulamento LQFP.

O meu muito obrigado para a minha família por todo o suporte e base que me forneceram ao longo de toda a vida universitária e pela paciência por aguardar a conclusão do meu trabalho.

## RESUMO

Com a crescente necessidade de reproduzirmos os sentidos humanos pelas máquinas surge a criatividade para a criação de sistemas cada vez mais complexos para atingir este fim. A língua eletrônica não fica para trás. A mesma é constituída por um arranjo de sensores capacitivos com o objetivo final de imitar e até superar a língua humana. Para realizar medições em laboratório com ela é possível usar equipamentos precisos, porém que são grandes e pesados, inviabilizando seu uso no "campo". Neste contexto é então inserido este trabalho. O sistema MEMSCLE (Medidor Multisensorial Capacitivo para o Uso em Língua Eletrônica) é basicamente o sistema complexo de medida existente no laboratório feito de maneira compacta e integrada, dando-lhe portabilidade, possibilitando levar a língua eletrônica para a indústria e para o uso em "campo", como deve ser o destino final de todas as tecnologias desenvolvidas na universidade. O sistema desenvolvido apresentou uma precisão nas medições de capacitores comerciais e dos sensores usados na língua eletrônica comparável aos equipamentos de medição usados atualmente com a língua eletrônica. Adicionalmente, apresentou grande estabilidade na resposta elétrica dos sensores. Este trabalho demonstra a possibilidade de passar tecnologia da universidade para a indústria através do uso da Engenharia para compactação de sistemas e redução de custos.

Palavras-chave: Língua Eletrônica, Compactação de sistemas, Tecnologia para a indústria.

## ABSTRACT

The humanity wants more and more to reproduce the natural senses of our body through the machines we made. In this contest appears the creativity and skills to create more and more complex electrical and electronical systems to achieve this goal. It is not different with the electronic tongue. It is composed by an array of capacitive sensor with the final goal of mimic the human tongue and achieve a sensibility higher than the one obtained with it. When samples are analyzed with the electronic tongue in laboratories, it is possible to use precise equipments, which are big and heavy, making its use in the "field" not possible. This work is inserted in that context. The MEMSCLE system (Capacitive Multisensorial Meter for the Electronic Tongue Use) is basically the complex system of the labs designed in a compact and integrated way, conferring to it portability, making possible to transfer the electronic tongue to the industry, as it should be the final destination of all technology developed in the university environment. The developed system has shown a comparable precision with the measuring equipments currently used with the electronic tongue when measuring commercial capacitors and the sensor array. Additionally, it showed great stability at sensors' electrical response. This work demonstrates the possibility to transfer technology from the University to the Industry through the use of engineering to compact electronic systems and reduce costs.

Key-words: Electronic Tongue, System's Integration, Industrial Technology

## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 1: Sensor da língua eletrônica: (a) Estrutura do sensor destacando as camadas que o compões. (b) fotografia dos microeletrodosinterdigitados e dos terminais. <i>Adaptado da referência [2]</i> ..... | 15 |
| Figura 2: Diagrama funcional de um multiplexador analógico com 4 entradas e 1 saída. <i>Extraído da referência [13]</i> .....  | 20 |
| Figura 3: Circuito esquemático da Ponte de Wheatstone AC. <i>Extraído da referência [14]</i>   | 21 |
| Figura 4: Circuito esquemático da Ponte de Schering. <i>Adaptado da referência [14]</i> .....  | 22 |
| Figura 5: Esquema para medir a diferença de fase causada por um DUT. <i>Extraído da referência [15]</i> .....  | 23 |
| Figura 6: Circuito equivalente de um filtro passa baixa. <i>Extraído da referência [16]</i> .....  | 24 |
| Figura 7: Modelo equivalente de um capacitor real. <i>Desenhado no Altium PCAD 2006</i> ...  | 26 |
| Figura 8: Soma vetorial para a determinação da defasagem por método algébrico. <i>Desenhado no Microsoft Power Point 2010</i> .....  | 26 |
| Figura 9: Eixo do modelo equivalente do capacitor. <i>Desenhado no Microsoft Power Point 2010</i> .....  | 27 |
| Figura 10: Circuito teórico do RLC meterda HP. <i>Extraído da referência [18]</i> .....  | 28 |
| Figura 11: Placa da Texas a ser utilizada.....   | 30 |
| Figura 12: Diagrama de blocos do sistema.....  | 34 |
| Figura 13: Circuito esquemático da fonte de alimentação do sistema.....  | 35 |
| Figura 14: Circuito esquemático da multiplexação dos sensores .....  | 35 |
| Figura 15: Circuito esquemático de amostragem e medição de tensão .....  | 36 |
| Figura 16: Circuito esquemático para identificação de fase.....  | 36 |
| Figura 17: PCI sendo montada: (a) fonte de alimentação; (b) circuito de multiplexação analógica dos sensores .....   | 38 |
| Figura 18: Placa PCI quase 100% montada .....  | 39 |
| Figura 19: Circuito montado executando testes com o gerador de funções .....   | 41 |
| Figura 20: Circuitos amplificadores diferenciais em funcionando .....  | 42 |
| Figura 21: Equipamentos necessários para a medida da língua eletrônica atualmente ....   | 43 |
| Figura 22: Língua eletrônica sendo medida através do equipamento atual.....  | 44 |
| Figura 23: Projeto MEMSCLE em funcionamento .....  | 44 |

|   |    |
|---|----|
| Figura 24: Projeto MEMSCLE (esquerda) e Língua Eletrônica (direita) ..... | 45 |
| Figura 25: Página de apresentação do sistema MEMSCLE .....                | 46 |
| Figura 26: Página de controle do sistema MEMSCLE .....                    | 46 |
| Figura 27: Layers da PCI .....  | 83 |
| Figura 28: PCI no software de projeto .....                               | 84 |

## LISTA DE GRÁFICOS

|   |    |
|---|----|
| Gráfico 1: Resposta em frequência de um filtro passa baixa de primeira ordem: (a) ganho de tensão em dB em função da frequência; (b) deslocamento de fase em função da frequência. <i>Extraído da referência [17]</i> ..... | 25 |
| Gráfico 2: Gráficos da comunicação entre o microcontrolador e o Gerador de sinais .....   | 40 |
| Gráfico 3: Funcionamento do circuito gerador .....  | 40 |
| Gráfico 4: Medida de tensão através do circuito com amplificadores operacionais .....   | 42 |
| Gráfico 5: Dados obtidos com capacitores comerciais .....   | 47 |
| Gráfico 6: Repetibilidade do sistema MEMSCLE.....   | 48 |
| Gráfico 7: Comparação do sistema MEMSCLE com o HP 4263A.....  | 48 |

## SUMÁRIO

|  |           |
|--|-----------|
| <b>1. INTRODUÇÃO.....</b>                                      | <b>11</b> |
| 1.2 OBJETIVO.....  | 12        |
| 1.2.1 Objetivo específico.....                                 | 12        |
| 1.3 JUSTIFICATIVA DO TRABALHO.....                             | 13        |
| 1.4 ESTRUTURA DO TRABALHO.....                                 | 13        |
| <b>2. REFERENCIAL TEÓRICO .....</b>                            | <b>14</b> |
| 2.1 UMA BREVE DESCRIÇÃO DA LÍNGUA ELETRÔNICA.....              | 14        |
| 2.2 APLICAÇÕES DA LÍNGUA ELETRÔNICA.....                       | 15        |
| 2.3 MULTISENSORIAMENTO PARA LÍNGUA ELETRÔNICA .....            | 16        |
| 2.4 PONTES RLC (RESISTOR, INDUTOR E CAPACITOR) .....           | 18        |
| 2.5 MULTIPLEXAÇÃO .....  | 19        |
| 2.6 PONTE DE WHEATSTONE CAPACITIVA.....                        | 21        |
| 2.7 MÉTODO SENSOR DE FASE .....                                | 22        |
| 2.8 MÉTODO FILTRO PASSA BAIXA.....                             | 24        |
| 2.9 MÉTODOS DE MEDIDA PONTE RLC HP4286 .....                   | 28        |
| <b>3. METODOLOGIA.....</b>                                     | <b>29</b> |
| 3.1 DESCRIÇÃO DE COMPONENTES CRÍTICOS .....                    | 29        |
| 3.2 DIAGRAMA DE BLOCOS.....                                    | 33        |
| 3.3 ESQUEMA TÉCNICO DO CIRCUITO.....                           | 34        |
| <b>4. RESULTADOS E DISCUSSÕES.....</b>                         | <b>37</b> |
| 4.1 PLACA DE CIRCUITO IMPRESSO PROJETADA E CONFECCIONADA ..... | 37        |
| 4.2 CIRCUITOS EM FUNCIONAMENTO .....                           | 39        |
| 4.2.1 Gerador de funções em funcionamento.....                 | 39        |



|  |           |
|--|-----------|
| 4.2.2 Circuitos medidores de tensão diferencial.....                                     | 41        |
| 4.2.3 Projeto MEMSCLE completo e medindo a língua eletrônica .....                       | 43        |
| 4.3 INTERFACE GRÁFICA .....  | 45        |
| 4.4 RESULTADOS COM CAPACITORES COMERCIAIS.....   | 47        |
| 4.5 RESULTADOS COM A LÍNGUA ELETRÔNICA .....   | 47        |
| 4.6 ANÁLISE DE CUSTO DO PROJETO.....   | 49        |
| <b>5. CONCLUSÕES.....</b>  | <b>53</b> |
| <b>6. REFERÊNCIAS BIBLIOGRÁFICAS.....</b>  | <b>54</b> |
| <b>ANEXOS .....</b>  | <b>58</b> |
| ANEXO A: PEDAÇOS DE CÓDIGO "C" DO PROGRAMA DO MICROCONTROLADOR .....                     | 58        |
| ANEXO B: PEDAÇOS DE CÓDIGO "HTML" E "JAVASCRIPT" DAS PÁGINAS EMBARCADAS NO SISTEMA ..... | 66        |
| ANEXO C: ESQUEMA TÉCNICO COMPLETO DO SISTEMA .....                                       | 76        |
| ANEXO D: PROJETO DA PLACA DE CIRCUITO IMPRESSO.....                                      | 83        |

## 1. INTRODUÇÃO

Desde o início da ciência moderna a humanidade procura uma forma de reproduzir nossos sentidos de forma artificial. Visão, Audição, Tato, Olfato e Paladar sempre estiveram na pauta de estudos de cientistas renomados do nosso planeta.

Graças ao avanço tecnológico atual já é possível reproduzir com boa precisão os sentidos da Visão e Audição através das câmeras de vídeo e dos microfones que hoje em dia está nos lares de boa parte da população mundial.

Este trabalho fornecerá suporte a um dos sentidos dentre os citados anteriormente: Paladar. A ideia do desenvolvimento do que chamamos de língua eletrônica já é antiga. Mas somente com a tecnologia atual surgiu a possibilidade de criarmos sensores que tendem a se aproximar dos inúmeros sensores que constituem a língua humana.

A criação de sensores é uma coisa, mas a leitura e interpretação dos dados fornecidos pelos mesmos é outra. Nesta monografia serão utilizados os sensores de líquidos criados e desenvolvidos pelo Grupo de Eletrônica Molecular do Departamento de Sistemas Eletrônicos da Escola Politécnica da Universidade de São Paulo. O grupo em questão utiliza, como base para realizar medidas e análises de líquidos, um arranjo de sensores. No entanto para a medição e interpretação dos dados fornecidos pelos sensores é necessário equipamentos de laboratórios relativamente grandes e precisos.

Os sensores são basicamente capacitores construídos com um polímero orgânico como dielétrico. O polímero varia de acordo com a sensibilidade desejada e com o líquido a ser estudado. Para a medição do arranjo de sensores é usado um multiplexador, uma ponte RCL (Resistência, Capacitância e Indutância) com boa precisão e um computador para interpretar os dados.

Este trabalho tornará possível a medição do sistema através de um circuito relativamente preciso e barato e uma placa de circuito impresso relativamente pequena de forma que o sistema possa ser considerado portátil.

Ao final deste trabalho teremos uma "caixa preta" portátil que substituirá o multiplexador e a ponte RLC do laboratório e terá precisão para diversas aplicações.

## 1.2 Objetivo

O objetivo geral deste trabalho é projetar, desenvolver e testar um sistema capaz de ler os sensores capacitivos de uma língua eletrônica. O equipamento deve ser concebido com pequenas dimensões para facilitar a portabilidade. Pretende-se, ao final do projeto, ter um sistema relativamente barato que possa substituir, para diversas aplicações, os equipamentos de laboratório utilizados atualmente para tais medidas.

### 1.2.1 Objetivo específico

O "MEMSCLE" será um sistema com as seguintes características:

- Medir uma Língua Eletrônica com 10 sensores independentes através da multiplexação de tais sensores;
- Medir capacitâncias de 100pF à 1mF;
- Medir as capacitâncias através de varredura em frequência e em frequência fixa dentro da faixa de 1Hz a 200kHz;
- A tensão máxima sobre os sensores a serem medidos será alternada e com valor de pico máximo em 0,5V;
- A interface de comunicação será via Ethernet;
- Medida indireta da capacitância através do método dos fasores, conhecendo amplitude de 3 tensões.
- O custo do sistema deve ser abaixo de R\$1000.

### 1.3 Justificativas do Trabalho

O equipamento a ser construído terá a versatilidade de levar a utilização da língua eletrônica para locais exteriores ao laboratório com boa facilidade. A interface Ethernet possibilitaria, por exemplo, que o sistema ficasse em uma estação no meio de um rio, alimentado por células solares e se comunicando com um PC na margem através da Ethernet. Desta forma seria possível medir características da água de um rio em qualquer ponto que se desejar. Somando tais facilidades ao custo barato do sistema, se torna interessante e viável a construção de um equipamento como o pretendido por este trabalho.

### 1.4 Estruturas do trabalho

Abaixo uma descrição capítulo a capítulo do que consta este trabalho.

Capítulo 2, Referencial Teórico: Aqui as referências utilizadas no trabalho serão discutidas. Citações de trabalhos já realizados e o porquê tais trabalhos se encaixam no universo deste trabalho podem ser encontrados neste capítulo;

Capítulo 3, Metodologia: Nesta parte do trabalho são discutidas as soluções escolhidas para o sistema bem como as justificativas das mesmas. Também são encontrados os diagramas de blocos do projeto e a análise de custos;

Capítulo 4, Resultados e Discussões: Resultados encontrados através do sistema MEMSCLE e comparação com outro sistema;

Capítulo 5, Conclusões: A conclusão do trabalho e as aplicabilidades alcançadas serão apresentadas e discutidas.

## 2. REFERENCIAL TEÓRICO

O referencial teórico será dividido em algumas partes chaves. Primeiramente será discutido o que é a língua eletrônica e a necessidade de multisensoriamento para tornar possível a construção da mesma. Depois serão analisadas as possibilidades de circuitos eletrônicos e análises matemáticas para a construção do sistema de medição.

### 2.1 Uma breve descrição da língua eletrônica

A língua eletrônica se baseia no reconhecimento de sabores e não na identificação em separado de cada substância química, nisto consiste a aplicação da seletividade global, classificar várias substâncias em grupos de sensações gustativas primárias e com o conjunto delas fazer a identificação [1]

Existem basicamente quatro tipos de línguas eletrônicas na literatura. A utilizada neste projeto é baseada em sensores químicos não-específicos que são feitos por microeletrodos interdigitados de ouro recobertos com filmes ultrafinos de polímeros condutores. Esta língua utiliza medidas de impedância como método de operação. Como a característica que se destaca nos sensores é a capacitiva então a concepção do circuito deste sistema foi baseada nela.[2]

Uma língua é composta por diversos sensores diferentes entre si. Esse tipo de sistema apresenta boa sensibilidade e ausência de um eletrodo de referência. E através de métodos estatísticos de análise, como análise das componentes principais (PCA) e Redes Neurais artificiais, por exemplo, é possível identificá-los. [3]

A Figura 1 (a) mostra um esquema de como é o sensor utilizado na língua eletrônica. É possível ver os terminais depositados de ouro sobre um suporte de vidro. Ainda na Figura 1 (a) observa-se a parte dos eletrodos interdigitados que fazem o papel das placas do capacitor e a camada protetora que representa o polímero depositado e que eletricamente falando é o dielétrico do modelo capacitivo do sensor.





Figura 1: Sensor da língua eletrônica: (a) Estrutura do sensor destacando as camadas que o compõem. (b) fotografia dos microeletrodos interdigitados e dos terminais. *Adaptado da referência [2]*

A Figura 1(b) mostra uma fotografia do sensor que está representado na Figura 1 (a). É possível observar nesta figura todos os itens descritos no parágrafo anterior.

## 2.2 Aplicações da língua eletrônica

A literatura mostra várias aplicações da língua eletrônica a base de polímeros condutores. Uma língua eletrônica composta pela combinação de filmes poliméricos condutores possui alta sensibilidade e reprodutibilidade e é capaz de distinguir os quatro tipos básicos de paladar (em alguns casos abaixo do limite de detecção humano), de detectar baixas concentrações de metais pesados em água ultra pura, de distinguir bebidas de mesmo paladar e também de detectar a supressão de paladares.[1]

Ainda na literatura existem resultados de experimentos realizados utilizando unidades sensoriais compostas por filmes finos feitos a partir de misturas de polímeros condutores com membranas lipídicas, imitando a língua humana, a qual se mostrou eficiente para a detecção dos padrões básicos de paladares e nos efeitos de supressão de paladares, sendo também estudada na análise de bebidas comerciais, separando diferentes tipos, dentre elas o café, água mineral, água destilada, cana de açúcar e água de coco. Existem também aplicações para a análise e diferenciação de vinhos. [4]

Nessas diferentes aplicações, a composição dos filmes finos é diferente, dependendo da composição, a sensibilidade a algumas substâncias pode aumentar e diminuir em relação a outras. [4]

### 2.3 Multisensoriamento para língua eletrônica

A Língua Eletrônica pode ser considerada um dispositivo multisensor automático dedicado a análise de amostras de composição complicada e para o reconhecimento de suas propriedades características. Recentemente, o número de publicações que cobre este tópico aumentou significativamente. Muitas arquiteturas possíveis de tais dispositivos foram propostas: potenciométricas, voltamétricas, bem como abordagens abrangentes incluindo sensores de massa e ópticos. Para a análise dos dados da matriz de sensores vários sistemas de reconhecimento foram propostos. [1]

A informação sobre a mudança do ambiente é recebida pelos organismos vivos através dos sentidos que são sistemas especializados e complicados. Eles são capazes de perceber e analisar os estímulos, e então reagir em relação a eles. [2]

A classificação dos sentidos assumidos pelos neurofisiologistas abrange cinco sentidos presentes em seres humanos: visão, audição, tato, paladar e olfato. As três primeiras são chamados sentidos físicos, porque eles estão adaptados para detectar estímulos físicos. Os dois últimos são sentido que podem perceber alterações na composição química do meio ambiente.[1]

Recentemente, tem havido muitas tentativas para simular os nossos sensores naturais de identificação química. A tarefa de construir um nariz eletrônico ou língua eletrônica é muito complicada, uma vez que os mecanismos da percepção química natural estão ainda sendo trabalhados. A meta principal é sentir milhares de compostos químicos.[5]

Embora a relação entre estrutura molecular e estímulos percebidos não é simples, pois moléculas que têm uma forma semelhante podem causar várias impressões (por exemplo, parte de hidratos de carbono é doce e parte deles não), enquanto aqueles com

grupos funcionais diferentes podem ser percebidos de forma semelhante (por exemplo, parte de hidratos de carbono é doce, e assim é o acetato de chumbo). Além disso, vários efeitos adicionais podem ocorrer: sinérgica, mascaramento, antagônica, compensatória, e por aí vai.[2]

Recentemente até o modelo de gostos básicos foi discutido e agora é considerado como sendo simplificado demais. No caso do cheiro, os odores básicos não são ainda conhecidos, e provavelmente não existem.[6]

Durante os últimos anos vários trabalhos na área foram publicados. Em alguns destes trabalhos foi utilizado um matriz de sensores químicos para simular receptores e métodos de reconhecimento de padrões para identificar os estímulos. Foi neste contexto que se percebeu que tal dispositivo poderia ser considerado como um nariz eletrônico ou língua eletrônica.[2]

Um desafio para o desenvolvimento de novos métodos analíticos e instrumentos consiste na busca da seletividade mais alta possível com respeito a um dado componente ou componentes. Para atender a essa meta, um grande número de sensores eletroquímicos, bem como muitos sistemas do tipo língua eletrônica, são baseados no uso de ISE ou sensores de natureza similar. É muito complicado, senão impossível, fazer um sistema com um único sensor, o motivo disto é a alta seletividade que precisamos para tornar a língua eletrônica viável. Este é o principal motivo que praticamente todos os trabalhos nesta área utilizando uma rede de sensores relativamente grande para poder caracterizar o ambiente.[7]

Dois campos de conhecimento tiveram a maior contribuição em relação ao desenvolvimento de sistemas multisensoriais. São eles, as novas conquistas da biologia, ou seja, a compreensão da estrutura e funcionamento do sistema humano sensorial, e, as novas possibilidades e abordagens para o tratamento de dados multidimensionais que foram propostas no âmbito da química.[1]

O desenvolvimento de sistemas químicos multisensoriais foi em grande parte uma tentativa para imitar a estrutura e as propriedades dos sistemas biológicos, a saber, o sistema de olfato e paladar dos. Inicialmente, essa abordagem foi utilizada no conceito de narizes eletrônicos e, mais tarde, princípios semelhantes foram utilizados para o desenvolvimento de analisadores de líquidos, as famosas línguas eletrônicas. No curso da evolução, o olfato tomou-se o mais eficiente sistema sensorial devido à alta sensibilidade do nariz humano com respeito a certas substâncias (pode-se identificar até



10 milhares de substâncias, no montante de até 40 moléculas). O limite de detecção do nariz humano com respeito à certas substâncias são várias ppb(parte por bilhão). Tais possibilidades de olfato são atribuídos ao grande número de tipos diferentes de receptores seletivos com uma combinação específica de processamento de sinais no sistema nervoso periférico e central. O sentido do gosto está organizado da mesma maneira como o olfato, embora o número de receptores de sabor em seres humanos é menor. Tradicionalmente existem as sensações gustativas básicas: doce, salgado, azedo e amargo.[6,8]

Recentemente, um gosto característico adicional foi introduzido, o qual é determinado, em particular pelo glutamato de sódio. As cinco principais características de sabor e sua combinação de ações são utilizadas para descrever e avaliar quantitativamente o sabor dos gêneros alimentícios invocando características como a nitidez do sabor, a duração do sabor, a textura dos alimentos.[5]

Atualmente, a relação entre a estrutura (e as propriedades químicas) de muitas substâncias e seu sabor não foi suficientemente estudada, bem como muitos outros efeitos interessantes, por exemplo, o de interação das substâncias com diferentes tipos de gosto ou o aumento no limite de sensibilidade (dessensibilizadora) devido ao efeito sinérgico das substâncias com diferentes tipos de sabores. O limite de detecção é geralmente mais elevado para o gosto, em comparação com aromas. No entanto, a diferença sensorial biológica entre o paladar e o olfato é comparável. Assim, os mecanismos de funcionamento do sistema de gostos e do olfato são semelhantes, embora o olfato é um sentido mais sofisticado na natureza.[1,2]

## **2.4 Pontes RLC (Resistor, Indutor e Capacitor)**

Existem inúmeros livros na literatura de Engenharia Elétrica que explicam o que é uma ponte RLC. O objetivo deste tópico é apenas discutir a aplicação de pontes RLC e circuitos RLC na medida de impedância de um dispositivo sobre teste.

Um medidor de RLC (resistência (R), indutância (L) e capacitância (C)) é um equipamento eletrônico utilizado para medir a indutância, capacitância e resistência de um

componente. Nas versões mais simples deste instrumento os verdadeiros valores dessas quantidades não são medidos, mas sim a impedância é internamente medida e convertido para o *display* como o respectivo valor da capacitância ou indutância. As leituras serão razoavelmente precisas se o capacitor ou o indutor em teste não tiverem uma componente resistiva significativa de impedância, como é o caso da língua eletrônica. Equipamentos mais avançados medem o real valor da indutância ou capacitância e ainda a componente resistiva do sistema. [9]

Geralmente, o dispositivo sob teste (DUT, do inglês *Device Under Test*) é submetido a uma fonte de tensão CA. O medidor mede a tensão e a corrente através do DUT. A partir desta razão o medidor pode determinar a magnitude da impedância. O ângulo de fase entre a tensão e a corrente também é medido em instrumentos mais avançados. Com a combinação entre impedância, a capacitância equivalente ou indutância e a resistência do DUT pode ser calculada e mostrada no *display*. O medidor pode assumir qualquer modelo para o cálculo, o modelo em série ou em paralelo para estes dois elementos. A hipótese mais útil, e que é geralmente adotada, é que as medições LR têm os elementos em série (como seria encontrado em uma bobina indutora) e que as medições de CR têm os elementos em paralelo (como seria encontrada na medição de um capacitor).[10]

Capacitância, indutância, resistência e fator de dissipação também podem ser medidos por vários circuitos de ponte. Eles envolvem ajustar elementos calibrados até que o sinal a um detector se torne nulo, em vez de medir a impedância e o ângulo de fase. Este projeto utilizará este conceito para o método de ponte de Wheatstone Capacitiva.[11]

## 2.5 Multiplexação

Assim como para pontes e circuitos RLC existem vários livros textos que explicam o que é um multiplexador e uma multiplexação. O objetivo deste tópico é apenas fazer uma resumo sobre este assunto.

Um multiplexador, *multiplexer*, *mux* ou *multiplex* é um dispositivo que codifica as informações de duas ou mais fontes de dados num único canal. São utilizados em situações onde o custo de implementação de canais separados para cada fonte de dados é maior que o custo e a inconveniência de utilizar as funções de multiplexação/demultiplexação. Em eletrônica, o multiplexador combina um conjunto de sinais elétricos num único sinal elétrico. Existem diferentes tipos de multiplexadores para circuitos analógicos e digitais.[12]

A Figura 2 mostra o funcionamento interno teórico de um multiplexador analógico com 4 entradas e uma saída. Nela é possível perceber que para cada entrada existem dois potenciais relacionados, S1A e S1B, S2A e S2B e assim por diante. O multiplexador chaveia estes potenciais na entrada para a saída DA e DB. [13]

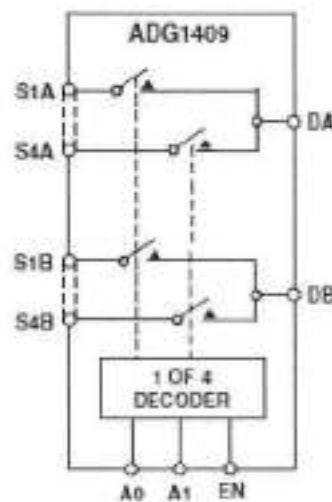


Figura 2: Diagrama funcional de um multiplexador analógico com 4 entradas e 1 saída. *Extraído da referência [13]*

No caso do sistema a ser projetado e desenvolvido neste trabalho é necessária a utilização de um multiplexador analógico, uma vez que o sinal enviado ao dispositivo sob teste deve ser uma senoide com média nula.

## 2.6 Ponte de Wheatstone Capacitiva

Como já explicado nos itens anteriores, quando a área de conhecimento a ser discutida já é amplamente divulgada será descrito apenas um resumo sobre o tópico.

Pontes são alguns dos dispositivos mais precisos para medir capacitância, impedância, resistência. Para o nosso propósito, ou seja, para medir Capacitância, utilizar a ponte de Sauty ou Schering é melhor. Esses circuitos são baseados no princípio da ponte de Wheatstone que têm dois braços, um dos quais tem o parâmetro desconhecido. Ao se equilibrar a ponte o detector de nulo marca zero, sendo assim podemos calcular o valor da capacitância desejada através das outras impedâncias conhecidas que temos no circuito.[11]

A ponte de Sauty é uma implementação direta da ponte de Wheatstone com a fonte de CC sendo substituída por uma fonte de corrente alternada. O detector de nulo a ser usando também tem um amplificador onde o ganho pode ser ajustado. A ponte em questão é mostrada na Figura 3.[11]

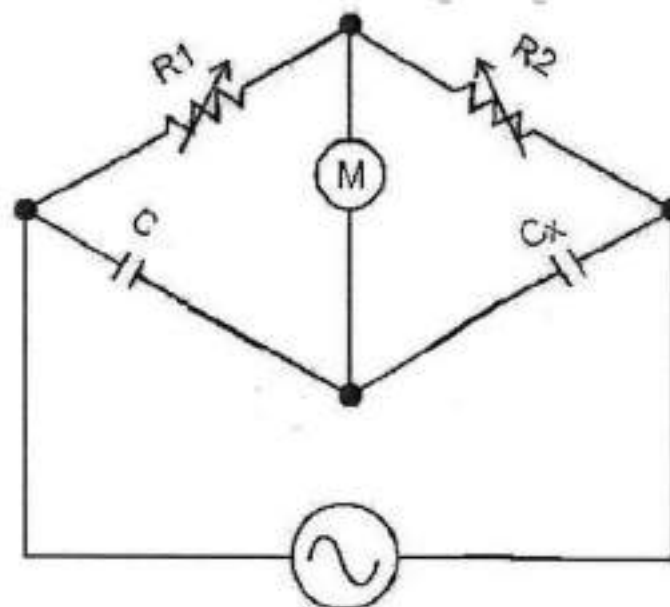


Figura 3: Circuito esquemático da Ponte de Wheatstone AC. *Extraído da referência [14]*

É possível utilizar o princípio da ponte de Wheatstone para calcular a capacitância  $C_x$  desconhecida como[14]:

$$C_x = \frac{R_1 C}{R_2}$$

Uma Ponte de Schering é uma ponte de corrente alternada usada para medir capacitância e fator de dissipação. O balanço da ponte é independente da frequência. É uma melhoria em relação a Ponte de Sauty pois permite uma medida mais precisa de capacitâncias muito pequenas devido ao elemento do capacitor paralelo o que permite melhorar a sensibilidade do balanceamento da ponte. Porém a ponte leva mais tempo para ser balanceada. O desenho da ponte é mostrado na Figura 4.[14]

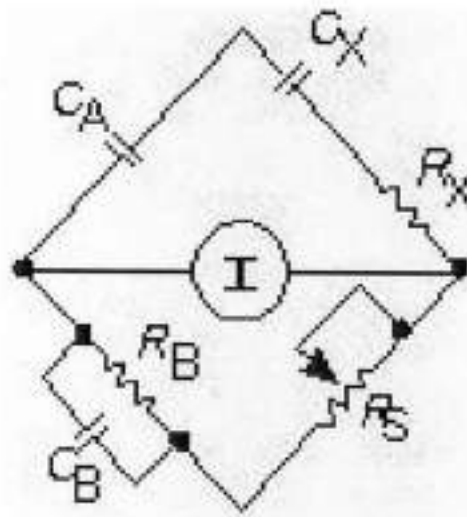


Figura 4: Circuito esquemático da Ponte de Schering. Adaptado da referência [14]

## 2.7 Método sensor de fase

Como um capacitor altera a fase de um sinal é possível, através de medida de tensão e corrente medir tal defasagem e calcular a partir daí a capacitância de uma DUT.[15]

A Figura 5 mostra uma sugestão de diagrama de blocos para a medida da capacitância. [15]

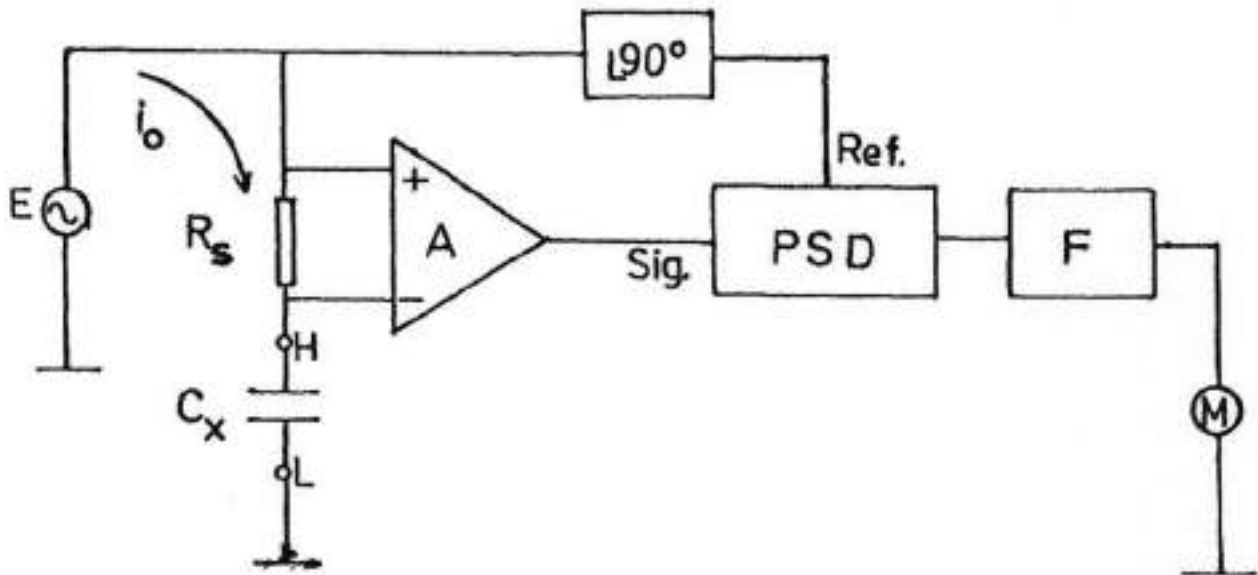


Figura 5: Esquema para medir a diferença de fase causada por um DUT. Extraído da referência [15]

Nesta técnica um capacitor a ser testado  $C_x$  é conectado em série com um resistor fixo  $R_s$ . A fonte tem onda senoidal de amplitude constante e frequência constante. Através de um circuito diferenciador é possível comparar a fase da corrente medida indiretamente através da tensão no resistor (no resistor ideal temos tensão e corrente sincronizadas) com a fase do sinal senoidal gerado pela fonte. O sinal senoidal é filtrado por um filtro passa baixa para um nível DC proporcional a diferença de fase entre a tensão e a corrente no circuito.[15]

Após o sistema entrar em regime permanente, utilizando a técnica de fasores e considerando a fase de  $E$  como referência, podemos calcular a amplitude da tensão senoidal sobre o resistor  $R_s$ . [15]

$$V_{rs} = \frac{j \cdot E \cdot R_s \cdot \omega \cdot C_x}{j \cdot R_s \cdot \omega \cdot C_x + 1}$$

Na equação acima  $w$  é a frequência angular e  $j$  a unidade imaginária. Na equação acima as variáveis desconhecidas são  $Cx$  e a fase do fasor. Uma vez que o nosso sistema é capaz de medir a fase entre  $E$  e  $Vrs$  então podemos calcular a capacitância sendo medida.

## 2.8 Método filtro passa baixa

Segundo a teoria de circuitos elétricos [9], um filtro passa baixa é amplamente conhecido pela literatura tendo como circuito equivalente o apresentado pela Figura 6:

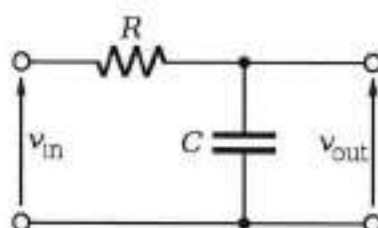


Figura 6: Circuito equivalente de um filtro passa baixa. *Extraído da referência [16]*

A função de transferência no domínio da frequência pode ser escrita como:

$$\frac{V_{out}}{V_{in}} = \frac{1}{1 + s \cdot R \cdot Cx}$$

Os gráficos de amplitude e fase de um filtro passa baixa de primeira ordem traçados a partir de uma varredura em frequência podem ser vistos no Gráfico 1 abaixo.



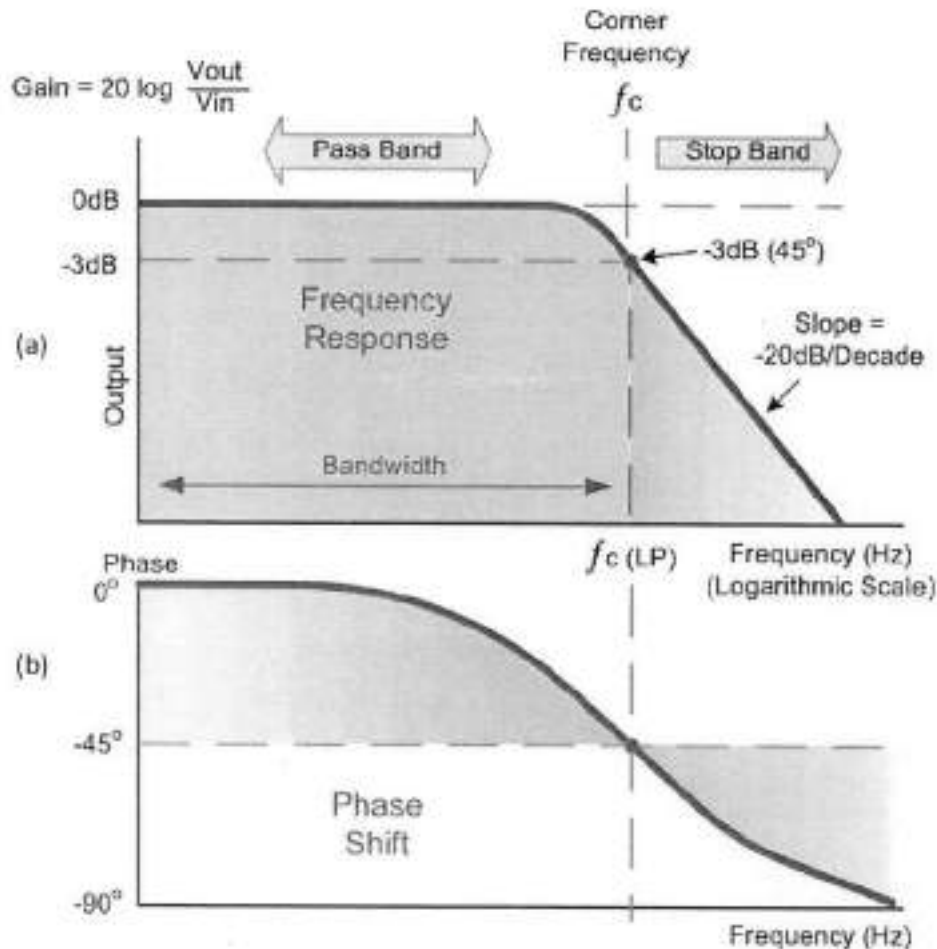


Gráfico 1: Resposta em frequência de um filtro passa baixa de primeira ordem: (a) ganho de tensão em dB em função da frequência; (b) deslocamento de fase em função da frequência. *Extraído da referência [17]*

Tendo isso em mente e a possibilidade do sistema descrito neste trabalho fazer uma varredura em frequência, o método idealizado e desenvolvido neste projeto e aqui intitulado método filtro passa baixa consiste na varredura em frequência através de um circuito integrado com a função de gerar sinais senoidais em variadas frequências e um conversor analógico – digital de alta precisão.

O circuito descrito por este método varia a resistência  $R$  do filtro passa baixa de primeira ordem de modo a alterar as características do filtro a fim de que a frequência de corte do filtro fique dentro do range da varredura em frequência do sistema, determinando a frequência de corte é possível calcular a capacitância.

Ainda utilizando a teoria de circuitos elétricos [9] para o cálculo da capacitância utilizando uma frequência fixa uma das formas é conhecer a tensão de entrada, a tensão sobre o resistor e a tensão sobre o capacitor. Conhecendo esses sinais e modelando o



sensor como um capacitor e um resistor no modelo paralelo (Figura 7) também podemos encontrar a capacitância e a resistência do sensor através de método algébrico.

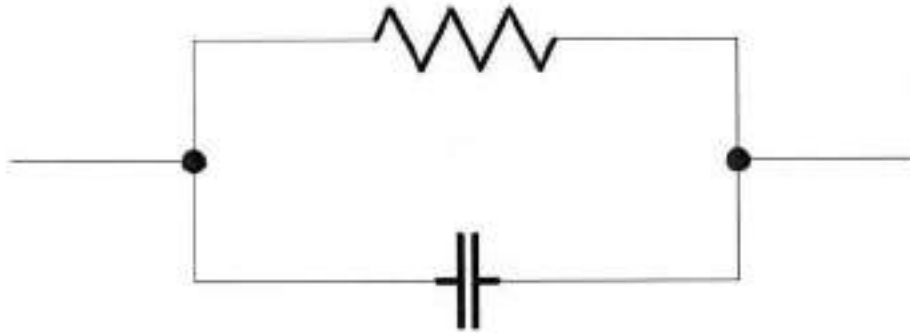


Figura 7: Modelo equivalente de um capacitor real. Desenhado no Altium PCAD 2006

Observando o triângulo da Figura 8, onde a tensão do gerador ( $V_g$ ) é a resultante da soma vetorial das tensões no resistor  $R$  da Figura 6 mais a tensão no capacitor  $C$  da Figura 6 modelado como o circuito da Figura 7, podemos calcular o ângulo Theta " $\theta$ " e por fim determinar a capacitância e a resistência do sensor. Abaixo a análise matemática que torna isso possível (lei dos cossenos).

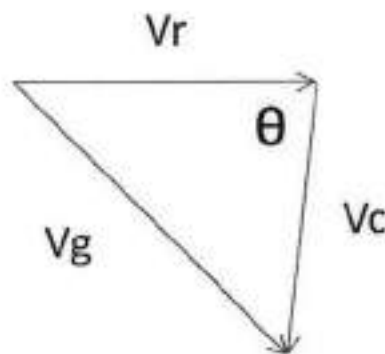


Figura 8: Soma vetorial para a determinação da defasagem por método algébrico. Desenhado no Microsoft Power Point 2010

$$\cos(\theta) = \frac{V_r^2 + V_c^2 - V_g^2}{2 * V_r * V_c}$$

Após determinar o cosseno do ângulo Theta podemos calcular o ângulo através da função arco cosseno e considerando que o ângulo está no segundo quadrante do círculo trigonométrico (Figura 9). Dessa forma temos o seguinte gráfico no plano complexo.

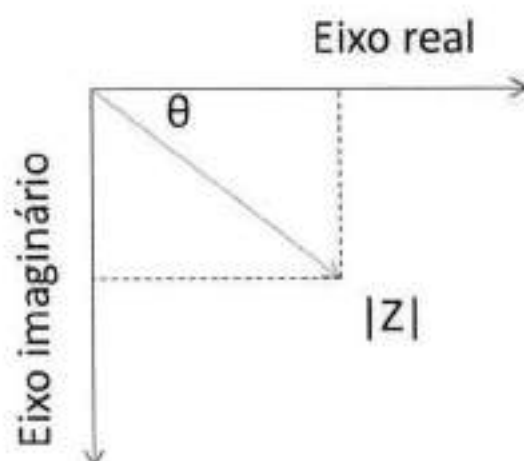


Figura 9: Eixo do modelo equivalente do capacitor. Desenhado no Microsoft Power Point 2010.

É possível calcular o módulo da impedância Z mostrado na Figura 9 da seguinte forma:

$$|Z| = \frac{V_c}{V_r} * R$$

Onde o R em questão é o mesmo da Figura 6. Logo é possível calcular a resistência parasitária (ou característica dependendo da aplicação) e a capacitância através das seguintes equações:

$$R_c = \sqrt{|Z|^2 * (1 + \tan^2(\theta))}$$

$$C = \frac{\tan(\theta)}{R_c * w}$$

No caso das equações acima,  $w$  é a frequência angular do sistema, que é conhecida a priori.

As fórmulas apresentadas acima foram deduzidas da análise de circuitos RC e podem ser encontradas em qualquer livro de engenharia elétrica, um deles é a referência [9].

## 2.9 Métodos de medida ponte RLC HP4286

O equipamento RLC meter da HP modelo 4286 utiliza o circuito teórico apresentado na Figura 10 para a medida da impedância [18]:

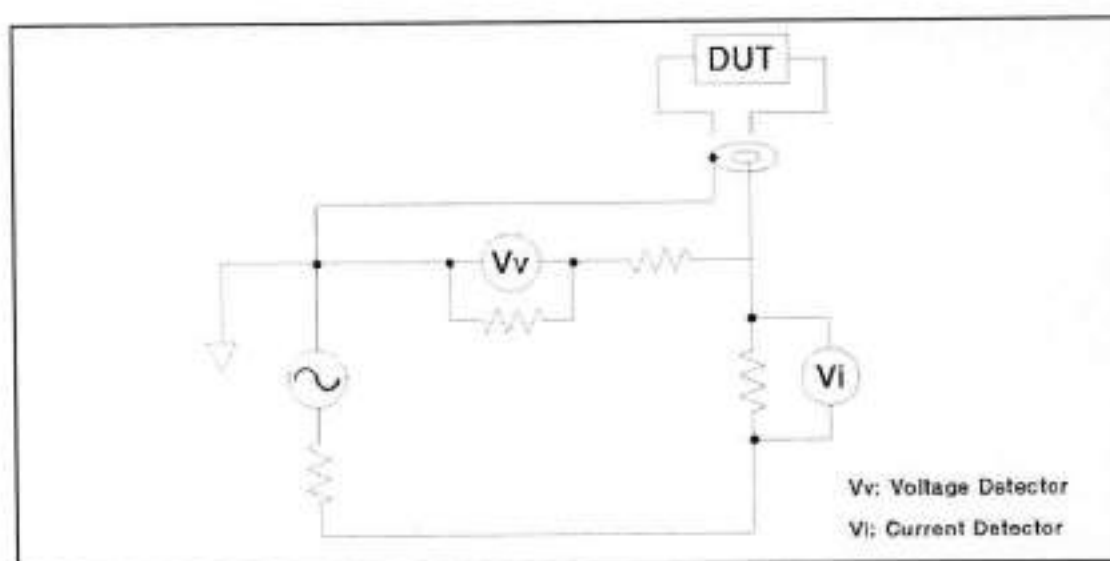


Figura 10: Circuito teórico do RLC meter da HP, Extraído da referência [18]

O circuito apresentado na figura 11 pode ser encontrado no manual de serviço do equipamento na página 11-23.

Sabendo a frequência e amplitude do sinal de entrada, bem como a tensão medida por  $V_v$  e a corrente medida por  $V_i$  podemos determinar o módulo da impedância do DUT.

Neste caso pode também ser utilizado um método trigonométrico parecido com o utilizado no item anterior.

### 3. METODOLOGIA

Baseado no referencial teórico foi previsto no hardware dois circuitos analógicos e dois métodos para medir a capacitância. O primeiro circuito utiliza o método do filtro passa-baixa a partir da análise trigonométrica em frequência constante ou através da determinação da frequência de corte como mostrado no item 2.8. O segundo circuito previsto no *hardware* utiliza o circuito teórico do equipamento HP4286 para o cálculo da capacitância. O método da ponte de Wheatstone, descrito anteriormente neste trabalho, seria o terceiro método de medida da capacitância, mas o mesmo mostrou-se muito complicado de ser implementado devido a falta no mercado de capacitores variáveis precisos e na gama de valores necessários para equilibrar a ponte.

Resumindo, no *hardware* do sistema foram previstos dois circuitos analógicos para a medida da capacitância. No *software* do sistema existe a possibilidade de seleção de circuito analógico para medir a capacitância sob teste.

#### 3.1 Descrição de componentes críticos

Por ser um circuito relativamente barato e preciso alguns dos principais componentes têm de ser específicos.

Como coração do sistema foi selecionado o microcontrolador da Texas LM3S6965, cujas principais características estão listadas abaixo, extraídas do próprio *datasheet* do componente [19]. Este componente foi selecionado devido a arquitetura 32bits de seu núcleo, por conter os *Layers Ethernet* MAC e PHY e por ter velocidade suficiente para adquirir os dados de um conversor AD externo de alta precisão, a ser discutido posteriormente.

Principais características do LM3S6965:

- Arquitetura 32-bit ARM Cortex-M3 v7M otimizada para aplicações embarcadas;
- Operação a 50MHz;

- 1.25DMIPS/MHz;
- 42 GPIOs, dependendo da configuração;
- Dois módulos I<sup>2</sup>C;
- Controlador de *Ethernet* 10/100;
- Conversor AD de bits e MS/s.

Para otimizar o trabalho será utilizado a placa de desenvolvimento da Texas Stellaris EKI-LM3S6965 que contem o microcontrolador selecionado, um display OLED, interface de programação *on-circuit* e interface *Ethernet*. Com a placa vem também todo o kit necessário à programação do chip. Na Figura 11 observamos uma foto da placa que será utilizada.



Figura 11: Placa da Texas a ser utilizada

Os outros dois componentes críticos a serem discutidos são o conversor AD externo e o gerador de sinais. Além do conversor AD interno do microcontrolador será previsto também um conversor AD externo e preciso. O conversor AD (ADS1675) externo

a ser utilizado também é da Texas Instrumente as principais características, extraídas do *datasheet* [20], são:

- Velocidade máxima de amostragem de 4MSPS;
- Precisão de 24 bits;
- Performance AC de 103dB de Range Dinâmico a 4MSPS;
- Performance AC de 111dB de Range Dinâmico a 125kSPS;
- Filtro digital programável com seleção *Low-Latency* ou *Wide-Bandwidth*;
- Interface serial de leitura flexível com suporte ao padrão CMOS e ao padrão diferencial LVDS;
- Fácil controle de conversão através do pino START;
- Detecção de *Out-of-Range*;
- Alimentação até +5V analógico e +3V digital;
- Consumo máximo de 575mW;

O gerador de sinal é o componente AD5932 da Analog Device e tem suas características, extraídas da referência [21], descritas abaixo:

- Gerador de formas de onda programável;
- Sem necessidade de componentes externos;
- Frequência máxima de saída até 25MHz;
- Perfil de frequência programável para diminuir o número de escritas do microcontrolador / DSP;
- Saída de onda selecionável entre senoidal, triangular ou quadrada;
- Controle automático ou manual de *frequencystepping*;
- Modo *Power-down* de 20 $\mu$ A;
- Faixa de alimentação de 2,3V a 5,5V;
- Faixa de temperatura automotiva de -40°C a +125°C;

Não se pode esquecer que quando se trata de um sistema que vai multiplexar e ler sinais analógicos, os outros dois principais componentes são o multiplexador analógico e o amplificador operacional.

No multiplexador analógico os pontos mais importantes são o quanto é pequena a resistência interna de condução e o *crosstalk*. Hoje em dia existem chips muito bons que fazem isso, mas nem sempre é fácil encontra-los a bom preço no Brasil. Apesar das dificuldades no mercado brasileiro para importação de componentes específico, foi encontrado um dispositivo muito bom para fazer o papel de multiplexador dos sensores. Abaixo é possível encontrar as principais características do mesmo, extraídas de seu *datasheet* [13]. O componente é fabricado pela Analog Devices com o código ADG1409:

- 4.7R de resistência ativa máxima a 25°C;
- 0.5R de resistência *flatness*;
- 190mA de corrente contínua máxima de operação;
- Alimentação em  $\pm 15V$  /  $+12V$  /  $\pm 5V$ ;
- Entradas lógicas compatível com 3V;
- Operação *rail-to-rail*;

Para completar a nossa lista de componentes críticos vamos apresentar agora o amplificador operacional escolhido para as medidas diferenciais e identificação de nulo para a medida de fase. Nesse caso é necessário um componente que tenha pequeno *offset* e grande largura de banda. Não se pode também perder o foco no preço. O componente escolhido para essa missão foi o AD823 da Analog Devices. As principais características, retiradas do manual [22], são apresentadas abaixo:

- 16MHz *rail-to-rail* e entrada FET;
- Operação com fonte de alimentação dupla de  $\pm 1,5V$  a  $\pm 18V$ ;
- Corrente de saída de 15mA;
- Performance AC excelente da faixa de 2,6mA / amplificador;
- -3dB *bandwidth* a 16MHz;
- *Settling time* de 350ns;
- *Slew rate* de 22V /  $\mu s$



- Excelente performance DC;
- 800 $\mu$ V de *offset* de entrada máximo;
- 2 $\mu$ V / °C de tensão de *offset* de deriva;
- 25pA de corrente máxima de entrada;
- Baixa distorção;
- Baixo ruído.

Com isso encerramos a lista dos componentes mais críticos do sistema.

### 3.2 Diagrama de blocos

A seguir, na Figura 12, será apresentado o diagrama de blocos completo do sistema. É possível observar no lado esquerdo da figura a entrada de sensores. A cabeça com os sensores da língua eletrônica será conectada no sistema através de um conector padrão DB25. Os sensores serão multiplexados pelo circuito integrado ADG1409 explicado anteriormente neste trabalho. No diagrama da Figura 12 é possível observar o bloco multiplexador ao lado do bloco de entrada dos sensores.

Após a multiplexação o sinal proveniente dos sensores passará por um seletor analógico de modo que selecionará um dos dois modos analógicos descritos da seção metodologia, o modo "HP" ou o modo "filtro passa baixas".

O microcontrolador será o cérebro do sistema, controlando o gerador de sinais (circuito integrado AD5932) como pode ser visto na Figura 12 no bloco "Gerador de sinais". O sinal gerado será senoidal.

O sinal analógico passará pelos blocos analógicos ("modo filtro passa baixas" e "modo HP4286") e em determinados pontos haverá a medição de amplitude e fase de acordo com os blocos "medidos de tensão e fase" da figura.

Por fim os sinais analógicos são convertidos para digitais através de um conversor AD.



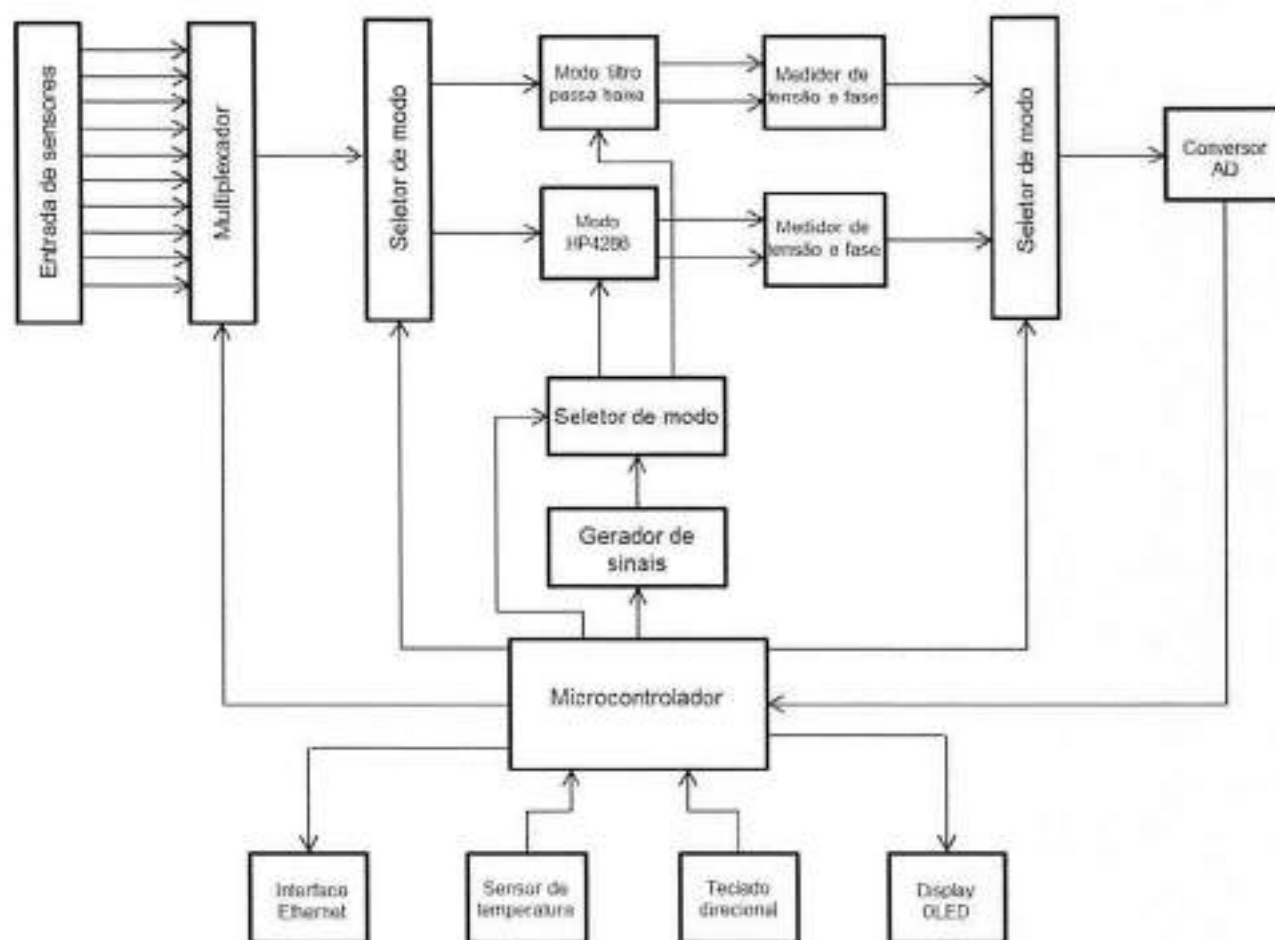


Figura 12: Diagrama de blocos do sistema

Além dos blocos já apresentados anteriormente, vemos no diagrama um sensor de temperatura, um teclado direcional e um display OLED. O sensor de temperatura é interno ao microcontrolador. O teclado direcional citado é o teclado já incluso na placa da Texas, com dois botões à direita e quatro a esquerda do display OLED citado.

### 3.3 Esquema Técnico do circuito

Nas próximas páginas será apresentado detalhes do esquema técnico do sistema. O projeto foi desenvolvido no software PCAD 2006 e os Gerbers foram impressos na placa de circuito impresso pela empresa Micropress. O esquema completo encontra-se anexo a este trabalho.

Na Figura 13 vemos o esquema da fonte de alimentação do sistema. É interessante ressaltar que a fonte é linear para não haver ruído de chaveamento proveniente da fonte. Na Figura 13 vemos três reguladores de tensão, o 7805 para regular a entrada para uma saída de +5V, o 7905 para regular a entrada para uma saída de -5V e o LM1117-3V3 para transformar os 5V da entrada para 3,3V na sua saída.

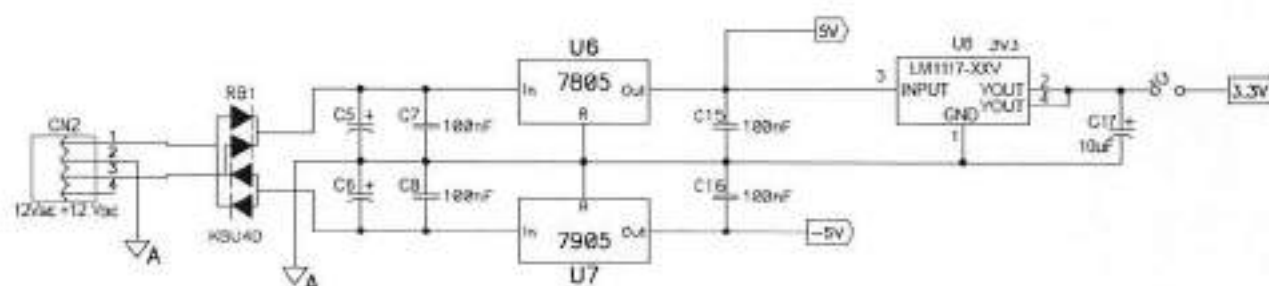


Figura 13: Circuito esquemático da fonte de alimentação do sistema

A Figura 14a seguir mostra os circuitos integrados multiplexadores como dispostos no hardware do sistema. O CI "U10" é um dos multiplexadores utilizados para a seleção de modo analógico do sistema como descrito no diagrama de blocos da Figura 12.

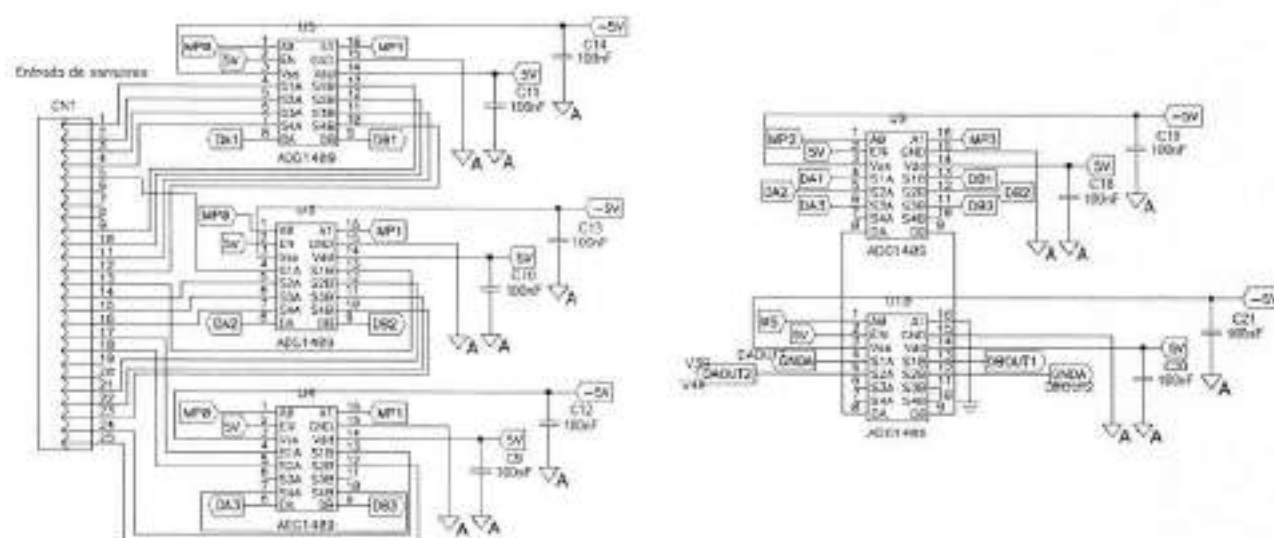


Figura 14: Circuito esquemático da multiplexação dos sensores

Na Figura 15 a seguir é possível observar um dos circuitos utilizados para a medida de tensão diferencial no sistema. Podemos observar à esquerda dois amplificadores na configuração seguidor de tensão. O amplificador central é um amplificador diferencial. O

bloco amplificador mais a direita do esquema é utilizado como amplificador somador a fim de incluir um nível DC no sinal de forma que possa ser lido pelo conversor AD, pois tal conversor não suporta níveis negativos de tensão.

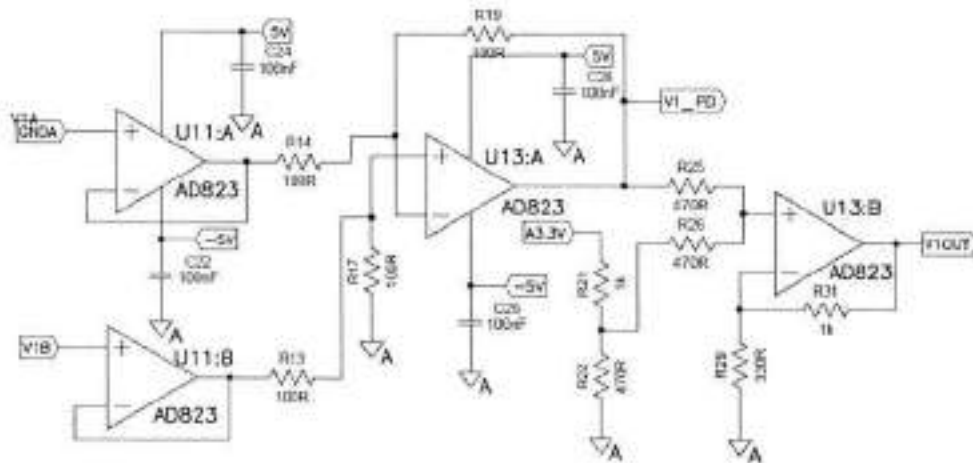


Figura 15: Circuito esquemático de amostragem e medição de tensão

A Figura 16 mostra um dos circuitos utilizados para medir a fase do sistema. O circuito foi desenhado baseado na referência [23]. No circuito da Figura 16 é possível ver dois blocos amplificadores utilizados como detectores de nulo, uma porta XOR para gerar um sinal quadrado de acordo com a diferença de fase detectada pelos geradores de nulo e por fim um filtro passa baixas para transformar os pulsos em nível DC.

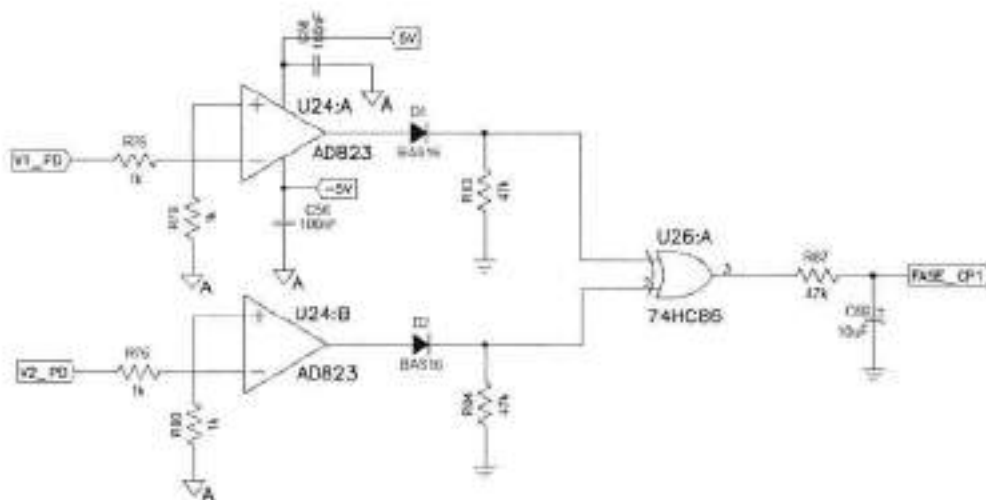


Figura 16: Circuito esquemático para identificação de fase

## **4. RESULTADOS E DISCUSSÕES**

O capítulo de resultados e discussões será dividido da seguinte maneira:

- Apresentação da Placa de Circuito Impresso (PCI) resultante deste trabalho;
- Apresentação do funcionamento dos principais circuitos durante o período de testes;
- Apresentação da interface gráfica;
- Resultados com capacitores comerciais;
- Resultados com a língua eletrônica;
- Análise de custo do projeto / produto.

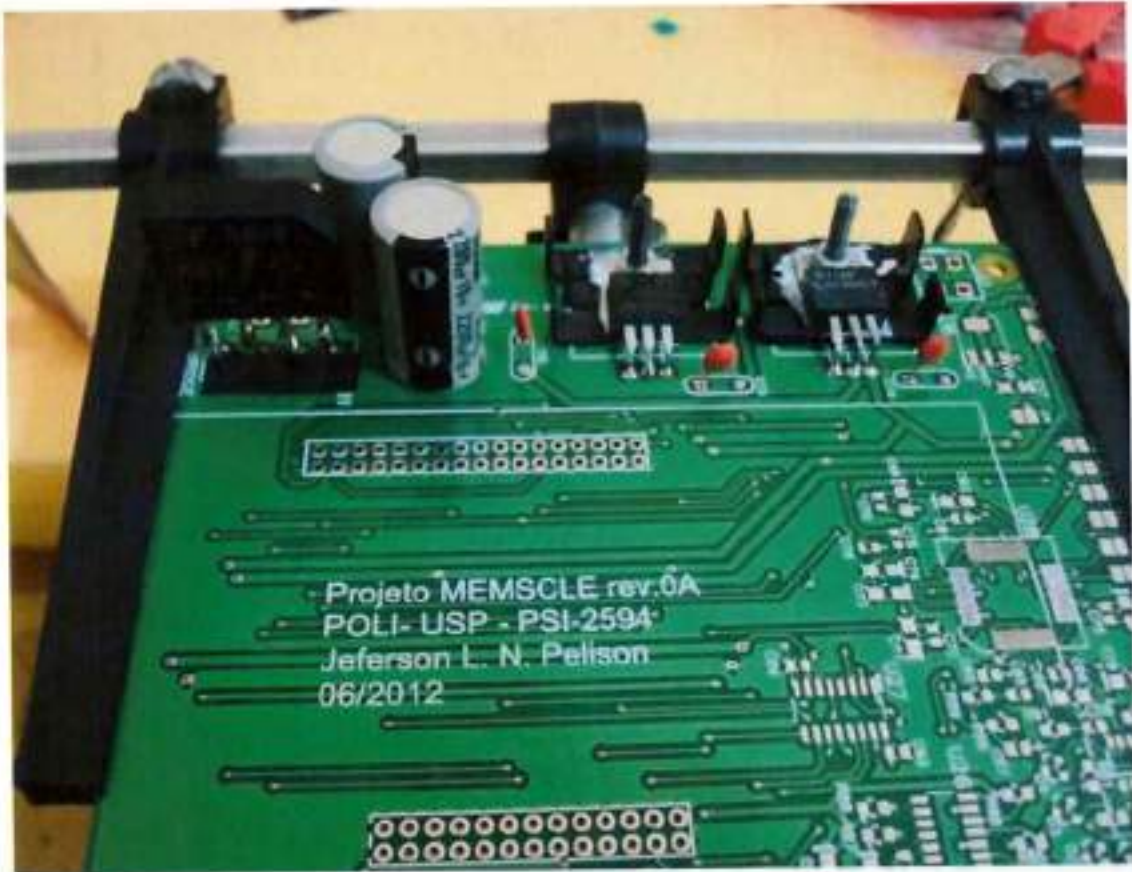
### **4.1 Placa de circuito impresso projetada e confeccionada**

A placa de circuito impresso foi projetada através do software "Altium PCAD 2006" e os arquivos Gerbers foram enviados à empresa Micropress para a confecção da placa. O projeto completo da placa de circuito impresso pode ser encontrado anexo a este trabalho.

A Figura 17 mostra fotos da placa quando estavam montados apenas alguns componentes do circuito de multiplexação e a fonte de alimentação.



(a)



(b)

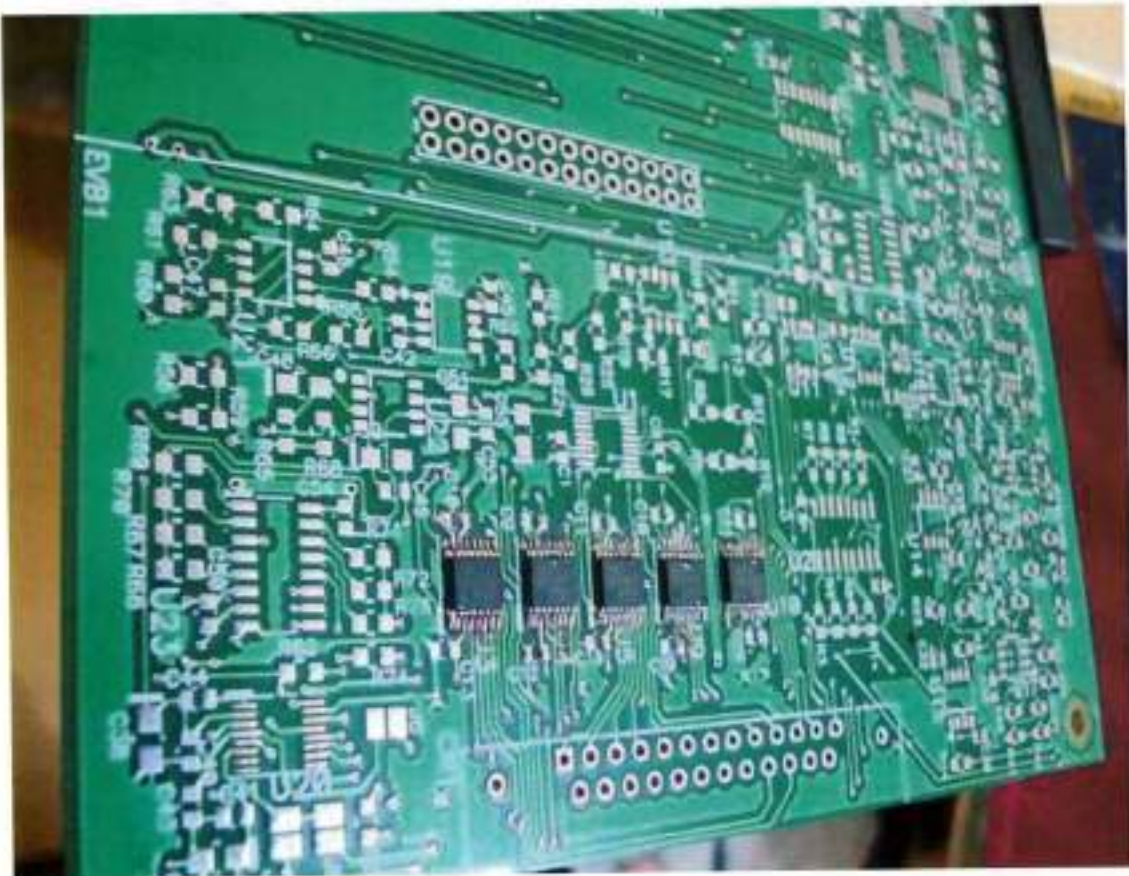


Figura 17: PCI sendo montada: (a) fonte de alimentação; (b) circuito de multiplexação analógica dos sensores

## 4.2 Circuitos em funcionamento

Para começar este item mostra-se abaixo na Figura 18 a placa que foi confeccionada após estar quase 100% montada.



Figura 18: Placa PCI quase 100% montada

### 4.2.1 Gerador de funções em funcionamento

O Gráfico 2 mostra o protocolo de comunicação entre o microcontrolador e o gerador de funções conforme descrito no manual do chip gerador de funções. A onda superior no gráfico é o *clock* necessário à comunicação e a onda inferior é o dado sendo enviado ao gerador de sinais.

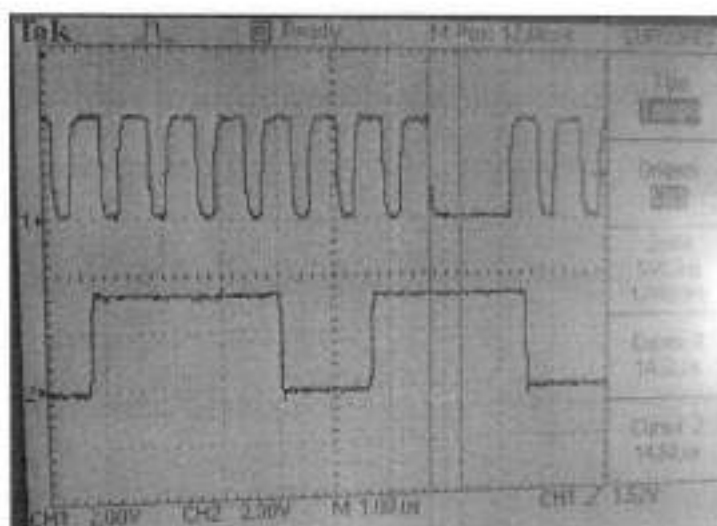


Gráfico 2: Gráficos da comunicação entre o microcontrolador e o Gerador de sinais

O Gráfico 3 mostra a esquerda a forma de onda que sai do gerador e à direita a forma de onda após passar por um filtro passa alta. A senoide simétrica, com média 0V e valor de pico aproximado em 0,5V é a onda enviada ao circuito analógico que contém o capacitor a ser medido. Pode-se observar que a condição de amplitude máxima a ser enviada para o sensor proposta por este projeto é respeitada. Na Figura 19 o circuito montado e com o gerador de funções funcionando.

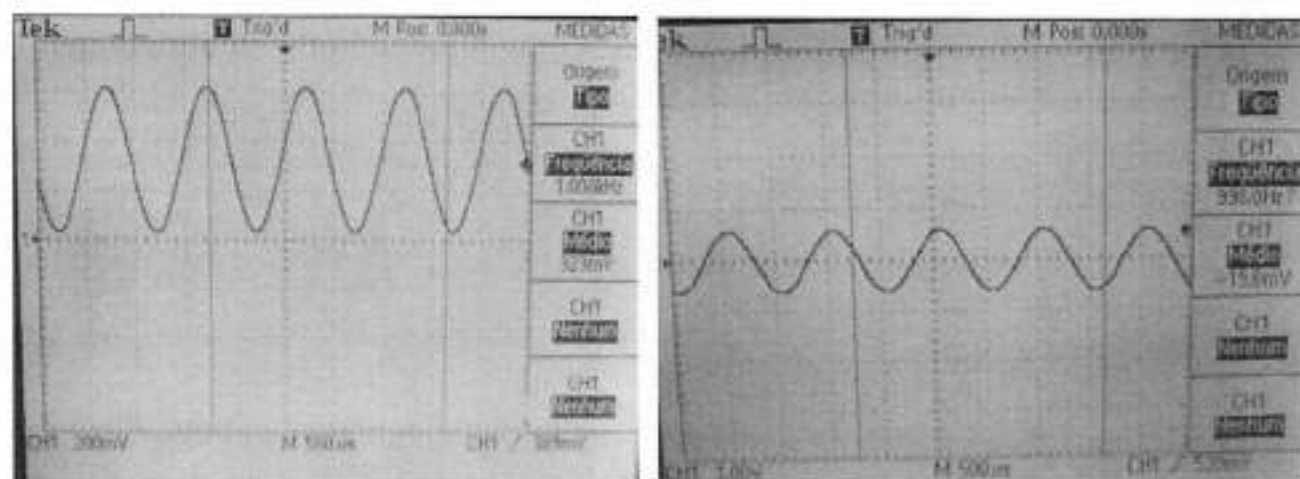


Gráfico 3: Funcionamento do circuito gerador





Figura 19: Circuito montado executando testes com o gerador de funções

#### 4.2.2 Circuitos medidores de tensão diferencial

A Figura 20 mostra o circuito funcionando e medindo as tensões do circuito contendo uma capacitância a ser medida. No Gráfico 4 é possível ver uma das tensões medidas no circuito através dos amplificadores diferenciais. O valor de pico a pico desta tensão, após as etapas de amplificação e o que efetivamente será amostrado pelo conversor AD é de 960mV como pode ser observado na tela do osciloscópio mostrado no Gráfico 4.





Figura 20: Circuitos amplificadores diferenciais em funcionamento

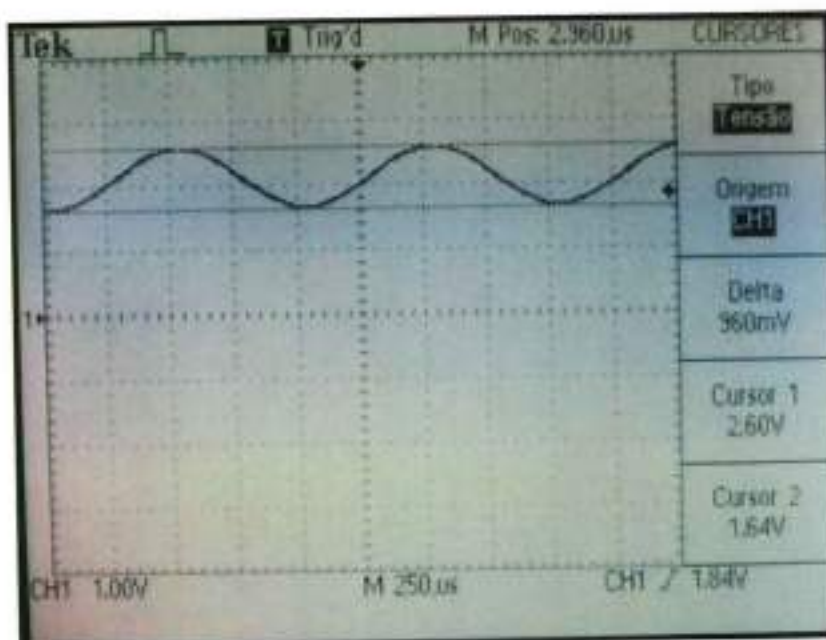


Gráfico 4: Medida de tensão através do circuito com amplificadores operacionais

#### 4.2.3 Projeto MEMSCLE completo e medindo a língua eletrônica

Após averiguar os circuitos um a um, chegou a hora de testar o funcionamento do circuito com a língua eletrônica. A Figura 21 mostra os equipamentos necessários para medir a língua eletrônica em laboratório, onde é possível observar no canto superior esquerdo o multiplexador utilizado atualmente, no canto superior direito o RLC *meter* e na parte debaixo da figura um *Desktop* executando o programa de controle dos equipamentos através de uma interface programada em LabView. A Figura 22 mostra a língua eletrônica conectada aos equipamentos da Figura 21 e sendo medida.

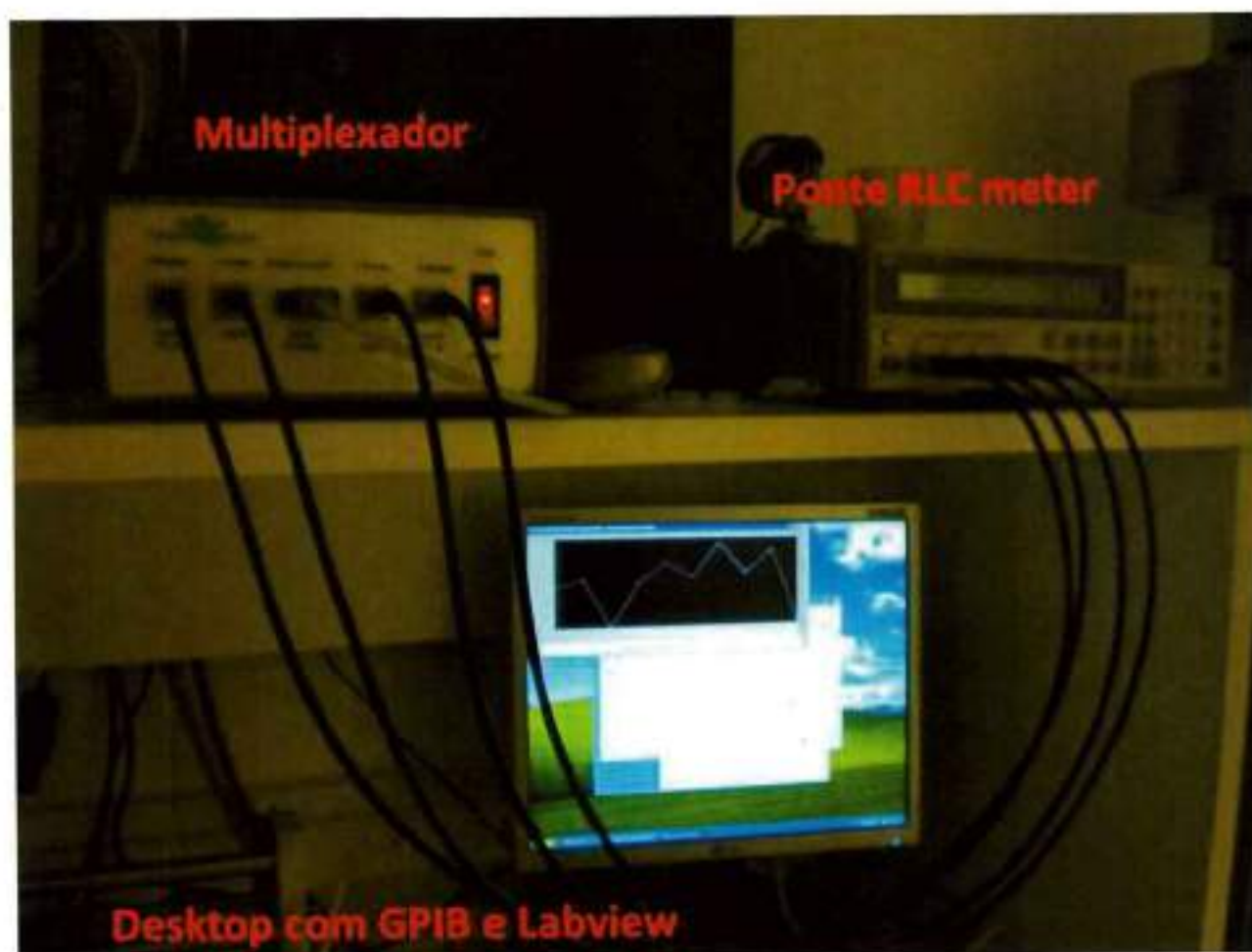


Figura 21: Equipamentos necessários para a medida da língua eletrônica atualmente



Figura 22: Língua eletrônica sendo medida através do equipamento atual

Agora chegou a hora de comparar os *Hardware*s com o projeto MEMSCLE medindo a língua eletrônica. A Figura 23 e a Figura 24 mostram as medidas da língua eletrônica sendo efetuadas com o projeto MEMSCLE. Na Figura 23 é possível observar o *hardware* do projeto MEMSCLE à esquerda, a língua eletrônica conectada no mesmo logo abaixo e um *Notebook* com interface de rede para acesso e controle do sistema.



Figura 23: Projeto MEMSCLE em funcionamento





Figura 24: Projeto MEMSCLE (esquerda) e Língua Eletrônica (direita)

### 4.3 Interface gráfica

O projeto MEMSCLE é controlado e lido através de uma interface web. Isso torna o projeto bem versátil, pois pode ser conectado diretamente à *internet* e ser controlado e lido de qualquer dispositivo com um *Browser* conectado a rede mundial de computadores.

Abaixo na Figura 25 a interface de apresentação do projeto MEMSCLE e na Figura 26 a interface de controle do sistema. As páginas visualizadas nestas duas figuras estão embarcadas no microcontrolador do sistema.

Na Figura 26 vemos na interface web alguns botões para controle. Os botões de "+" ou "-" selecionam o sensor a ser medido. O botão de "Atualizar Frequência" configura a frequência a ser utilizada para a leitura do sistema. O botão "Aciona Gerador" liga o gerador de funções com saída de onda senoidal. Por fim o botão "Ler sistema" executa a leitura do sensor selecionado na frequência selecionada.

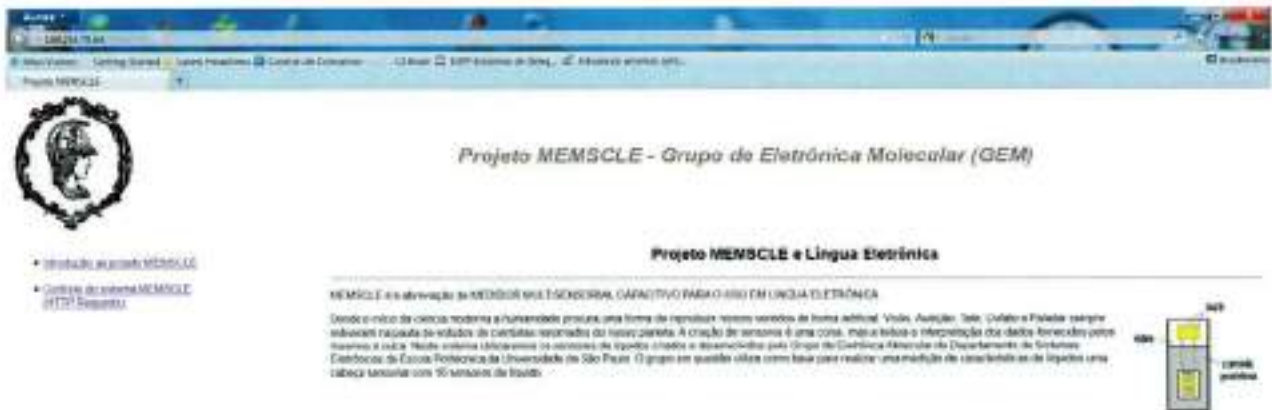


Figura 25: Página de apresentação do sistema MEMSCLE

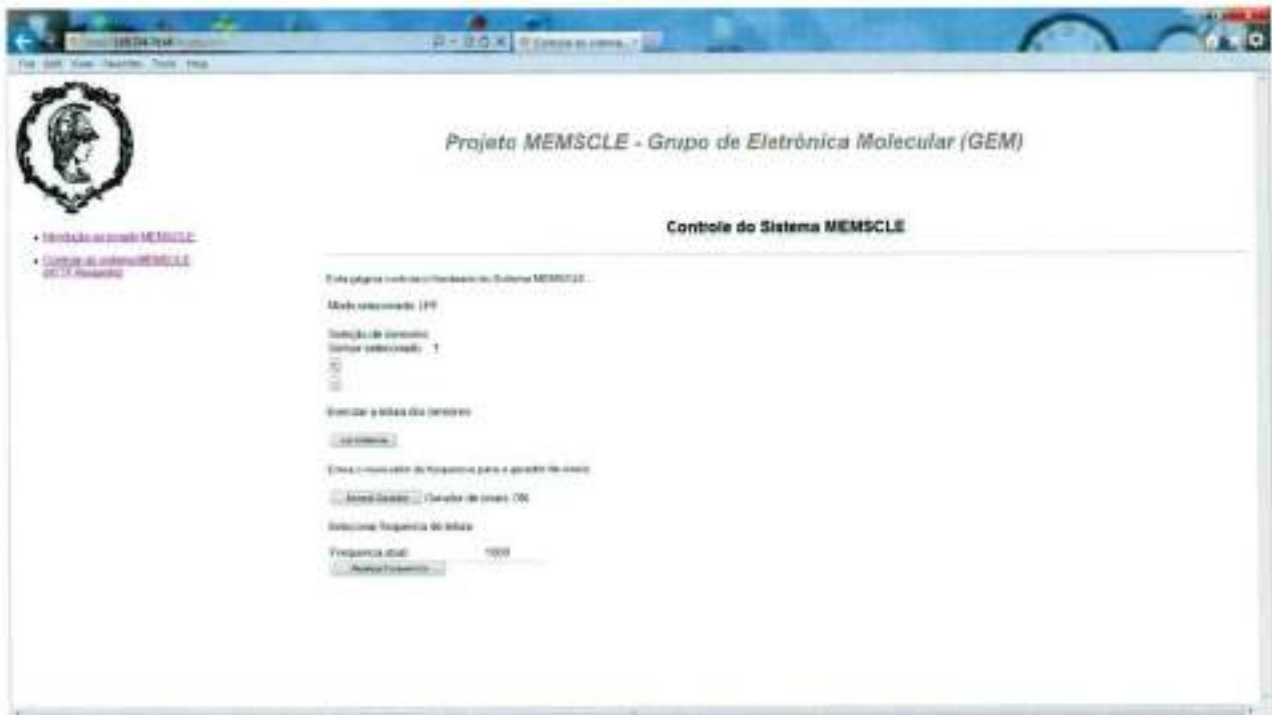


Figura 26: Página de controle do sistema MEMSCLE

#### 4.4 Resultados com capacitores comerciais

Foi implementado no sistema uma escala de medição no range de 20nF a 1000nF. Para tal o sistema foi testado com dois capacitores comerciais e comparado com os valores medidos por um LCR meter da HP modelo 4263A. O Gráfico 5 mostra os dados obtidos.

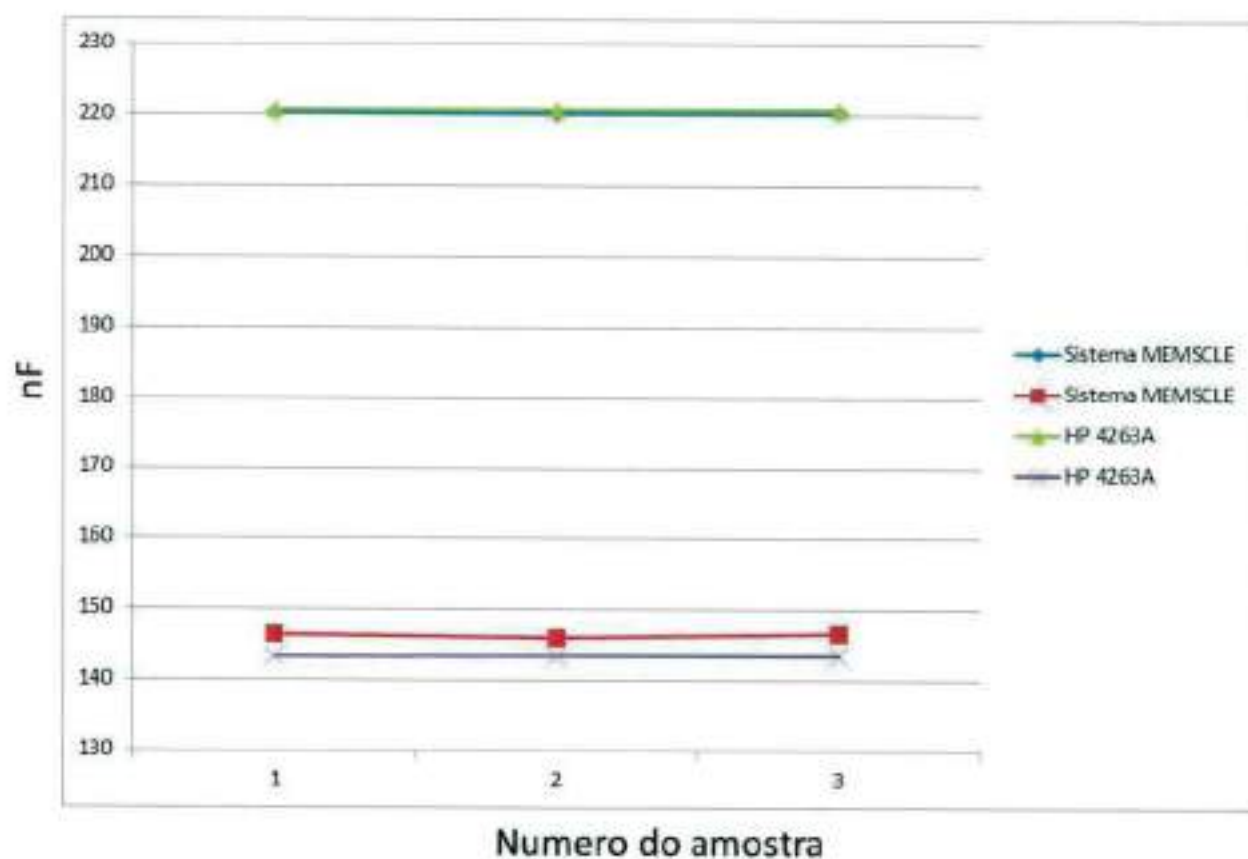


Gráfico 5: Dados obtidos com capacitores comerciais

#### 4.5 Resultados com a língua eletrônica

A seguir serão apresentados dois gráficos. Um dos gráficos (Gráfico 6) mostra a repetibilidade do sistema MEMSCLE ao medir um arranjo de sensores da língua

eletrônica. O outro (Gráfico 7) compara os valores medidos pelo sistema MEMSCLE com os valores medidos pelo sistema atual.

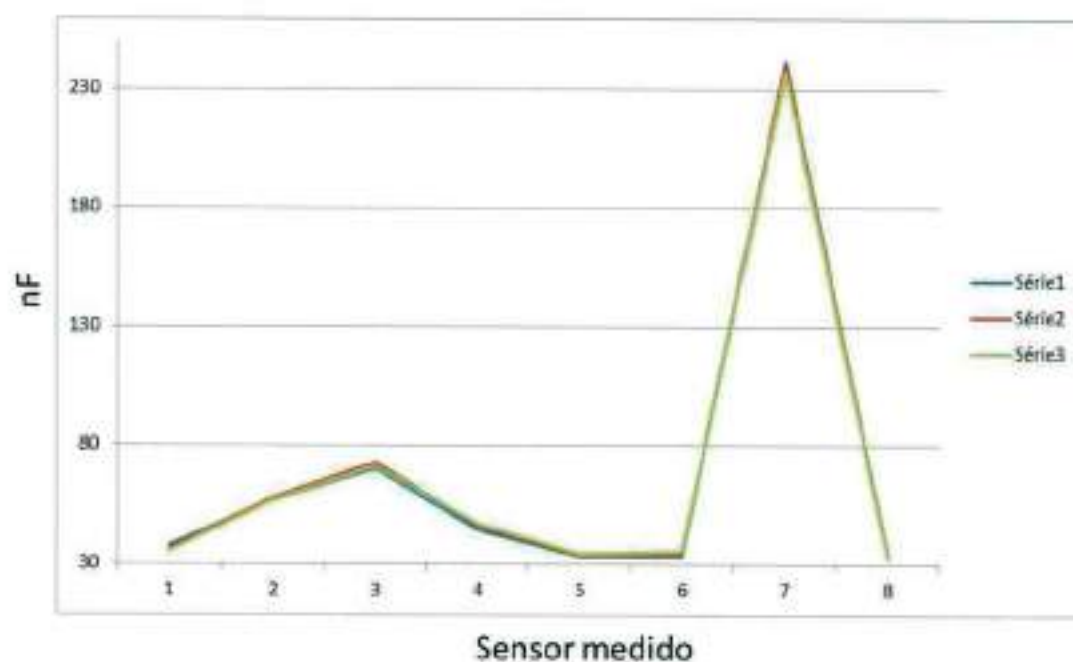


Gráfico 6: Repetibilidade do sistema MEMSCLE

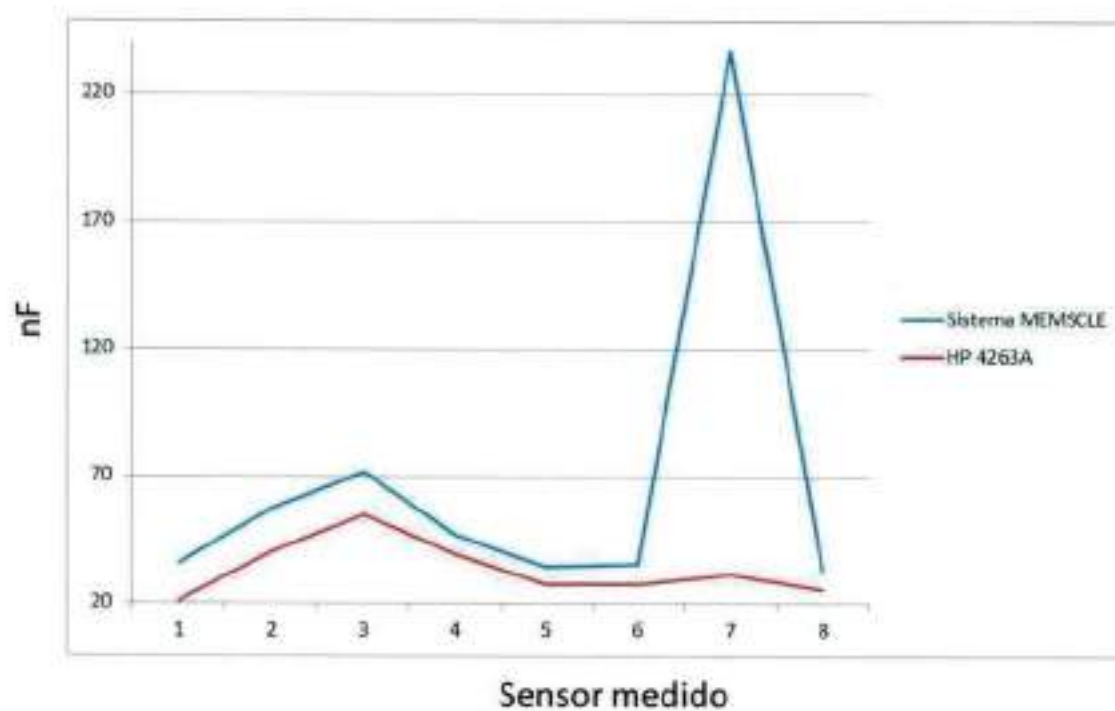


Gráfico 7: Comparação do sistema MEMSCLE com o HP 4263A



Observando os gráficos é possível perceber uma anomalia relativamente grande no Gráfico 7. Ao medir o sensor nomeado como "7" o sistema MEMSCLE forneceu uma medida totalmente fora do padrão. Isso aconteceu porque o sensor em questão apresenta resistência ôhmica muito baixa e a princípio o sistema MEMSCLE foi pensado para medir sensores com resistência relativamente alta. Para o caso do sensor "7" ainda é necessário melhor manipulação matemática para calcular o correto valor da capacitância a partir do valor medido.

Também é possível observar que o padrão de medidas dos outros sensores é totalmente coerente entre o sistema MEMSCLE e o LCR meter HP 4263. O sistema MEMSCLE segue o padrão perfeitamente da língua eletrônica. A língua eletrônica utilizada nos testes apresentou comportamento resistivo influente. Apesar de o sistema MEMSCLE utilizar o modelo paralelo de capacitor (mostrado no item 2.8) para calcular a capacitância e a resistência equivalente ainda é necessário alguma manipulação matemática para corrigir as medidas.

Apesar dos desvios observados no gráfico o sistema MEMSCLE funciona relativamente bem, apresentando valores com baixo desvio para capacitâncias comerciais e seguindo o mesmo padrão de medida na língua eletrônica. Para sistemas que necessitem identificar variações mais do que a medida exata de valores o sistema MEMSCLE é uma ótima opção ainda que sem as melhores manipulações matemática.

#### 4.6 Análise de custo do projeto

Para a análise dos custos do projeto separaremos o sistema em dois. O primeiro é o sistema completo, com todo o *hardware* previsto de forma a melhorar o desempenho do sistema. O segundo é o sistema com *Hardware* reduzido, onde é utilizado apenas o AD do microcontrolador e o sistema de medida das tensões  $V_g$ ,  $V_r$  e  $V_c$  apresentadas no item 2.8.

Sistema completo (preços a varejo e para compra no Brasil):

Placa de circuito impresso (produção): R\$60,00

Resistores: R\$10,00

Capacitores: R\$10,00

Circuitos para a fonte de alimentação e transformador: R\$30,00

Placa de desenvolvimento da Texas Instrument com o LM3S6965: R\$250,00

Circuitos Integrados ADG1409 (6 unidades): R\$148,20

Circuitos Integrados MAX4051 (3 unidades): R\$37,83

Circuitos Integrados AD823 (12 unidades): R\$237,84

Circuito Integrado LMH6624 (1 unidade): R\$12,44

Circuito Integrado AD5932 (1 unidade): R\$20,00

Circuito Integrado 74HC86 (1 unidade): R\$0,48

Circuito Integrado ADS1675 (1 unidade): R\$84,00

Diodos: R\$2,00

Caixas: R\$30,00

Custos de produção: a definir

Custos de software: a definir

**TOTAL:** R\$932.79 + custos de produção + custos de software

É possível observar que o preço do sistema completo ficou abaixo dos R\$1000,00 conforme foi proposto neste projeto. Agora será feita a análise de custo do sistema reduzido, o qual foi apresentado em bom funcionamento neste trabalho.

Sistema reduzido (preços a varejo e para compra no Brasil):

Placa de circuito impresso, se reduzida (produção): R\$40,00

Resistores: R\$8,00

Capacitores: R\$10,00

Circuitos para a fonte de alimentação e transformador: R\$30,00

Placa de desenvolvimento da Texas Instrument com o LM3S6965: R\$250,00

Circuitos Integrados ADG1409 (4 unidades): R\$98,80

Circuitos Integrados MAX4051 (2 unidades): R\$25,22

Circuitos Integrados AD823 (6 unidades): R\$118,92

Circuito Integrado LMH6624 (1 unidade): R\$12,44

Circuito Integrado AD5932 (1 unidade): R\$20,00

Circuito Integrado 74HC86 (0 unidade): R\$0,00

Circuito Integrado ADS1675 (0 unidade): R\$0,00

Diodos: R\$0,00

Caixas: R\$20,00

Custos de produção: a definir

Custos de software: a definir

**TOTAL:** R\$633,38 + custos de produção + custos de software

Observa-se que por menos de R\$700,00 é possível construir um sistema com características semelhantes ao sistema completo, porém menor custo. Se considerarmos que os custos apresentados são a varejo, então para a produção de 1000 unidades dos produtos haveria uma redução de aproximadamente 20% nos custos, resultando em um custo de material de aproximadamente R\$500,00.

É claro que para o custo final é necessário somar os custos de produção e os custos de *software*, mas é possível ter-se ideia de que o sistema ficaria bem mais barato que os equipamentos mais similares vendidos no mercado. Não existe no mercado, por exemplo, nenhum equipamento dedicado a medir a língua eletrônica com um multiplexador e medidor de capacitância embutidos.

## 5. CONCLUSÕES

O sistema MEMSCLE é uma excelente alternativa de baixo custo para o uso com a língua eletrônica. Deve-se ainda melhorar as manipulações dos dados medidos para otimizar a performance do sistema. Até sem nenhuma manipulação matemática o sistema MEMSCLE apresenta uma excelente solução para identificar variações de capacitância e resistência considerando o modelo paralelo para este sistema.

A medida do sistema é realizada através de tensão senoidal com média 0V de forma que não danifica os sensores da língua eletrônica, pois os mesmos não podem receber corrente contínua para não ocorrer polarização permanente dos sensores.

Para a medida de capacitores comerciais o sistema MEMSCLE mostrou-se muito bom. Ainda falta corrigir alguns detalhes de cálculos matemáticos a serem integrados no microcontrolador.

De modo geral o sistema MEMSCLE é uma ótima opção para a medida de capacitâncias através de multiplexação e para o uso com língua eletrônica devido a sua capacidade de identificar padrões.

Pelo fato das dimensões do sistema serem reduzidas o mesmo pode ser considerado portátil. A língua eletrônica é utilizada na análise de diversas substâncias e uma delas é a água que bebemos. Empresas de saneamento básico precisam medir a qualidade da água de suas represas rotineiramente. O resultado da análise da água pode demorar alguns dias, mas se utilizarmos o sistema MEMSCLE existe a possibilidade de esse resultado ser instantâneo.

É possível incluir junto ao sistema um módulo autônomo de energia, como células fotovoltaicas, e uma interface para a internet. Dessa forma o *Hardware* do sistema MEMSCLE pode ser instalado a distância e todos os dados acessados remotamente através de uma interface web.

Tudo o discutido neste tópico demonstra alguns exemplos dos possíveis usos para o sistema MEMSCLE. O projeto descrito neste trabalho é um exemplo real da possibilidade da portabilidade da análise de substâncias.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] VLASOV, YU. G.; LEGIN, A. V.; RUDNITSKAYA, A. M. Multisensor systems of the electronic tongue type as novel opportunities in design and application of chemical sensors. *Russian Chemical Reviews* 75, 125 ± 132, 2006.
- [2] CIOSEK, PATRYCJA; WROBLEWSKI, WOJCIECH. Sensor arrays for liquid sensing – electronic tongue systems. *Analyst* 132, 963–978, 2007.
- [3] BEEBE, K.R., PEEL, R.J., SEASHOLTZ, M.B.; *Chemometrics: a practical guide*, John Wiley and Sons, New York, 1998.
- [4] MUSIO, F.; FERRARA, M. C. Low frequency a.c. response of polypyrrole gas sensors. *Sensors and actuators B* 41, 97-103, 1997
- [5] KAWAMURA, Y.; KARE, M. R. *Umami: a Basic Taste*. New York: Marcel Dekker, 1987
- [6] STEWART, G. F.; AMERINE, M. A. *Introduction to Food Science and Technology*. New York: Academic Press, 1973
- [7] VLASOV; LEGIN, A.; RUDNITSKAYA, A. *Anal. Bioanal. Chem.* 373 136, 2002
- [8] SCHIFFMAN, S. S.; PEARCE, T. C. *Handbook of Machine Olfaction. Electronic Nose Technology* (Eds T C Pearce, S S Schiffman, H T Nagle, J W Gardner), Darmstadt: Wiley-VCH Ch. 1, 2003
- [9] ORSINI, L. Q.; CONSONNI, D. *Curso de Circuitos Elétricos, 2a Edição, Volume 1*, Editora Edgard Blucher LTDA
- [10] Howto Use a LCR Meter; <http://kb.bkprecision.com/questions.php?questionid=22>; site acessado em 05/2012
- [11] AC bridge circuits; [http://www.allaboutcircuits.com/vol\\_2/chpt\\_12/5.html](http://www.allaboutcircuits.com/vol_2/chpt_12/5.html); site acessado em 05/2012
- [12] PEDRONI, V. A. *Eletrônica Digital Moderna e VHDL*, Editora Campus

- [13] Analog Devices Datasheet; 5 $\Omega$  Max Ron, 4-/8-Channel  $\pm 15V/12V/\pm 5V$  Multiplexers; ADG1408/ADG1409
- [14] UPPU, R. AC Bridges : Measuring Capacitance using de Sauty and Schering Bridges
- [15] RAVEN, M. S.; RAVEN, D. New Approaches to the Direct Measurement of Capacitance. *Electrocomponent Science and Technology* Vol. 4: 37-42, 1977.
- [16] File:1st Order Lowpass Filter RC.svg;  
[http://en.wikipedia.org/wiki/File:1st\\_Order\\_Lowpass\\_Filter\\_RC.svg](http://en.wikipedia.org/wiki/File:1st_Order_Lowpass_Filter_RC.svg); site acessado em 05/2012
- [17] Passive LowPassFilter; [http://www.electronics-tutorials.ws/filter/filter\\_2.html](http://www.electronics-tutorials.ws/filter/filter_2.html); site acessado em 05/2012
- [18] HP4286A LCR Meter Service Manual
- [19] Texas Instrument Datasheet, Stellaris® LM3S6965 Microcontroller DATA SHEET
- [20] Texas Instrument Datasheet; 4MSPS, 24-Bit Analog-to-Digital Converter; ADS1675
- [21] Analog Devices Datasheet; Programmable Frequency Scan Waveform Generator; AD5932
- [22] Analog Devices Datasheet; Dual, 16 MHz, Rail-to-Rail FET Input Amplifier; AD823
- [23] TASSARA, B. P. Analisador de Amplitude e Fase, Universidade Federal do Rio de Janeiro, Março de 2009
- [24] CARDOSO, D. H. Medidor de Impedância de Componentes Passivos Utilizando Microcontrolador PIC, Universidade Federal do Rio de Janeiro, Março de 2011
- [25] NOSOLINE, M. V. S., Estudo Comparativo entre dois métodos de medição de impedância
- [26] CARTER, B. Fully Differential Op Amps Made Easy, Application Report SLOA099 - May 2002
- [27] BULL, K. Methods of Accurately Measuring Capacitive RH Sensors, 5th International Symposium on Humidity and Moisture – ISHM 2006 Brazil May 02 – 05, 2006 – Rio de Janeiro, Brazil



[28] National Semiconductor Datasheet, Single Ultra Low Noise Wideband Operational Amplifier, LMH6624

[29] WALDRIP, M.; TSE, R. Measurement Tips for High Capacitance MLCC's, TDK publications

[30] DONTI, S. S. Capacitance based Tomography for Industrial Applications, M. Tech. credit seminar report, Electronic Systems Group, EE Dept. IIT Bombay, submitted November 2004

[31] The operational amplifier;

[http://www.physics.udel.edu/~nowak/phys645/The\\_operational\\_amplifier.htm](http://www.physics.udel.edu/~nowak/phys645/The_operational_amplifier.htm); site acessado em 05/2012

[32] RLC BRIDGE;

[http://www.seekic.com/circuit\\_diagram/Measuring\\_and\\_Test\\_Circuit/RLC\\_BRIDGE.html](http://www.seekic.com/circuit_diagram/Measuring_and_Test_Circuit/RLC_BRIDGE.html); site acessado em 05/2012

[33] Stellaris® ARM® Cortex™-M3-based Microcontrollers - 6000 Series;

<http://www.ti.com/mcu/docs/mcoluminaryfamilynode.tsp?sectionId=95&tabId=2599&familyId=1758&docCategoryId=10&viewType=mostrecent>; site acessado em 05/2012

[34] An Even Better LC Meter Based on the AVR ATTINY861;

[http://cappels.org/dproj/EvenBetterLCMeter/Even\\_Better\\_LC\\_Meter.html](http://cappels.org/dproj/EvenBetterLCMeter/Even_Better_LC_Meter.html); site acessado em 05/2012

[35] All About Circuits; <http://forum.allaboutcircuits.com/showthread.php?t=49625>; site acessado em 05/2012

[36] Extech LCR meter schematics; <http://music-electronics-forum.com/t14501/>; site acessado em 05/2012

[37] XD201 – Eletrônica Digital; <http://pt.scribd.com/doc/49494181/78/Multiplexadores-e-demultiplexadores-analogicos>; site acessado em 05/2012

[38] Building a differential amplifier; [http://www.allaboutcircuits.com/vol\\_3/chpt\\_8/9.html](http://www.allaboutcircuits.com/vol_3/chpt_8/9.html); site acessado em 05/2012

[39] OperationalAmplifierCircuits; <http://webpages.ursinus.edu/lriley/ref/circuits/node5.html>; site acessado em 05/2012

[40] Summing amplifier using opamp; <http://www.circuitstoday.com/summing-amplifier>; site acessado em 05/2012

[41] How-to use PWM to Generate Analog (or Analogue) Voltage in Digital Circuits – Part 2; <http://dev.emcelettronica.com/how-to-use-pwm-to-generate-analog-or-analogue-voltage-digital-circuits-part-2>; site acessado em 05/2012

[42] High-speed comparators provide many useful circuit functions when used correctly; <http://www.analog.com/library/analogDialogue/Anniversary/5.html>; site acessado em 05/2012

## ANEXOS

### Anexo A: Pedacos de código “c” do programa do microcontrolador

```

//
// Retorna o modo analogico selecionado
//
void
io_get_moda(char *pcBuf, intBuflen)
{
    //
    // Escreve o modo selecionado
    //
    if(GPIORnRead(GPIO_PORTC_BASE, GPIO_PIN_0))
    {
        usprintf(pcBuf, iBuflen, "LPF");
        RIT128x56x4Enable(1000000);
        RIT128x56x4StringDraw("Modo LPF", 0, 53, 15);
    }
    else
    {
        usprintf(pcBuf, iBuflen, "HP");
        RIT128x56x4Enable(1000000);
        RIT128x56x4StringDraw("Modo HP", 0, 53, 15);
    }
}

//
// Retorna o modo: 0 se HP e 1 se LPF.
//
int
io_is_moda_LPF(void)
{
    if(GPIORnRead(GPIO_PORTC_BASE, GPIO_PIN_0))
    {
        return(1);
    }
    else
    {
        return(0);
    }
}

//
// Seleciona o sensor a ser medido
//
void
selecionarSensor(int sensor)
{
    switch(sensor)
    {
        case 1:
            GPIOWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
            GPIOWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
            RIT128x56x4Enable(1000000);
            RIT128x56x4StringDraw("Sensor 1 selecionado.", 0, 60, 12);
            break;
        case 2:
            GPIOWrite(GPIO_PORTA_BASE, GPIO_PIN_7, ~0);
            GPIOWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
            GPIOWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
    }
}

```

```

RIT128x96x4Enable(1000000);
RIT128x96x4StringDraw("Sensor 2 seleccionado.", 0, 60, 12);
break;
case 3:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 3 seleccionado.", 0, 60, 12);
  break;
case 4:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, -0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 4 seleccionado.", 0, 60, 12);
  break;
case 5:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 5 seleccionado.", 0, 60, 12);
  break;
case 6:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, -0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 6 seleccionado.", 0, 60, 12);
  break;
case 7:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 7 seleccionado.", 0, 60, 12);
  break;
case 8:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, -0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, 0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 8 seleccionado.", 0, 60, 12);
  break;
case 9:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, -0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 9 seleccionado.", 0, 60, 12);
  break;
case 10:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, -0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, -0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Sensor 10 seleccionado.", 0, 60, 12);
  break;
case 11:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, 0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
  GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, -0);
  RIT128x96x4Enable(1000000);
  RIT128x96x4StringDraw("Aberto seleccionado.", 0, 60, 12);
  break;
case 12:
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_7, -0);
  GPIOPinWrite(GPIO_PORTA_BASE, GPIO_PIN_6, -0);

```

```

        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_4, 0);
        GPIOPinWrite(GPIO_PORTC_BASE, GPIO_PIN_5, -0);
        RTT128x96x4Enable(1000000);
        RTT128x96x4StringDraw("Outseledonado.  ", 0, 60, 12);
        break;
    default:
        RTT128x96x4Enable(1000000);
        RTT128x96x4StringDraw("Sensor invalido! ", 0, 60, 12);
    }
}

int
getSensor(void)
{
    if (GPIOPinRead(GPIO_PORTC_BASE, GPIO_PIN_5))
    {
        if (GPIOPinRead(GPIO_PORTC_BASE, GPIO_PIN_4))
        {
            RTT128x96x4Enable(1000000);
            RTT128x96x4StringDraw("Sensor invalido! ", 0, 60, 12);
            return (0);
        }
        else
        {
            if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_5))
            {
                if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
                {
                    return (12);
                }
            }
            else
            {
                return (11);
            }
        }
    }
    else
    {
        if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
        {
            return (10);
        }
        else
        {
            return (9);
        }
    }
}
else
{
    if (GPIOPinRead(GPIO_PORTC_BASE, GPIO_PIN_4))
    {
        if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_6))
        {
            if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
            {
                return (8);
            }
        }
        else
        {
            return (7);
        }
    }
    else
    {
        if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
        {
            return (6);
        }
        else
        {
            return (5);
        }
    }
}
else

```

```

    }
    if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_6))
    {
        if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
        {
            return (4);
        }
        else
        {
            return (3);
        }
    }
    else
    {
        if (GPIOPinRead(GPIO_PORTA_BASE, GPIO_PIN_7))
        {
            return (2);
        }
        else
        {
            return (1);
        }
    }
}

void enviarDados_gr(nNewFreq)
{
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_2, 0);
    GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_3, 0);

    float freqF;
    int freq;

    freqF = 4.026 * nNewFreq;
    freq = (int)freqF;

    int freqV[24];
    int cnt = 23;
    while (cnt > -1) {
        freqV[cnt] = 0;
        cnt--;
    }
    cnt = 23;
    while (freq > 0) {
        if (freq % 2 == 1) {
            freqV[cnt] = 1;
        }
        cnt--;
        freq = freq / 2;
    }

    // set-up controle
    int dado[16];
    dado[0] = 0;
    dado[1] = 0;
    dado[2] = 0;
    dado[3] = 0;
    dado[4] = 0;
    dado[5] = 1;
    dado[6] = 1;
    dado[7] = 0;
    dado[8] = 1;
    dado[9] = 1;
    dado[10] = 1;
    dado[11] = 1;
    dado[12] = 0;
    dado[13] = 0;
    dado[14] = 1;
    dado[15] = 1;
    comunicarGerador(dado);

    // set-apnúmero de incrementos
    dado[0] = 0;
    dado[1] = 0;

```

```

dado[2]=0;
dado[3]=1;
dado[4]=0;
dado[5]=0;
dado[6]=0;
dado[7]=0;
dado[8]=0;
dado[9]=0;
dado[10]=0;
dado[11]=0;
dado[12]=0;
dado[13]=0;
dado[14]=1;
dado[15]=0;
comunicarGerador(dado);

// set-up data 1 baixo
dado[0]=0;
dado[1]=0;
dado[2]=1;
dado[3]=0;
dado[4]=0;
dado[5]=0;
dado[6]=0;
dado[7]=0;
dado[8]=0;
dado[9]=0;
dado[10]=0;
dado[11]=0;
dado[12]=0;
dado[13]=0;
dado[14]=1;
dado[15]=0;
comunicarGerador(dado);

// set-up data 1 alto
dado[0]=0;
dado[1]=0;
dado[2]=1;
dado[3]=1;
dado[4]=0;
dado[5]=0;
dado[6]=0;
dado[7]=0;
dado[8]=0;
dado[9]=0;
dado[10]=0;
dado[11]=0;
dado[12]=0;
dado[13]=0;
dado[14]=0;
dado[15]=0;
comunicarGerador(dado);

// set-up tempo de intervalo
dado[0]=0;
dado[1]=1;
dado[2]=1;
dado[3]=0;
dado[4]=0;
dado[5]=0;
dado[6]=0;
dado[7]=0;
dado[8]=0;
dado[9]=0;
dado[10]=0;
dado[11]=0;
dado[12]=1;
dado[13]=1;
dado[14]=1;
dado[15]=1;
comunicarGerador(dado);

// set-up Fc&n baixo
dado[0]=1;
dado[1]=1;
dado[2]=0;
dado[3]=0;

```



```

    dado[4]=FreqV[12];
    dado[5]=FreqV[13];
    dado[6]=FreqV[14];
    dado[7]=FreqV[15];
    dado[8]=FreqV[16];
    dado[9]=FreqV[17];
    dado[10]=FreqV[18];
    dado[11]=FreqV[19];
    dado[12]=FreqV[20];
    dado[13]=FreqV[21];
    dado[14]=FreqV[22];
    dado[15]=FreqV[23];
    comunicarGerador(dado);

// set-up Freqt alto
dado[0]=1;
dado[1]=1;
dado[2]=0;
dado[3]=1;
dado[4]=FreqV[0];
dado[5]=FreqV[1];
dado[6]=FreqV[2];
dado[7]=FreqV[3];
dado[8]=FreqV[4];
dado[9]=FreqV[5];
dado[10]=FreqV[6];
dado[11]=FreqV[7];
dado[12]=FreqV[8];
dado[13]=FreqV[9];
dado[14]=FreqV[10];
dado[15]=FreqV[11];
comunicarGerador(dado);

freq = newfreq;

charFrequencia[32];
sprintf(Frequencia, 32, "Freq: %d Hz", freq);
RIT128x56x4Enable(100000);
RIT128x56x4StringDraw(Frequencia, 0, 55, 15);
}

voidcomunicarGerador(int * dado)
{
    GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_1, 0);
    for (int i = 0; i < 10000; i++)
    {
    }
    for (int i = 0; i < 16; i++)
    {
        GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_2, ~0);
        for (int j = 0; j < 10000; j++)
        {
        }
        if (dado[i] == 0)
        {
            GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_3, 0);
            for (int j = 0; j < 10000; j++)
            {
            }
        }
        else
        {
            GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_3, ~0);
            for (int j = 0; j < 10000; j++)
            {
            }
        }
        GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_2, 0);
        for (int j = 0; j < 10000; j++)
        {
        }
    }
    GPIOWrite(GPIO_PORTB_BASE, GPIO_PIN_1, ~0);
    for (int i = 0; i < 10000; i++)
    {
    }
}

```

```

void startGerador()
{
    GPIOWrite(GPIO_PORTD_BASE, GPIO_PIN_7, 0);
    for (int j = 0; j < 10000; j++)
    {
        GPIOWrite(GPIO_PORTD_BASE, GPIO_PIN_7, -0);
        for (int i = 0; i < 10000; i++)
        {
            GPIOWrite(GPIO_PORTD_BASE, GPIO_PIN_7, 0);
            for (int j = 0; j < 10000; j++)
            {
            }
        }
    }

    floatlerAD()
    {
        unsigned long uData[1];
        int leitura[100], maximo[16], minimo[16];
        float media_maximo, media_minimo, vpp;

        for (int j = 0; j < 16; j++)
        {
            maximo[j] = 0;
            minimo[j] = 1023;
        }

        SysCtlPeripheralEnable(SYSCTL_PERIPH_ADC0);
        GPIOPinTypeADC(GPIO_PORTD_BASE, GPIO_PIN_7);
        ADCSequenceDisable(ADC0_BASE, 3);
        ADCSequenceConfigure(ADC0_BASE, 3, ADC_TRIGGER_PROCESSOR, 0);
        ADCSequenceStepConfigure(ADC0_BASE, 3, 0, ADC_CTL_CH0 | ADC_CTL_E | ADC_CTL_END);
        ADCSequenceEnable(ADC0_BASE, 3);
        ADCIntClear(ADC0_BASE, 3);

        for (int j = 0; j < 16; j++)
        {
            for (int i = 0; i < 100; i++)
            {
                // Iniciar a amostra da sequencia.
                // Iniciar a conversao.
                //
                ADCProcessorTrigger(ADC0_BASE, 3);
                //
                // Aguarde ate que a conversao seja completada.
                //
                while(ADCIntStatus(ADC0_BASE, 3, false))
                {
                }
                //
                // Lero o valor do AD.
                //
                ADCSequenceDataGet(ADC0_BASE, 3, uData);
                ADCIntClear(ADC0_BASE, 3);
                leitura[i] = uData[0];
            }
            ADCIntClear(ADC0_BASE, 3);
            for (int i = 0; i < 100; i++)
            {
                if (leitura[i] > maximo[j] && (leitura[i] < 320))
                {
                    maximo[j] = leitura[i];
                }
            }
            for (int i = 0; i < 100; i++)
            {
                if (leitura[i] < minimo[j] && (leitura[i] > 130))
                {
                    minimo[j] = leitura[i];
                }
            }
        }

        media_maximo = 0;
        media_minimo = 0;
    }
}

```

```

int verifica1, verifica2;
verifica1 = 15;
verifica2 = 15;
for (int j = 0; j < 16; j++)
{
    if(maximo[j] == 0)
    {
        verifica1 = verifica1 - 1;
    }
    if(minimo[j] == 1023)
    {
        minimo[j] = 0;
    }
    verifica2 = verifica2 - 1;
}

for (int j = 0; j < 16; j++)
{
    media_maximo = (float)maximo[j] + media_maximo;
    media_minimo = (float)minimo[j] + media_minimo;
}

media_maximo = media_maximo / verifica1;
media_minimo = media_minimo / verifica2;

vpp = media_maximo - media_minimo;

return vpp;
}

void selectOut(int res)
{
    switch(res)
    {
        case 0: // selecionar Vout 5
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_4, 0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_5, 0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_6, ~0);
            break;
        case 1: // selecionar Vout 3
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_4, 0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_5, ~0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_6, 0);
            break;
        case 2: // selecionar Vout 4
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_4, ~0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_5, ~0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_6, 0);
            break;
        default:
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_4, 0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_5, ~0);
            GPIOPinWrite(GPIO_PORTB_BASE, GPIO_PIN_6, 0);
    }
}

void calcula_Cap_Res(float Gvpp, float Rvpp, float Cvpp, float resistencia[1], float capacitancia[1], int escala)
{
    float Rt, cosT, Tl, Rcl, I, nistor;

    resistencia[0] = 0;
    capacitancia[0] = 0;

    if (escala == 2)
    {
        resistor = 530;
    }
    else
    {
        resistor = 1060;
    }

    cosT = ((pow(Rvpp, 2)) + (pow(Cvpp, 2)) - pow(Gvpp, 2)) / (2 * Rvpp * Cvpp);
    if (cosT < 0)
    {
        cosT = (float)(-1) * cosT;
    }

```

```

}
Tl = acos(cosT);
I = Rcpp / resistor;
Rt1 = (1 * 50) + resistor;
Rcpp = 1 * Rt1;
Rd = (Cupp / Rcpp) * resistor;
resistencia[0] = (pow(Rt1, 2)) * (1 + (pow(tan(Tl), 2)));
resistencia[0] = (sqrt(resistencia[0]));
capacitancia[0] = tan(Tl) / (resistencia[0] * 6.283185307 * freq);
}

void selecionarEscala(int escala)
{
    switch(escala)
    {
        case 1: // selecionar resistor de 1k
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_2, 0);
            GPIOPinWrite(GPIO_PORTG_BASE, GPIO_PIN_1, -0);
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_3, 0);
            break;
        case 2: // selecionar resistor de 470R
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_2, 0);
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_1, -0);
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_3, -0);
            break;
        default:
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_2, 0);
            GPIOPinWrite(GPIO_PORTG_BASE, GPIO_PIN_1, -0);
            GPIOPinWrite(GPIO_PORTD_BASE, GPIO_PIN_3, 0);
    }
}

```

## Anexo B: Pedacos de código “html” e “Javascript” das páginas embarcadas no sistema

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<!-- Copyright (c) 2009 Luminary Micro, Inc. All rights reserved. -->
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-1"><title>Controle do sistema MEMSCLC</title>

<script language="JavaScript">
<!--

function Modo_selecionado()
{
    var modo = false;
    function modoCompleto()
    {
        if(modoseadyState == 4)
        {
            if(modo.status == 200)
            {
                document.getElementById("modoSelecionado").innerHTML = "<div>" + modo.responseText + "</div>";
            }
        }
    }
    if(window.XMLHttpRequest)
    {
        modo = new XMLHttpRequest();
    }
    else if(window.ActiveXObject)
    {

```

```

modo = new ActiveXObject("Microsoft.XMLHTTP");
}
if(modo)
{
modo.open("GET", "modoseleccionado?id=" + Math.random(), true);
modo.onreadystatechange = modoCompleto;
modo.send(null);
}
}

functionSebeSensor()
{
var sensor = false;
var sensorAtual = false;
function sensorCompleto()
{
if(sensorAtual.readyState == 4)
{
if(sensorAtual.status == 200)
{
document.getElementById("sensorSeleccionado").innerHTML = "<div>" + sensorAtual.responseText + "</div>";
}
}
}
if(window.XMLHttpRequest)
{
sensor = new XMLHttpRequest();
sensorAtual = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
sensor = new ActiveXObject("Microsoft.XMLHTTP");
sensorAtual = new ActiveXObject("Microsoft.XMLHTTP");
}
if(sensor)
{
sensor.open("GET", "cgi-bin/sobe_sensor?id=" + Math.random(), true);
sensor.send(null);
}
if(sensorAtual)
{
sensorAtual.open("GET", "sensorseleccionado?id=" + Math.random(), true);
sensorAtual.onreadystatechange = sensorCompleto;
sensorAtual.send(null);
}
}

functionDesceSensor()
{
var sensor = false;
var sensorAtual = false;
function sensorCompleto()
{
if(sensorAtual.readyState == 4)
{
if(sensorAtual.status == 200)
{

```

```

document.getElementById("sensorSelecioneado").innerHTML = "<div>" + sensorAtual.responseText + "</div>";
    }
    }
}
if(window.XMLHttpRequest)
{
    sensor = new XMLHttpRequest();
    sensorAtual = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
    sensor = new ActiveXObject("Microsoft.XMLHTTP");
    sensorAtual = new ActiveXObject("Microsoft.XMLHTTP");
}
if(sensor)
{
    sensor.open("GET", "icg-bin/desce_sensor?id=" + Math.random(), true);
    sensor.send(null);
}
if(sensorAtual)
{
    sensorAtual.open("GET", "sensorselecioneado?id=" + Math.random(), true);
    sensorAtual.onreadystatechange = sensorCompleto;
    sensorAtual.send(null);
}
}

functionSensor_selecioneado()
{
    var sensorAtual = false;
    function sensorCompleto()
    {
        if(sensorAtual.readyState == 4)
        {
            if(sensorAtual.status == 200)
            {
                document.getElementById("sensorSelecioneado").innerHTML = "<div>" + sensorAtual.responseText + "</div>";
            }
        }
    }
}
if(window.XMLHttpRequest)
{
    sensorAtual = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
    sensorAtual = new ActiveXObject("Microsoft.XMLHTTP");
}
if(sensorAtual)
{
    sensorAtual.open("GET", "sensorselecioneado?id=" + Math.random(), true);
    sensorAtual.onreadystatechange = sensorCompleto;
    sensorAtual.send(null);
}
}

functionselecuaLetura()

```



```

{
  var req = false;
  var led = false;
  function ledComplete()
  {
    if(led.readyState == 4)
    {
      if(led.status == 200)
      {
        document.getElementById("estadoLectura").innerHTML = "<div>" + led.responseText + "</div>";
      }
    }
  }
  if(window.XMLHttpRequest)
  {
    req = new XMLHttpRequest();
    led = new XMLHttpRequest();
  }
  else if(window.ActiveXObject)
  {
    req = new ActiveXObject("Microsoft.XMLHTTP");
    led = new ActiveXObject("Microsoft.XMLHTTP");
  }
  if(req)
  {
    req.open("GET", "logi-bin/lextula_leitura?id=" + Math.random(), true);
    req.send(null);
  }
  if(led)
  {
    led.open("GET", "ledstate?id=" + Math.random(), true);
    led.onreadystatechange = ledComplete;
    led.send(null);
  }
}

function aonaFrecuencia()
{
  var req = false;
  var pwm = false;
  function pwmComplete()
  {
    if(pwm.readyState == 4)
    {
      if(pwm.status == 200)
      {
        document.getElementById("pwmstate").innerHTML = "<div>" + pwm.responseText + "</div>";
      }
    }
  }
  if(window.XMLHttpRequest)
  {
    req = new XMLHttpRequest();
    pwm = new XMLHttpRequest();
  }
  else if(window.ActiveXObject)
  {
    req = new ActiveXObject("Microsoft.XMLHTTP");
  }
}

```

```

pwm = new ActiveXObject("Microsoft.XMLHTTP");
}
if(req)
{
req.open("GET", "cgi-bin/acionar_frequencia?id=" + Math.random(), true);
req.send(null);
}
if(pwm)
{
pwm.open("GET", "pwmstate?id=" + Math.random(), true);
pwm.onreadystatechange = pwmComplete;
pwm.send(null);
}
}

function pwm_freq_get()
{
var req = false;
var pwmfreq = false;
var freqText = document.getElementById("pwmfreqtxt");
function pwmfreqComplete()
{
}
if(pwmfreq.readyState == 4)
{
if(pwmfreq.status == 200)
{
document.getElementById("pwmfreq").innerHTML = "<div>" + pwmfreq.responseText + "</div>";
}
}
}
if(window.XMLHttpRequest)
{
req = new XMLHttpRequest();
pwmfreq = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
req = new ActiveXObject("Microsoft.XMLHTTP");
pwmfreq = new ActiveXObject("Microsoft.XMLHTTP");
}
if(req)
{
if(freqText.value != "")
{
req.open("GET", "pwm_freq?value=" + freqText.value + "&id=" + Math.random(), true);
req.send(null);
}
}
if(pwmfreq)
{
pwmfreq.open("GET", "pwmfreqget?id=" + Math.random(), true);
pwmfreq.onreadystatechange = pwmfreqComplete;
pwmfreq.send(null);
}
}

function ledstateGet()

```

```

{
var led = false;
function ledComplete()
{
if(led.readyState == 4)
{
if(led.status == 200)
{
document.getElementById("ledstate").innerHTML = "<div>" + led.responseText + "</div>";
}
}
}
if(window.XMLHttpRequest)
{
led = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
led = new ActiveXObject("Microsoft.XMLHTTP");
}
if(led)
{
led.open("GET", "ledstate?id=" + Math.random(), true);
led.onreadystatechange = ledComplete;
led.send(null);
}
}
function pwmstateGet()
{
var pwm = false;
function pwmComplete()
{
if(pwm.readyState == 4)
{
if(pwm.status == 200)
{
document.getElementById("pwmstate").innerHTML = "<div>" + pwm.responseText + "</div>";
}
}
}
if(window.XMLHttpRequest)
{
pwm = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
pwm = new ActiveXObject("Microsoft.XMLHTTP");
}
if(pwm)
{
pwm.open("GET", "pwmstate?id=" + Math.random(), true);
pwm.onreadystatechange = pwmComplete;
pwm.send(null);
}
}
function pwmfreqGet()
{

```

```

var pwmfreq = false;
function pwmfreqComplete()
{
  if(pwmfreq.readyState == 4)
  {
    if(pwmfreq.status == 200)
    {
      document.getElementById("pwmfreq").innerHTML += "<div>" + pwmfreq.responseText + "</div>";
    }
  }
}
if(window.XMLHttpRequest)
{
  pwmfreq = new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
  pwmfreq = new ActiveXObject("Microsoft.XMLHTTP");
}
if(pwmfreq)
{
  pwmfreq.open("GET", "pwmfreqget?id=" + Math.random(), true);
  pwmfreq.onreadystatechange = pwmfreqComplete;
  pwmfreq.send(null);
}
}
//-->
</script>
<style type="text/css">
body
{
  font-family: Arial;
  background-color: white;
  margin: 10px;
  padding: 0px;
}
h1
{
  color: #7C7369;
  font-family: Arial;
  font-size: 24pt;
  font-style: italic;
}
h2
{
  color: #000000;
  font-family: Arial;
  font-size: 18pt;
  font-style: bold;
}
h3
{
  color: #7C7369;
  font-family: Arial;
  font-size: 12pt;
  font-style: bold;
}

```

```

</style>
</head>
<body onload="ledstateGet();pwmstateGet();pwmfreqGet();Modo_selecionado();Sensor_selecionado()">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="left" valign="center">

</td>
<td align="center" valign="center">
<h1>Projeto MEMSCL - Grupo de Eletrônica Molecular (GEM)</h1>
</td>
</tr>
</table>
<table width="100%">
<tbody>
<tr>
<td align="left" valign="top" width="25%">
<br>
<a href="index.htm">Introdução ao projeto MEMSCL</a>
<br>
<br>
<br>
<br>
<br>
<a href="io_http.htm">Controle do sistema MEMSCL<br>
(HTTP Requests)</a>
<br>
<br>
</td>
<td align="left" valign="top" width="75%">
<center>
<h2 align="center">Controle do Sistema MEMSCL</h2>
<center>
<hr size="2" width="100%">
<p>Esta página controla o Hardware do Sistema MEMSCL.</p>
<p></p>
<table>
<tbody>
<tr>
<td>Modo selecionado:</td>
</tr>
</tbody>
</table>
<div id="modoSelecionado" align="center"> - </div>
</td>
</tr>
</tbody>
</table>
<table>
<tbody>
<tr>
<td>Seleção de Sensores:</td>
</tr>
<tr>
<td>Sensor selecionado:</td>
</tr>
</tbody>
</table>
<div id="sensorSelecionado" align="center"> - </div>

```

```

</td>
</tr>
<tr>
<td><input id="Sensor +" value="+" onclick="SobeSensor()" type="button"></td>
</tr>
<tr>
<td><input id="Sensor -" value="-" onclick="DesceSensor()" type="button"></td>
</tr>
</tbody>
</table>
<p>
</p>
<p>Executar a leitura dos sensores:
</p>
<table>
<tbody>
<tr>
<td><input id="executa" value="Ler sistema" onclick="executaLeitura()" type="button"></td>
</tr>
</tbody>
</table>
<p>
</p>
<p>
</p>
<table>
<tbody>
<tr>
<td><input id="geradoronoff" value="Acciona Gerador" onclick="acionaFrecuencia()" type="button"></td>
<td>Gerador de sinais:</td>
</tr>
</tbody>
</table>
<p>
</p>
<div id="pwmstate" align="center"> - </div>
</td>
</tr>
</tbody>
</table>
<p>
</p>
<p>Selecionar frecuencia de leitura
</p>
<table>
<tbody>
<tr>
<td>Frecuencia actual</td>
</tr>
</tbody>
</table>
<div id="pwmfreq" align="center"> - </div>
</td>
</tr>
</tbody>
</table>
<td><input id="pwmfreqset" value="Atualiza frecuencia" onclick="pwm_freq_set()" type="button"></td>
<td><input id="pwmfreqset" type="text"></td>
</tr>
</tbody>
</table>
<table border="0" cellpadding="0" cellspacing="0" width="100%">
<tbody>
<tr>
<td align="right" valign="center">
</td>
</tr>

```

</t1>

</td>

</td>

</tr>

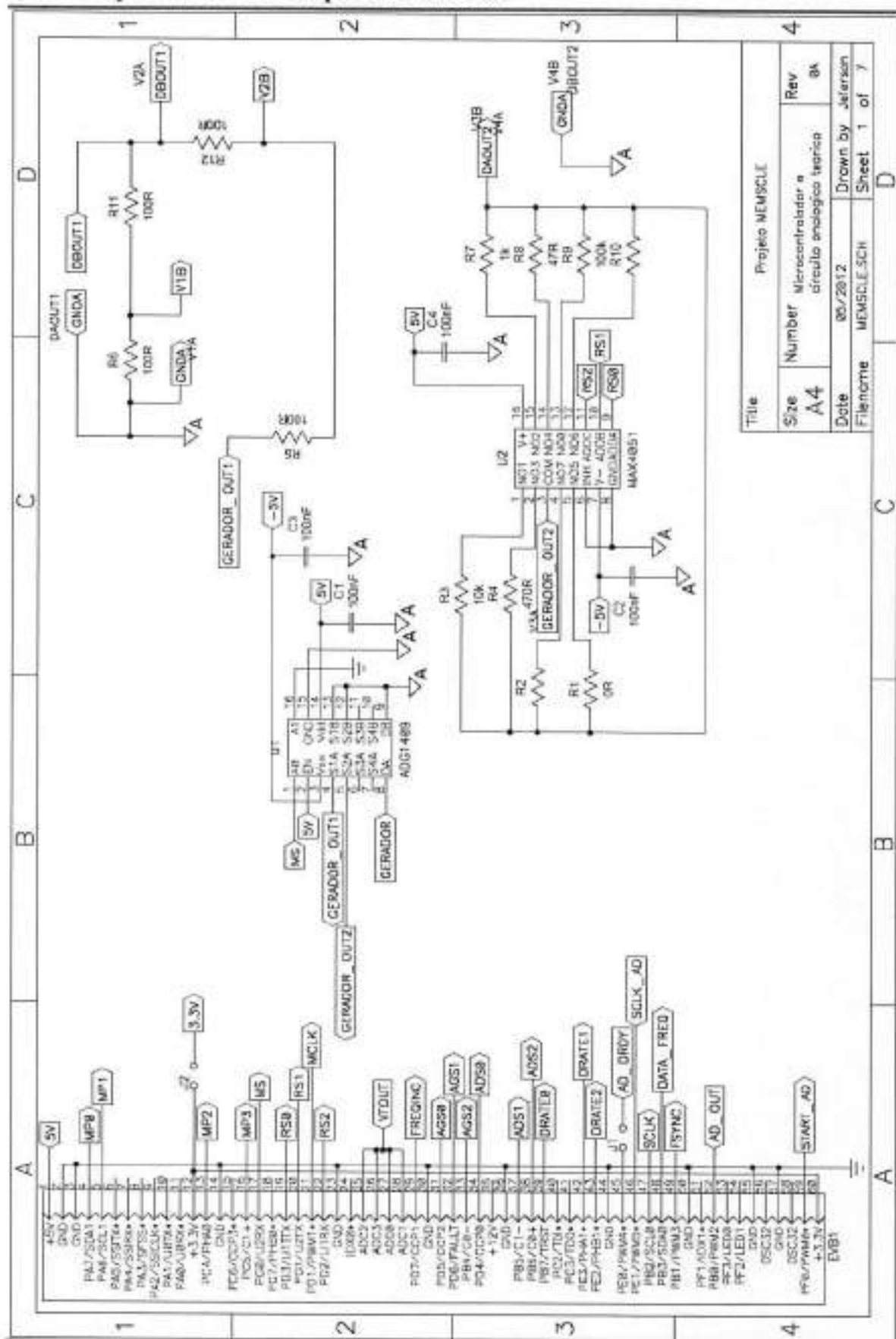
</tbody>

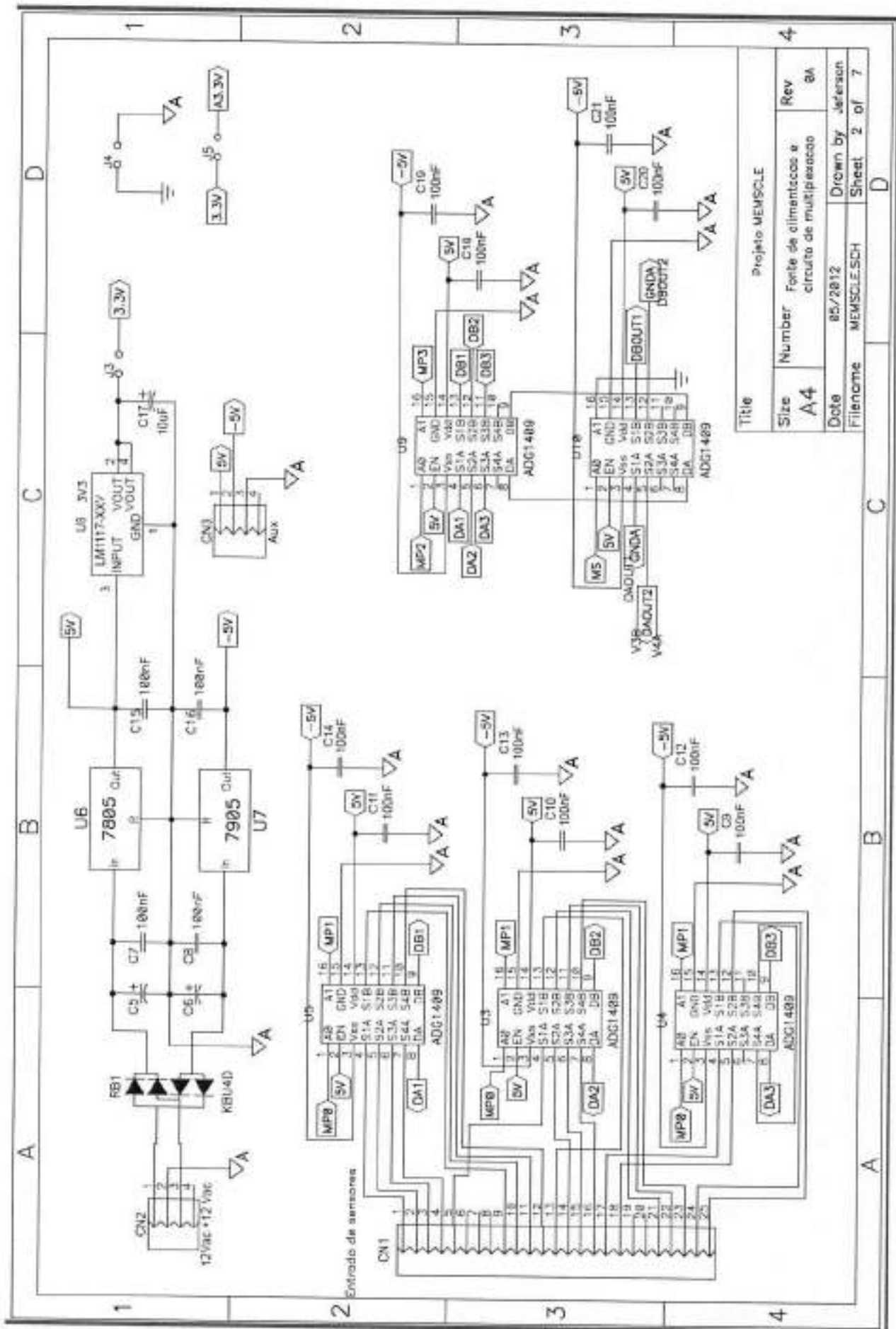
</table>

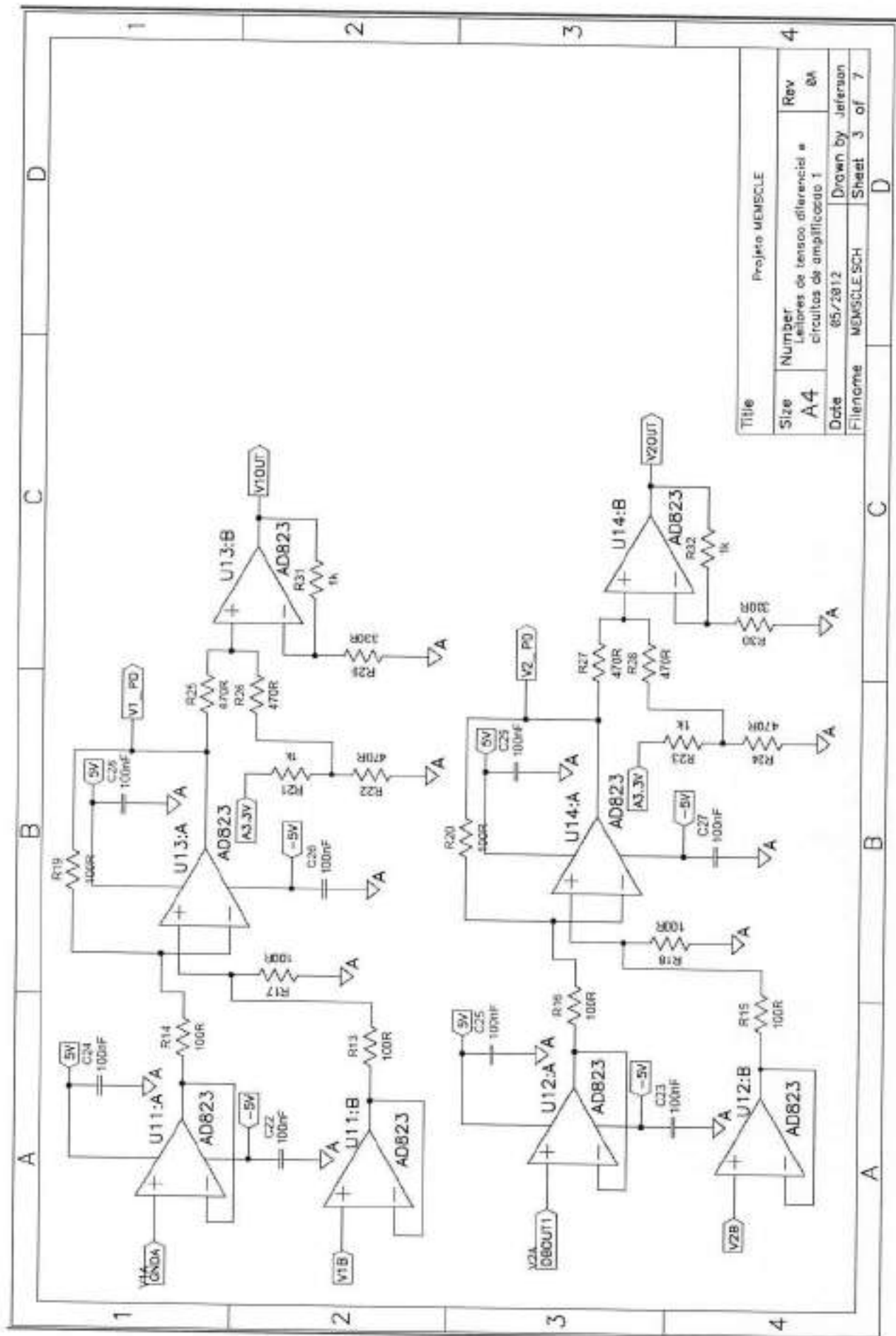
</body></html>

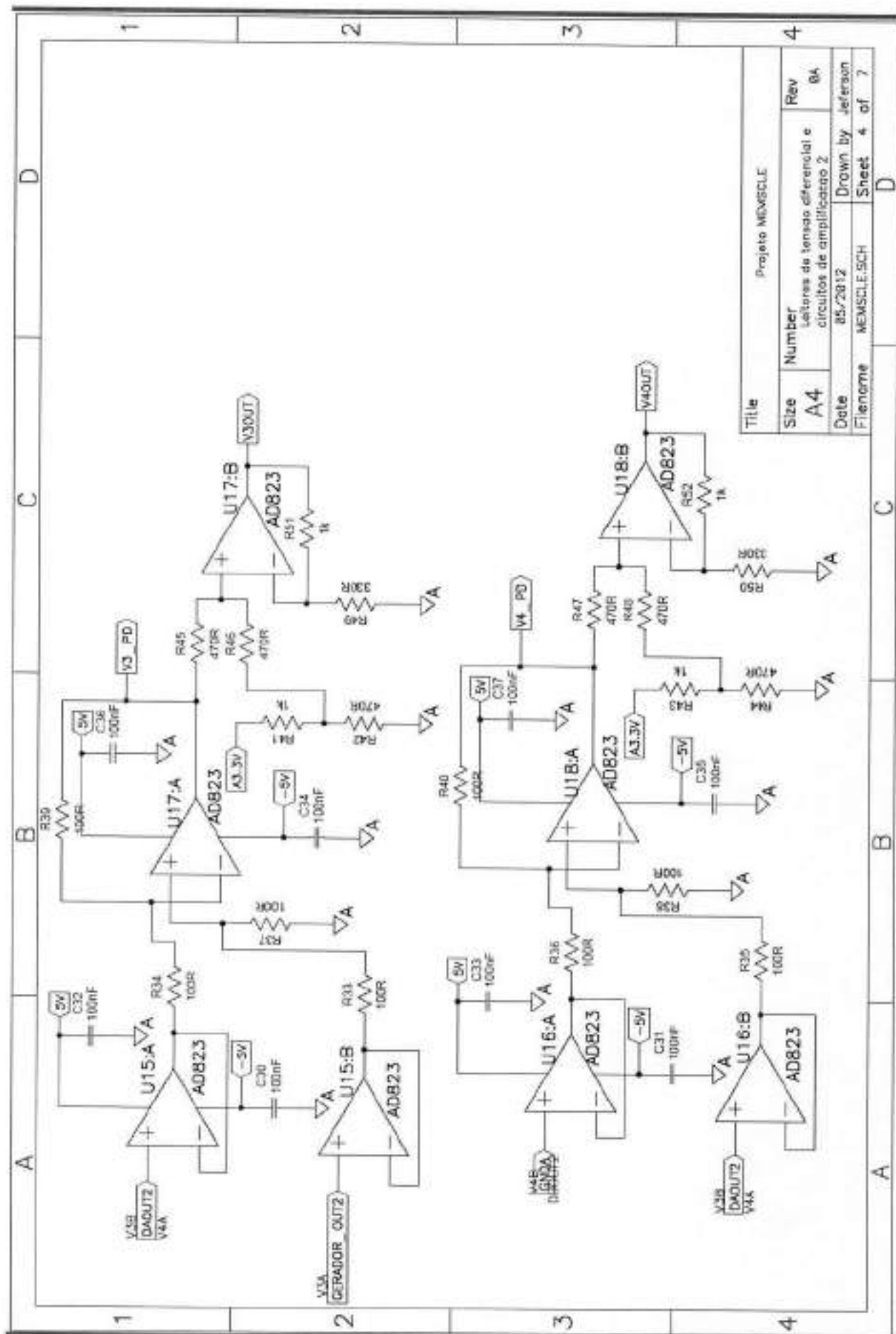


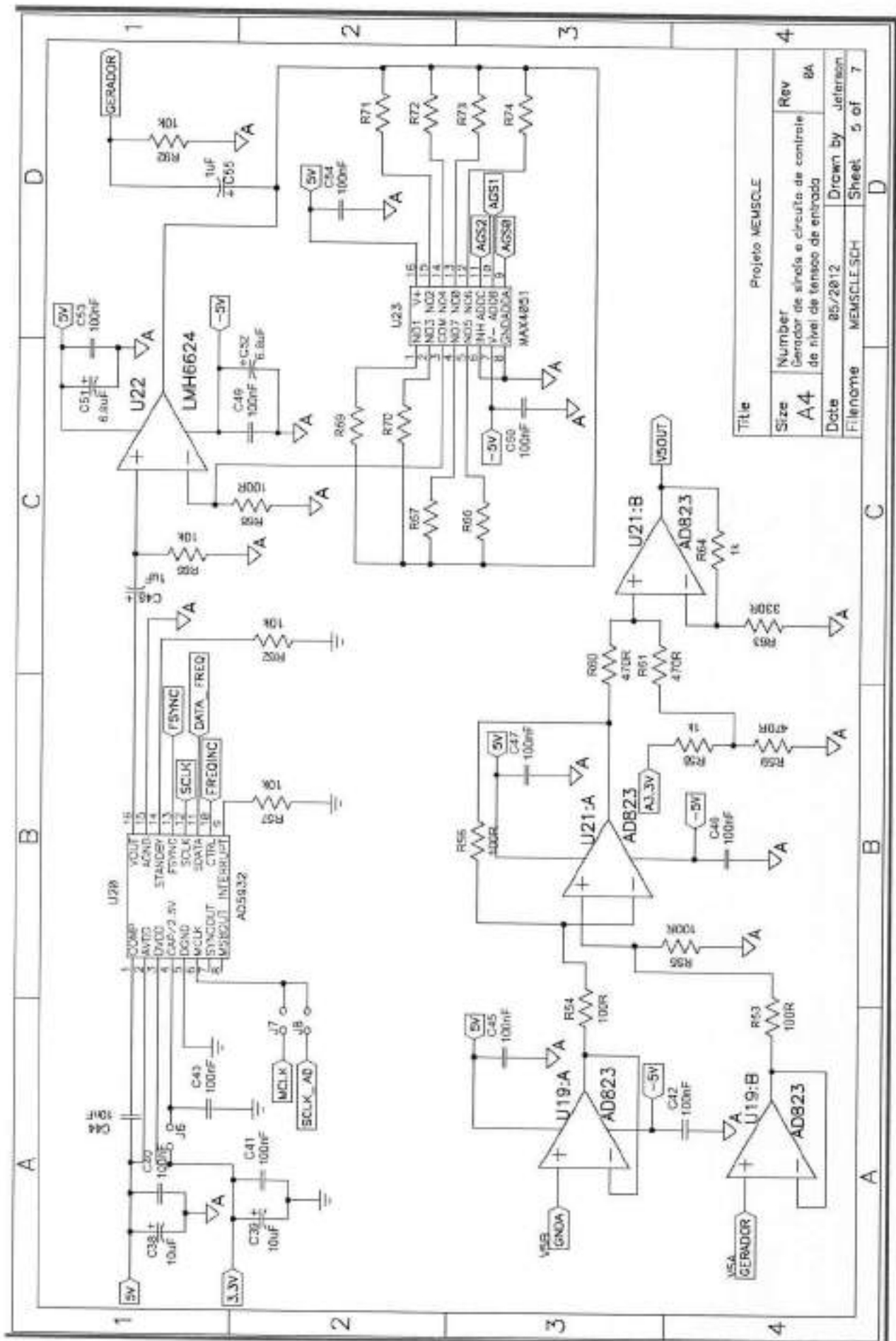
## Anexo C: Esquema técnico completo do sistema

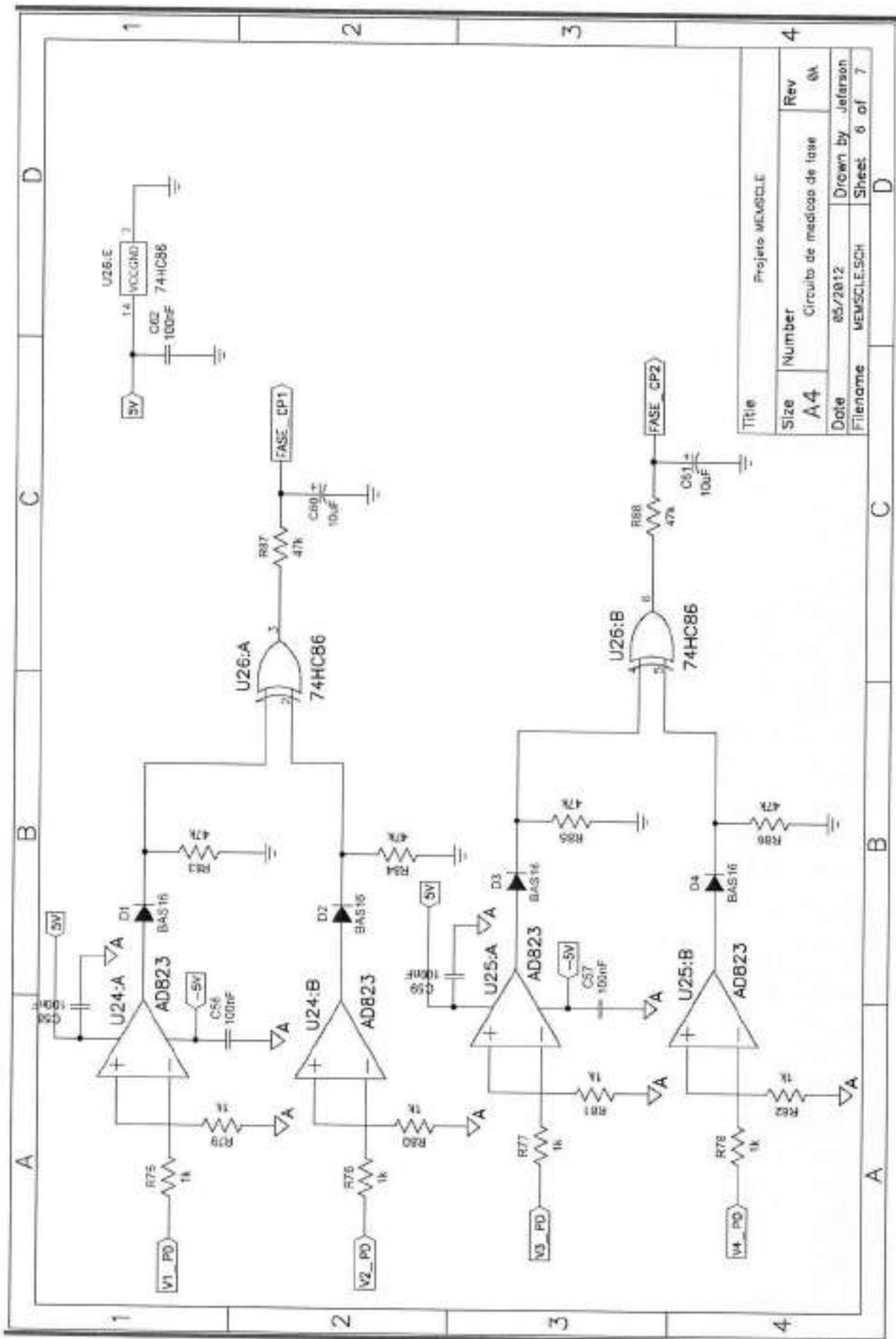










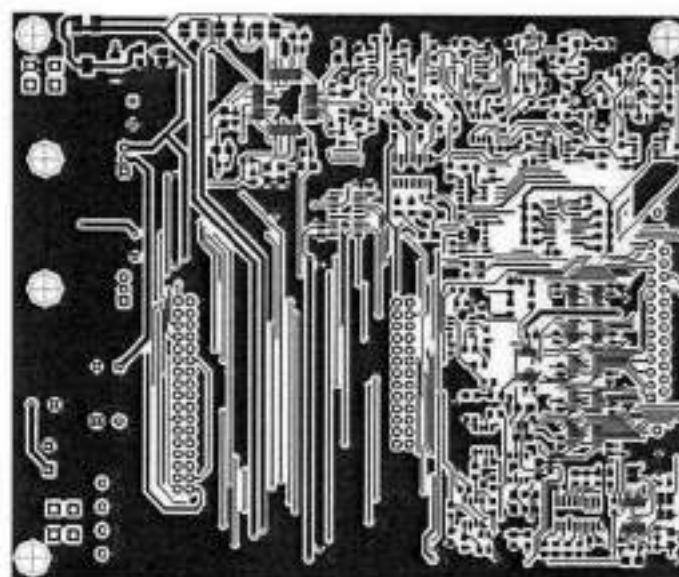




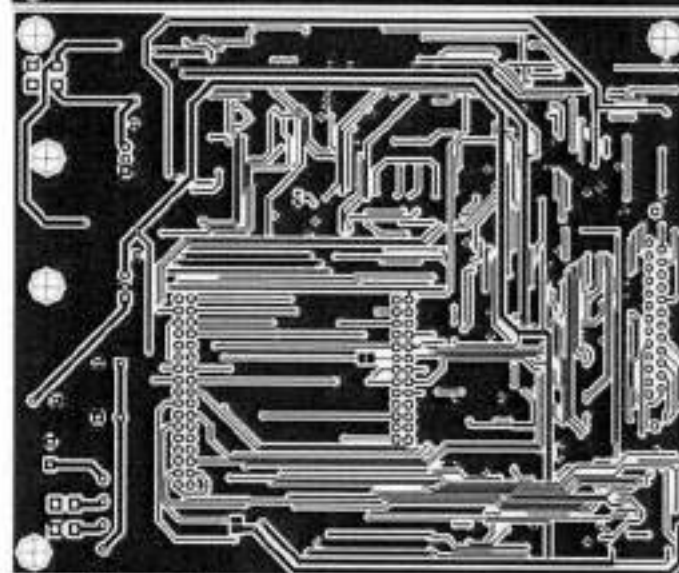


**Anexo D: Projeto da placa de circuito impresso**

Layer Top



Layer Bot



Layer Tsi

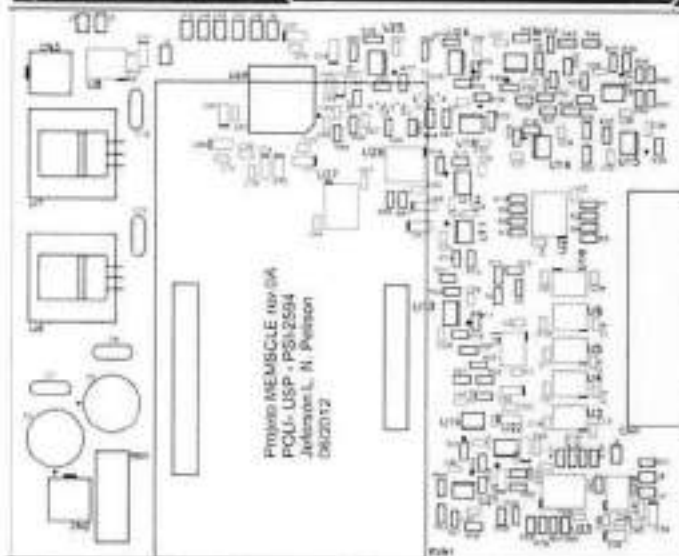


Figura 27: Layers da PCI

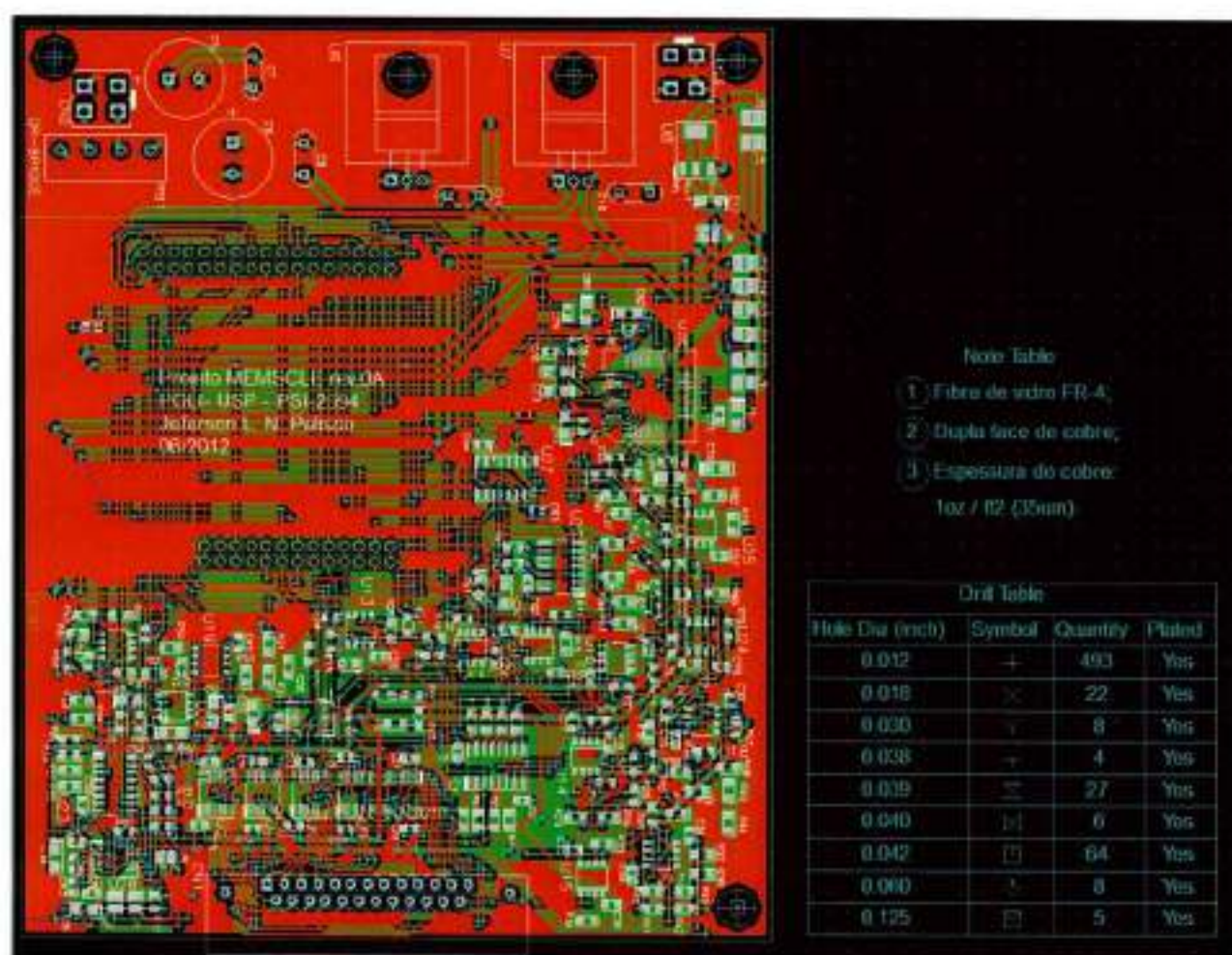


Figura 28: PCI no software de projeto