



Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação

# **Ferramentas Aplicadas no Desenvolvimento de Laboratório Remoto e/ou Presencial no Ensino de Engenharia Eletrônica.**

**Autor: André Ricardo Marchezan**  
**Orientador: Prof. Dr. Elnatan Chagas Ferreira**

**Dissertação de Mestrado** apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Eletrônica, Optoeletrônica e Microeletrônica.**

Banca Examinadora  
**Elnatan Chagas Ferreira, Dr.**  
Wilmar Bueno de Moraes, Dr.  
Eurípides Guilherme de O. Nóbrega, Dr.  
Luis Otavio Saraiva Ferreira, Dr.

FEEC/Unicamp  
CENPRA/Campinas  
FEM/Unicamp  
FEM/Unicamp

Campinas, SP  
Agosto/2006



# Dedicatória

A minha esposa Karen,  
aos meus avós Osvaldo e Duília  
e meu filho Ricardo.

# Agradecimentos

Ao meu orientador, Professor Doutor Elnatan Chagas Ferreira, sou grato pela orientação e paciência.

Aos colegas Marco Chella e Fernando Melo, pelo grande apoio no trabalho e força nos momentos difíceis.

A minha família pelo apoio durante esta jornada.

A CAPES, pelo apoio financeiro.

# RESUMO

Nesta dissertação são apresentados os resultados obtidos na elaboração de um sistema modular de desenvolvimento de experimentos na área de ensino de eletrônica através de uma rede de computadores.

É uma proposta de infra-estrutura baseada em instrumentação remota, com intuito do desenvolvimento de um laboratório remoto multi-usuário com acesso presencial ou via rede Internet, utilizando uma interface gráfica amigável e intuitiva.

São discutidas ferramentas e formas de integração entre hardware e software para atender de uma forma eficiente e prática o controle dos equipamentos remotos, utilizando para a validação uma experiência real do laboratório da UNICAMP.

A proposta global visa contribuir para o desenvolvimento de uma plataforma de experimentos remotos, com situações reais, na área de engenharia de uma forma mais simples, com custo reduzido e atendendo o maior número de alunos.

**Palavras-chaves** : laboratório remoto, instrumentação baseada em computador, ensino a distancia

# ABSTRACT

This paper presents the results obtained from the elaboration of a modular system used for conducting experiments on electronics teaching via a computer network.

This is an infrastructure proposal based on remote instrumentation and its main goal is developing a multi-user remote laboratory whose access occurs via local and/or Internet, with a friendly interface and aiming to conduct/develop laboratorial experiments in a generic way.

The objective is to discuss the several tools and integration forms between hardware and software, so as to offer an effective and practical control of the equipment, using for the validation a real experience of the laboratory at UNICAMP.

The global proposal intend to contribute to the development of a remote platform of experiments, presenting real engineering situations in a simple and accessible way and for the largest possible number of students.

**Keywords** : remote laboratory, instrumentation based on computer, distance learning

# SUMÁRIO

<b>Lista de Figuras .....</b>	<b>IV</b>
<b>Glossário .....</b>	<b>VII</b>
<b>Trabalhos Acadêmicos do Autor.....</b>	<b>IXIII</b>
<b>Objetivo .....</b>	<b>XIV</b>
<b>Organização do Trabalho .....</b>	<b>XVI</b>
<b>CAPÍTULO 1 - INTRODUÇÃO.....</b>	<b>1</b>
<b>CAPÍTULO 2 - O ESTADO DA ARTE .....</b>	<b>5</b>
<b>CAPÍTULO 3 - ESTRUTURA GERAL E FUNCIONALIDADE DO HARDWARE .</b>	<b>15</b>
<b>3.1 - Introdução.....</b>	<b>15</b>
<b>3.2 – O Hardware de Controle Geral - PLCON .....</b>	<b>17</b>
<b>3.2.1 – Estrutura Geral .....</b>	<b>19</b>
<b>3.2.2 – A USB (Universal Serial Bus).....</b>	<b>20</b>
<b>3.2.3 A USB e o FT245BM .....</b>	<b>23</b>
<b>3.2.4 – Microcontrolador PIC 16f877 .....</b>	<b>25</b>
<b>3.2.5 – Barramento Diferencial .....</b>	<b>26</b>
<b>3.3 – Sistema de Controle Local - PLEXP.....</b>	<b>27</b>
<b>3.3.1 – Estrutura Geral .....</b>	<b>30</b>
<b>3.3.2 – O Shift Register (SR).....</b>	<b>30</b>
<b>3.3.2 – Seleção da PLEXP .....</b>	<b>32</b>
<b>3.3.3 – Seleção do Ponto de Medição.....</b>	<b>33</b>
<b>3.3.4 – Acionando Chaves .....</b>	<b>35</b>
<b>3.3.5 – Seleção de Resistores .....</b>	<b>35</b>

---

3.3.6 – Atuando nos Potenciômetros - Protocolo SPI.....	36
3.3.7 – Circuito da Experiência .....	38
3.4 – Blocos Funcionais .....	39
3.4.1 – Bloco Chave – BLCH .....	41
3.4.2 – Bloco Lista de Resistores – BLRES .....	42
3.4.3 – Bloco Potenciometro Digital – BLPOT .....	42
3.4.4 – Novos Instrumentos USB.....	45
3.5– O Barramento de Dados e Alimentação .....	45
<b>CAPÍTULO 4 - ASPECTOS DE SOFTWARE .....</b>	<b>47</b>
4.1 – Trabalhando COM A Dll - Host .....	48
4.2 – O Protocolo de Comunicação - PIC.....	51
4.3 – Testando com Visual Basic .....	54
<b>CAPÍTULO 5 - RESULTADOS.....</b>	<b>57</b>
5.1 – Aplicação Experimental.....	63
5.2 – Objetivo do Experimento.....	63
5.3 - Material Necessário.....	63
5.4 - Roteiro da Experiência e Resultados Obtidos .....	64
<b>CAPÍTULO 6 - CONCLUSÃO .....</b>	<b>79</b>
Sugestões para Trabalhos Futuros.....	82
Referências Bibliográficas .....	85
<b>APÊNDICE A - ESQUEMÁTICO E LAYOUT DA PLCON.....</b>	<b>89</b>
<b>APÊNDICE B - ESQUEMÁTICO E LAYOUT DA PLEXP .....</b>	<b>90</b>

---

<b>APÊNDICE C - ROTEIRO DA EXPERIÊNCIA 4 – DISCIPLINA EE832 .....</b>	<b>91</b>
<b>APÊNDICE D - O SOFTWARE DO MICROCONTROLADOR (PIC) .....</b>	<b>114</b>
<b>APÊNDICE E - O SOFTWARE DO HOST (PC) .....</b>	<b>132</b>

---

## LISTA DE FIGURAS

<b>FIGURA 01 – Arquitetura do laboratório da Universidade Nacional de Singapura.....</b>	<b>6</b>
<b>FIGURA 02 – Exemplo de um dos laboratórios proposto pela Universidade de Singapura para o aprendizado do uso do osciloscópio.....</b>	<b>7</b>
<b>FIGURA 03 – Apresentação da estrutura do iLabs do MIT.....</b>	<b>7</b>
<b>FIGURA 04 - Exemplo de uma tela do ilabs.....</b>	<b>8</b>
<b>FIGURA 05 - Gráfico estatístico de avaliação do sistema iLabs do MIT .....</b>	<b>8</b>
<b>FIGURA 06 - Tela do laboratório ISILab da Universidade de Genova .....</b>	<b>9</b>
<b>FIGURA 07 – Arquitetura do Laboratório Remoto USP.....</b>	<b>10</b>
<b>FIGURA 08 – Laboratório Remoto da USP .....</b>	<b>11</b>
<b>FIGURA 09 – Propaganda do NI ELVIS da National Instruments.....</b>	<b>12</b>
<b>FIGURA 10 – Modelo da Matriz apresentada pela Unicamp para rotear circuitos eletrônicos.....</b>	<b>13</b>
<b>FIGURA 11 – Tabela Comparativa das Características dos Laboratórios Remotos/Virtuais com a nova proposta (UNICAMP 2).....</b>	<b>14</b>
<b>FIGURA 12 – Arquitetura Geral do Laboratório Proposto, com as Placas de Controle, Barramento e Blocos Funcionais.....</b>	<b>16</b>
<b>FIGURA 13 – Placa de Controle (PLCON) .....</b>	<b>18</b>
<b>FIGURA 14 – Tabela Resumo dos Subsistemas da Placa de Controle (PLCON).....</b>	<b>19</b>
<b>FIGURA 15 – Seqüência de eventos desde a conexão da placa até a inicialização dos drivers.....</b>	<b>22</b>
<b>FIGURA 16 – Esquema Físico de Conexão do Barramento .....</b>	<b>23</b>
<b>FIGURA 17 – Arquitetura do FT245BM.....</b>	<b>25</b>
<b>FIGURA 18 – Diagrama Esquemático da Placa de Experiência (PLEXP) .....</b>	<b>28</b>
<b>FIGURA 19 – Gabinete horizontal com as placas: experiência e controle .....</b>	<b>29</b>
<b>FIGURA 20 – Pin-Out do Shift Register com Entradas e Saídas (DIP) .....</b>	<b>31</b>

---

<b>FIGURA 21 – Tabela Verdade do Componente - 4094 .....</b>	<b>31</b>
<b>FIGURA 22 – Trecho do Esquemático de Seleção da PLEXP.....</b>	<b>33</b>
<b>FIGURA 23 – Entradas, Saídas e Chaves (7501) das Medições (PLEXP).....</b>	<b>34</b>
<b>FIGURA 24 – Circuito das Chaves e Conexão (PLEXP) .....</b>	<b>35</b>
<b>FIGURA 25 – Circuito do Bloco - Lista de Resistores (BLRES) .....</b>	<b>36</b>
<b>FIGURA 26 – Conectores de Conexão para os Blocos Potenciômetros (BLOSC) no Circuito Fixo da Placa da Experiência (PLEXP) .....</b>	<b>37</b>
<b>FIGURA 27 – A Placa de Experiência (PLEXP).....</b>	<b>38</b>
<b>FIGURA 28 – A Placa de Experiência Completa (PLEXP) .....</b>	<b>40</b>
<b>FIGURA 29 – Bloco Chave - BLCH .....</b>	<b>41</b>
<b>FIGURA 30 – Diagrama Funcional do BLPOT .....</b>	<b>43</b>
<b>FIGURA 31 – Tabela de Descrição dos Pinos BLPOT .....</b>	<b>43</b>
<b>FIGURA 32 – A Placa do BLPOT (AD7376).....</b>	<b>44</b>
<b>FIGURA 33 – Placa e PCB do Barramento .....</b>	<b>46</b>
<b>FIGURA 34 – Arquitetura do Software .....</b>	<b>48</b>
<b>FIGURA 35 – Arquitetura do FT2XX .....</b>	<b>51</b>
<b>FIGURA 36 – Esquemático Realizada no Laboratório da UNICAMP.....</b>	<b>57</b>
<b>FIGURA 37 – Detalhe dos Conectores dos Módulos – BLCH .....</b>	<b>58</b>
<b>FIGURA 38 – Placa de Experiência (PLEXP) com a experiência 4 implementada .....</b>	<b>59</b>
<b>FIGURA 39 – Software de Controle Implementado em Visual Basic para Realização da Experiência Proposta .....</b>	<b>60</b>
<b>FIGURA 40 – Detalhe da Interface Gráfica com Chave Aberta .....</b>	<b>61</b>
<b>FIGURA 41 – Detalhe da Interface Gráfica com Chave Ligada .....</b>	<b>61</b>
<b>FIGURA 42 – Detalhe da Interface Gráfica com Potenciômetro .....</b>	<b>61</b>
<b>FIGURA 43 – Osciloscópio Desenvolvido em Visual Basic Conectado via GPIB.....</b>	<b>62</b>
<b>FIGURA 44 – Configuração do Esquemático para Análise do Oscilador de Onda Quadrada.....</b>	<b>65</b>
<b>FIGURA 45 – Resultado obtido com o Potenciômetro (BLPOT1) em VCC .....</b>	<b>66</b>
<b>FIGURA 46 – Resultado obtido com o Potenciômetro (BLPOT1) em GND.....</b>	<b>67</b>

---

<b>FIGURA 47 – Resultado obtido em relação ao tempo de subida.....</b>	<b>68</b>
<b>FIGURA 48 – Resultado obtido em relação ao tempo de descida .....</b>	<b>69</b>
<b>FIGURA 49 – Configuração do Esquemático Solicitado pelo item 5.1.3.21b.....</b>	<b>70</b>
<b>FIGURA 50 – Resultado Obtido utilizando o Diodo 1N5404 (CH 9) .....</b>	<b>71</b>
<b>FIGURA 51 – Resultado Obtido na medição de corrente sobre o Diodo 1N5404.....</b>	<b>72</b>
<b>FIGURA 52 – Resultado Obtido na medição de corrente sobre o Diodo 1N5404 com aterramento .....</b>	<b>73</b>
<b>FIGURA 53 – Configuração do Esquemático Solicitado pelo item 5.1.3.3a .....</b>	<b>75</b>
<b>FIGURA 54 – Resultado Obtido com CH3 na posição Transistor, sem CH4 .....</b>	<b>76</b>
<b>FIGURA 55 – Resultado obtido com a resistência máxima (TIP 54).....</b>	<b>77</b>
<b>FIGURA 56 – Arquitetura Completa do Sistema do Laboratório Remoto.....</b>	<b>80</b>
<b>FIGURA 57 – Arquitetura de Comunicação do Novo Módulo Proposto .....</b>	<b>83</b>
<b>FIGURA 58 – Esquemático da Nova Proposta Sugerida .....</b>	<b>84</b>

---

## GLOSSÁRIO

**RS-232** (Recommended Standard-232) - padrão de interface aprovado pela EIA (Electronic Industries Association) para conexão de dispositivos com comunicação serial.

**USB** (Universal Serial Bus) - padrão de barramento que suporta taxas de transferência de até 12 Mbps. Uma única porta suporta a conexão de até 127 periféricos com suporte a Plug-and-Play.

**LED** (Light Emitting Diode) - Diodo Emissor de Luz

**SMD** (Surface Mount Device) - Dispositivo de Montagem em Superfície

**DAQ** – Placas de Aquisição de Dados

**GPIB** (General-Purpose Interface Bus) – Protocolo de comunicação desenvolvido pela Hewlett-Packard comum na área de instrumentação

**iLabs** – Laboratório Remoto desenvolvido pelo MIT (Massachusetts Institute of Technology)

**ISILab** - Laboratório Remoto desenvolvido pela Universidade de Genova

**ELVIS** - Laboratório Remoto desenvolvido pela National Instruments (NI)

**PLCON** – Placa de Controle Geral

**PLBUS** – Placa de Barramento Geral

**PLEXP** – Placa de Controle Local (Placa de Experiência)

**BLOSC** – Bloco Funcional Osciloscópio

**BLRES** – Bloco Funcional Almojarifado de Resistores

**BLCH** – Bloco Funcional Chave

**BLPOT** – Bloco funcional Potenciômetro

**VB** – Visual Basic

**DEMIC** – Departamento de Microeletrônica da UNICAMP

**Mbps** – Velocidade de Transmissão de Dados - Mega bits por segundo

**PIC** – Nomenclatura dos Microcontroladores da MICROCHIP

**SPI** (Serial Peripheral Interface) – Protocolo de Comunicação desenvolvido pela Motorola (3 fios)

**WEB** (World-Wide Web) - Interface Gráfica para Internet

**DLL** (Dynamic Link Libraries) - conjunto de funções autônomas que podem ser utilizados por qualquer aplicação no ambiente Windows

**FOB** (Free On Board) - Preço sem Frete Incluso (posto a bordo). Denominação da cláusula de contrato segundo a qual o frete não está incluído no custo da mercadoria.

## TRABALHOS ACADÊMICOS DO AUTOR

Revista Brasileira de Informática na Educação (RBIE) 2006 – Submissão do artigo “*Sistema para Desenvolvimento de Laboratório Remoto e/ou Presencial no ensino de Engenharia Eletrônica.*” – em análise.

WORKCOMP-SUL 2004 - I Workshop de Ciências da Computação e Sistemas da Informação da Região Sul - Palhoça – SC, "*Laboratório Remoto aplicado ao Ensino de Engenharia Eletrônica*"

XIV Simpósio Brasileiro de Informática na Educação – SBIE – NCE/UFRJ 2003 - “*Interface de Hardware para Instrumentação Baseada em Computador*”.

Future Technology Devices International Ltd. Scotland U.K - “*Working Dynamic Link Library (DLL) from LabVIEW for Control and Communication with FTDI Chip*”.

VIII Congresso de Iniciação Científica – 96 Unesp - Campus de Guaratinguetá – SP, "*Interface gráfica para ambiente inteligente de modelagem em redes*"

III Simpósio de Iniciação Científica em Ciências Matemáticas e da Computação CICTE - 95 USP - São Carlos - SP, "*Interface gráfica para ambiente inteligente de modelagem em redes*"

## OBJETIVO

Elaborar uma plataforma de suporte a multi-experimentos na área de eletrônica, onde seja possível realizar *experimentos remotos*\*\* distintos, implementados a partir de laboratório reais atendendo as seguintes características:

- *Custo baixo* e decrescente por estudante. O laboratório remoto tende a ficar mais econômico em relação ao convencional, pois pode atender a um número maior de alunos.
- *Modularidade*; permitindo que sejam agregados novos experimentos e equipamentos.
- Execução dos experimentos em “*tempo real*”, como por exemplo, o acionamento de chaves adicionando componentes nos circuitos e a utilização dos recursos do osciloscópio (*trigger, escala*).
- Permita a utilização dos procedimentos experimentais já existentes nos laboratórios convencionais, podendo facilmente integrar os recursos didáticos como livros-texto, ilustrações, entre outros meios.
- Aplicação no *modo remoto* (rede) e *presencial* (local).
- *Acesso compartilhado* a equipamentos de custo alto, dispensando a aquisição por cada instituição ou laboratório.
- *Flexibilização de horários* para execução dos experimentos, otimizando o tempo de estudo dos alunos e a utilização dos equipamentos [5].

Para alcançar este objetivo será necessário desenvolver um conjunto de ferramentas (software/hardware) integrados, empregando tecnologias disponíveis comercialmente no momento, observando as tendências futuras.

\*\* *Laboratório Remoto refere-se neste trabalho a laboratórios reais automatizados que podem ser acessados localmente, guiados pelo instrutor ou disponibilizados na Internet.*

## ORGANIZAÇÃO DO TRABALHO

No desenvolvimento desta dissertação de mestrado serão apresentados conceitos teóricos relacionados ao ambiente de laboratório remoto aplicado no ensino, os aspectos técnicos relacionados à implementação, resultados práticos obtidos e novas propostas de trabalho.

Os capítulos do trabalho estão na seqüência citada acima, organizados da seguinte forma:

**Capítulo 1** – Apresenta, de uma forma geral, parte dos conceitos teóricos envolvidos neste assunto. Apresenta também características de cada sistema (remoto e virtual). Experiências de sua inserção em ambientes de ensino/aprendizagem.

**Capítulo 2** – Discute e apresenta as diversas propostas sobre o tema da dissertação em relação a outras instituições de ensino e empresas interessadas no assunto. Apresentando características de cada sistema de uma forma comparativa.

**Capítulo 3** – Expõe o funcionamento do sistema de uma forma ampla e as tecnologias utilizadas na implementação. Apresenta, detalhadamente, cada componente integrante do subsistema, os hardwares, softwares e seus desenvolvimentos.

**Capítulo 4** – Descreve todo o sistema em relação ao desenvolvimento dos softwares de comando e controle do projeto. Arquiteturas e conexões.

**Capítulo 5** – Apresenta os resultados obtidos na realização de uma experiência real, a avaliação do sistema em relação a melhorias, conclusão e as perspectivas para novas propostas de trabalho.

Referência Bibliográfica – Autores referenciados na dissertação, incluindo aplicativos e documentos eletrônicos da Internet.

Apêndices com os projetos eletrônicos, esquemáticos e listagem de programas utilizados no desenvolvimento do trabalho.

# CAPÍTULO 1

## INTRODUÇÃO

A formação do profissional de engenharia depende das aulas teóricas tradicionais, mas a experiência prática proporcionada pelos laboratórios é elemento vital [12].

Entretanto configurar e manter um laboratório de boa qualidade por longos períodos representa um custo financeiro alto [4].

Os Laboratórios acessíveis via Internet desenvolveram-se a partir da generalização do uso das novas tecnologias da informação e da necessidade de comunicação na educação.

A consolidação do uso da Internet como ferramenta de compartilhamento de informações, e o desenvolvimento de ferramentas modernas de controle, aquisição e distribuição de dados via rede de computadores vêm impulsionando o controle e o acompanhamento de ensaios experimentais remotos [16].

Há duas classes distintas de Laboratórios: Remotos e Virtuais. Ambos podem ser acessados, via Internet, mas apresentam diferenças básicas na estrutura e na forma como são implementados.

Um Laboratório Remoto existe fisicamente em algum lugar como um laboratório de eletrônica real, enquanto que o Laboratório Virtual é basicamente arquivos de um sistema de computação.

A estrutura básica de um *Laboratório Remoto* é formada por um conjunto de instrumentos interfaceados a um computador conectado à Web ou a uma rede interna. Este computador tem como função principal de controlar os equipamentos a ele interfaceados e poder ser acessado remotamente a partir de outros computadores ligados à Internet ou interligados a uma rede interna. O usuário remoto acessa e controla o computador do laboratório e, a partir daí, aciona equipamentos, faz observações, testa condições e coleta dados. Podem-se adicionar artifícios, como câmeras de vídeo, ao ambiente do laboratório de modo que o usuário consiga ter uma visão do que se passa no laboratório real. No caso particular do ensino de engenharia eletrônica é possível vivenciar a maioria dos problemas pertinentes ao laboratório real, como a variação da temperatura ao longo do tempo, problemas de conexão, componentes danificados, etc. Estes efeitos são de grande importância para a prática dos conceitos apresentados em sala de aula.

A tecnologia do laboratório remoto é muito extensa, podendo ser desde um simples laboratório, que apenas permite a realização de uma experiência, ou seqüência predeterminada de experimentos, até laboratórios remotos de alcance mundial, formados pela conexão de várias instalações menores.

Uma das grandes vantagens desta classe de laboratório reside no fato de que recursos instrumentais sofisticados, dedicados e de alto custo, geralmente não estão disponíveis para as pequenas instituições, e dessa forma podem ter seu uso compartilhado. Também é possível interligar vários laboratórios menores para a criação de outro maior, com mais recursos [2].

A estrutura de um Laboratório Virtual é formada por um conjunto de simulações, modelagens e visualizações. Eles não existem no mundo físico, real, e isto tem levado alguns educadores a relutar em classificá-los como laboratórios no sentido estrito do termo. No entanto, eles podem ser projetados para cumprir alguns dos pressupostos de um laboratório real, muito embora não consigam substituí-los como fonte de experiências reais. Essa não é, e nem deve ser, a intenção das pessoas que trabalham com esse tipo de recurso educacional.

O laboratório virtual é usado para enriquecer, complementar, a função do laboratório real, permitindo a realização de atividades que este último não tenha condições de suprir. Eles possibilitariam, por exemplo, a visualização de estruturas internas, inacessíveis, de equipamentos, ou a simulação de atividades onde o tempo e o custo de produção tornassem inviável a sua realização real. Poderiam, também, ser usados no treinamento de alunos para atividades envolvendo risco, como a realização de experiências com altas tensões ou o caminhar em determinadas áreas de uma central nuclear. Em um laboratório virtual, o aluno poderia provocar acidentes virtuais deliberadamente, manusear equipamentos nos limites de suas funções ou até extrapolar, conscientemente, esses limites, sem com isso correr o perigo de danificar um instrumento caro ou de atentar contra a sua própria segurança. Os laboratórios virtuais podem, ainda, ser usados como etapa preparatória para realização de atividades em laboratórios reais [6].

*\*\* Laboratório Remoto refere-se neste trabalho a laboratórios reais automatizados que podem ser acessados localmente, guiados pelo instrutor ou disponibilizados na Internet.*

## **CAPÍTULO 2**

### **O ESTADO DA ARTE**

O ensino tradicional envolve duas etapas: conceitos e prática, onde os conceitos são transmitidos em aulas expositivas e a prática através de laboratórios, onde é possível verificar a simulação dos conceitos assimilados [16].

Existe uma dificuldade particular no ensino de engenharia elétrica e eletrônica pelo tempo necessário para elaborar, montar e testar experimentos [10], ou seja, há uma grande dificuldade em colocar em prática a teoria. Uma das soluções amplamente divulgadas trata-se da disponibilização de instrumentos reais para acesso remoto a alunos geograficamente distantes do aparato experimental.

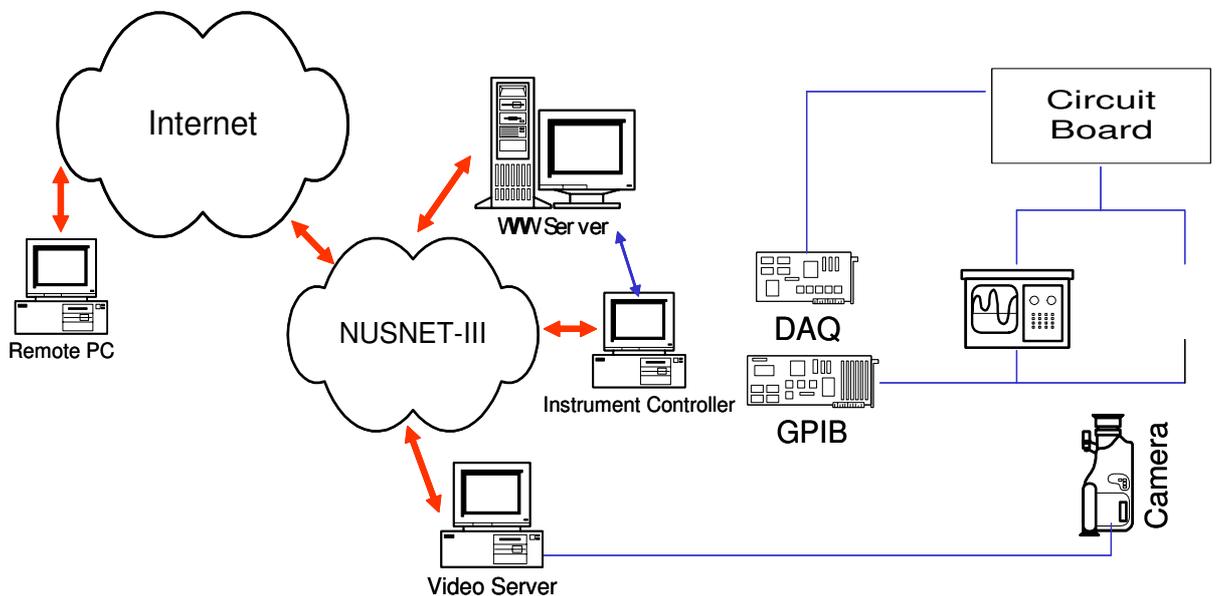
O laboratório remoto pretende melhorar a qualidade do ensino, incentivando o uso de novas tecnologias [29], atendendo um maior número de alunos em aulas práticas.

Com as novas ferramentas disponíveis no mercado (software/hardware) pode-se destacar uma nova forma de ensino [6] envolvendo o uso de redes de computadores, ensaios experimentais monitorados via computador, disponibilizando as atividades experimentais aos estudantes localizados à distância dos laboratórios convencionais.

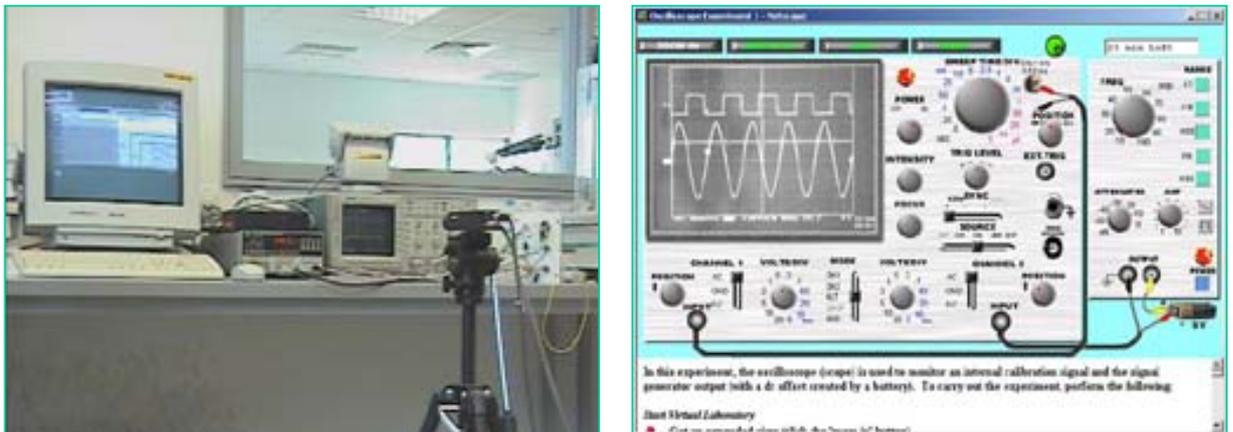
O desenvolvimento e uso de laboratórios virtuais onde estudantes e pesquisadores controlam instrumentos particulares de forma remota para o aprendizado ou pesquisa mostra-se como uma atividade promissora que integra diferentes tecnologias. Já existem diversos laboratórios remotos e virtuais no mundo [16].

Na Universidade Nacional de Singapura (<http://vlab.ee.nus.edu.sg/index.html>) há diversos experimentos laboratoriais disponíveis para realização remota. São experimentos simples, mas de grande valia no aprendizado.

Utilizam os recursos do software LabView, placas de aquisição (GPIB e DAQ's) e câmeras para simular o ambiente real. Os experimentos têm configurações fixas, ou seja, não permitem a flexibilidade para criação de novos experimentos. Cada experimento é um novo projeto.

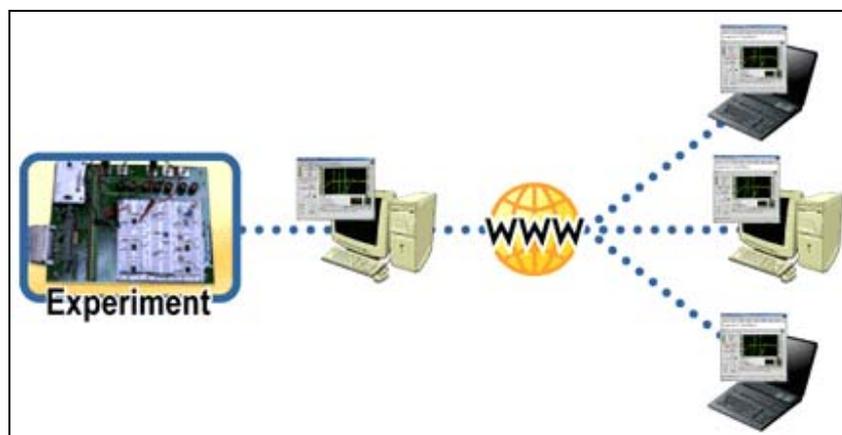


**FIGURA 01 – Arquitetura do laboratório da Universidade Nacional de Singapura.**



**FIGURA 02 – Exemplo de um dos laboratórios proposto pela Universidade de Singapura para o aprendizado do uso do osciloscópio.**

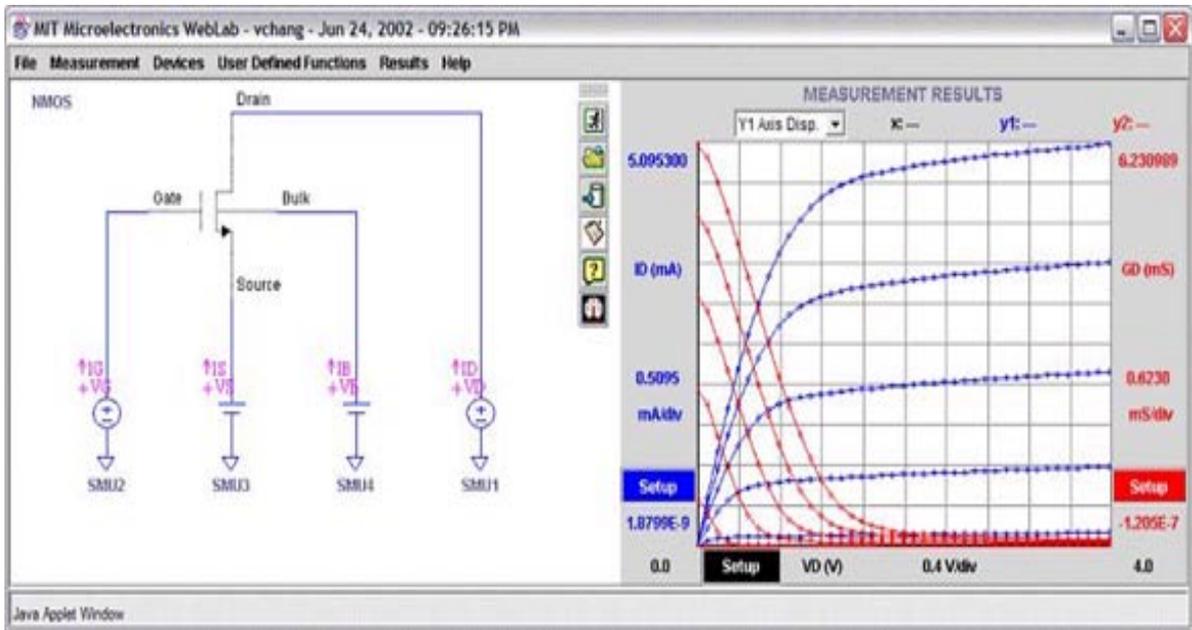
O laboratório Remoto do Massachusetts Institute of Technology – MIT (<http://icampus.mit.edu/iLabs/default.aspx>) foi batizado de **iLabs**, com a seguinte nota : “ Internet access to real labs - anywhere, anytime”.



**FIGURA 03 – Apresentação da estrutura do iLabs do MIT**

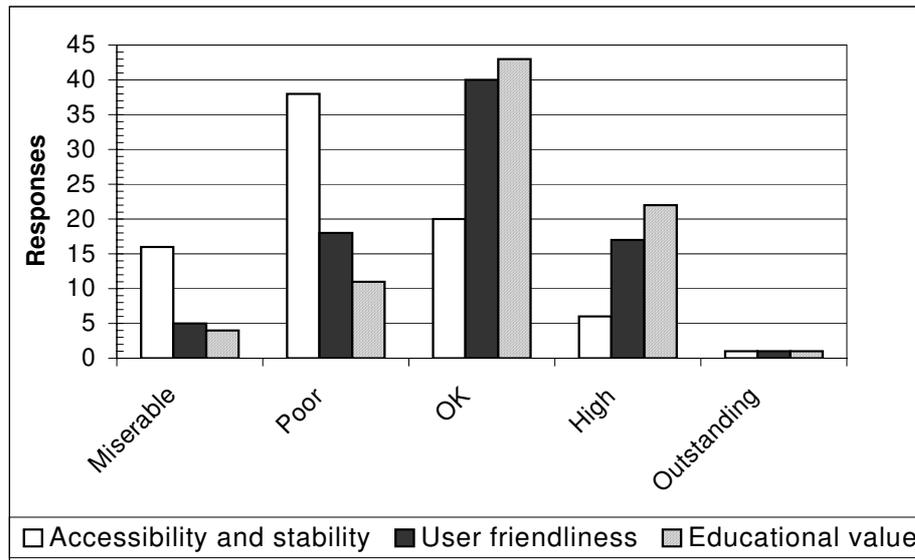
O **iLabs** é uma estrutura pré-estabelecida que permite, através de métodos de elaboração, gerar laboratórios remotos em diversas áreas, compartilhando equipamentos de custo elevado com outras universidades do Mundo.

Atualmente, o **iLabs** tem criado laboratórios remotos nas áreas de microeletrônica, química, engenharia estrutural e processamento de sinais.



**FIGURA 04 - Exemplo de uma tela do ilabs**

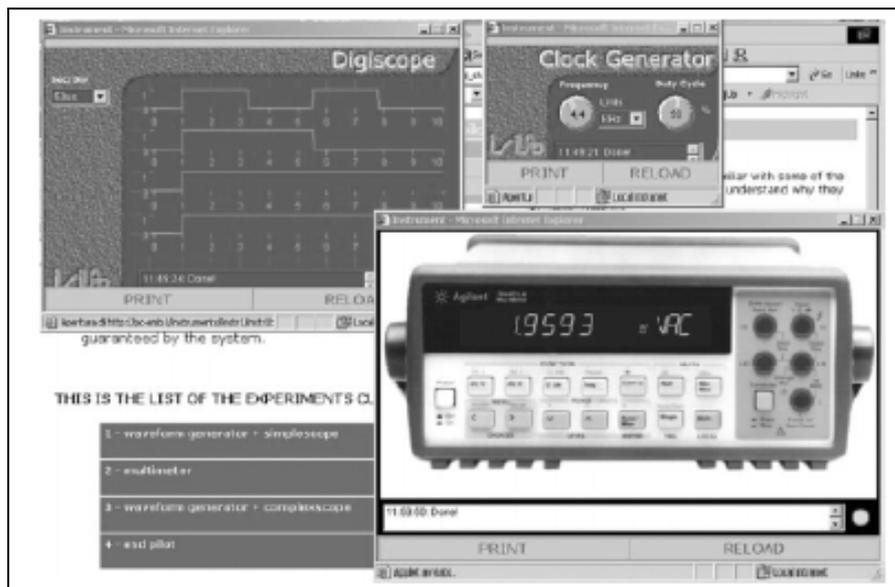
O resultado sobre a qualidade e funcionalidade do sistema **iLabs** pode ser verificado na FIGURA 05.



**FIGURA 05 - Gráfico estatístico de avaliação do sistema iLabs do MIT**

Na Universidade de Genova, Itália (<http://www.unige.it/>) foi criado o **ISILab**, onde basicamente temos dois modelos possíveis de experimentos remotos para o ensino de eletrônica. O primeiro chamado de dirigido, o professor realiza a experiência e os alunos ou interessados acompanham, este formato é muito ágil e útil para apresentar a prática a um grupo grande de alunos. O outro é chamado de modo independente, onde todos os usuários podem acessar e interagir com os experimentos, verificando os resultados [2].

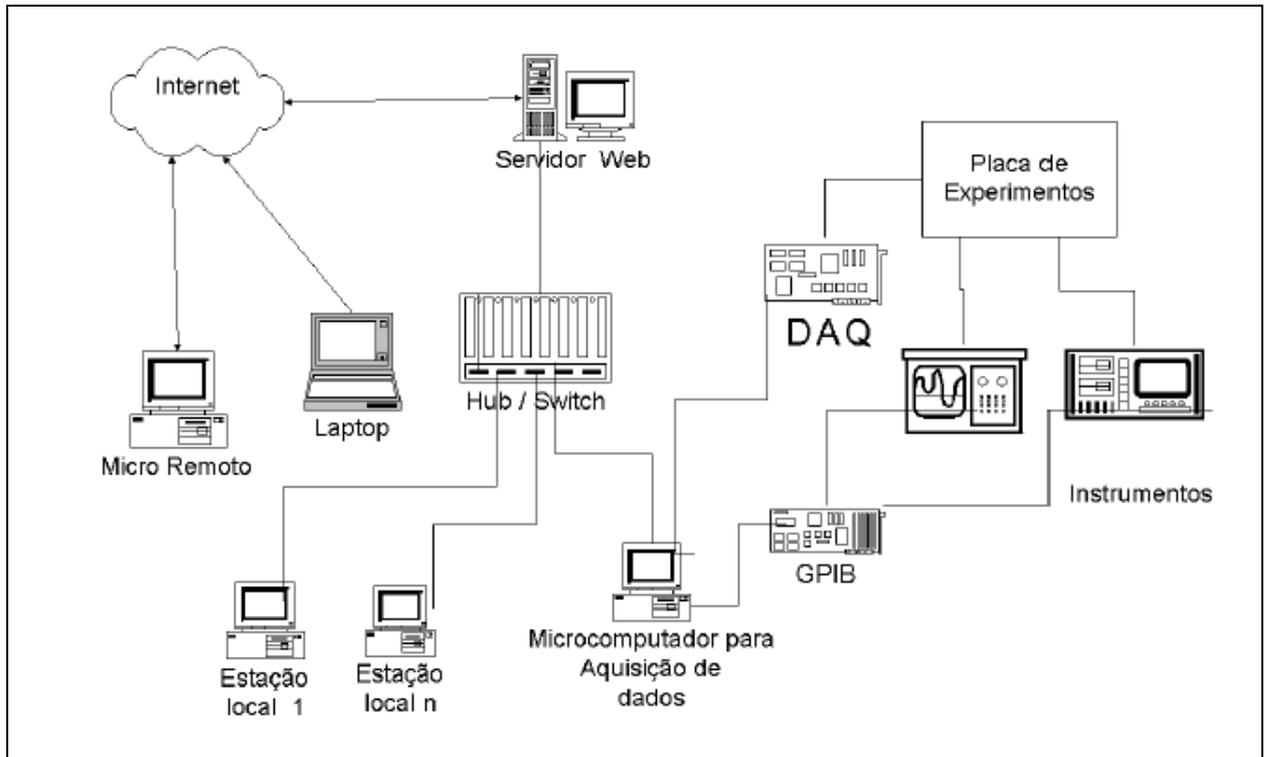
Outra característica do **ISILab** é sua modularidade e estrutura escalar, permitindo assim incorporar outras experiências através do desenvolvimento de placas dedicadas ao sistema desenvolvido.



**FIGURA 06 - Tela do laboratório ISILab da Universidade de Genova**

Na USP (SIM/Microeletrônica) o Laboratório Remoto em desenvolvimento permite o acompanhamento e o controle de experimentos remotos a partir de um computador que possua acesso à Internet, placas de aquisição de dados (DAQs) e placas de interface GPIB.

Seu funcionamento consiste na troca de dados entre este computador localizado no laboratório (Servidor), e os computadores de acesso remoto (Clientes), de modo que estes possam monitorar e controlar os experimentos realizados no laboratório, via Internet, em “tempo real”, mostrado na FIGURA 07



**FIGURA 07 – Arquitetura do Laboratório Remoto USP**

Para aumentar a interatividade entre o usuário remoto e a experiência local, utilizaram as entradas e saídas digitais das DAQs, comutando relés, possibilitando inserir ou remover alguns componentes, tais como resistores, capacitores e diodos, a fim de mudar a configuração dos circuitos das experiências. Atualmente este sistema permite acesso a cinco diferentes experiências: diodo Zener; retificadores; transistor bipolar; transistor MOS e Oscilador Astável.

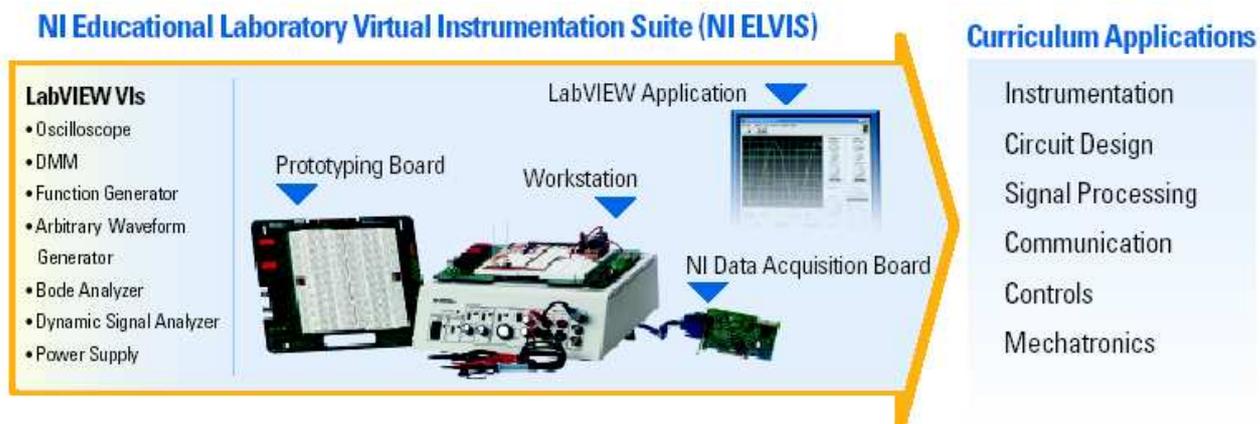
Estas experiências foram escolhidas porque demonstram os conceitos importantes apresentados na maioria de cursos introdutórios da eletrônica. O servidor foi desenvolvido em LabVIEW 6i em um ambiente utilizando o Sistema Operacional Windows.



**FIGURA 08 – Laboratório Remoto da USP**

O laboratório apresentado pela USP tem um custo elevado devidos às placas de aquisição DAQ's e instrumentos GPIB. Elaborar experimentos de uma forma dedicada reduz consideravelmente o custo e não compromete a flexibilidade do sistema.

A National Instruments lançou no mercado um kit laboratório chamado NI ELVIS baseado no software LabVIEW, com dispositivo de aquisição de dados multifunção e estação de trabalho com protoboard. Este produto pode através de desenvolvimento via software disponibilizar dados via Internet.



**FIGURA 09 – Propaganda do NI ELVIS da National Instruments**

A estação de trabalho tem vários componentes: 01 osciloscópio com 02 canais (500 kHz), 01 multímetro completo, gerador de funções (5 Hz a 250 kHz) e fonte variável (0, 5,  $\pm 12$ ,  $\pm 15$ V) comunicando-se com o LabVIEW.

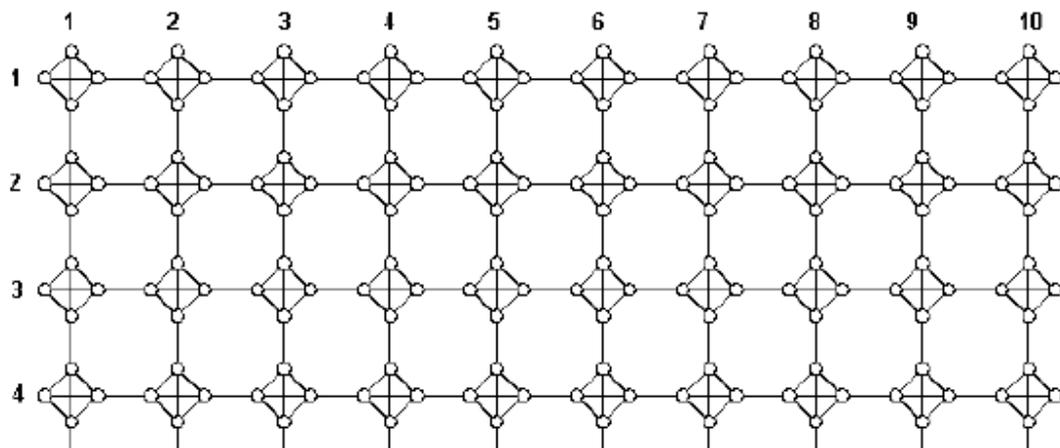
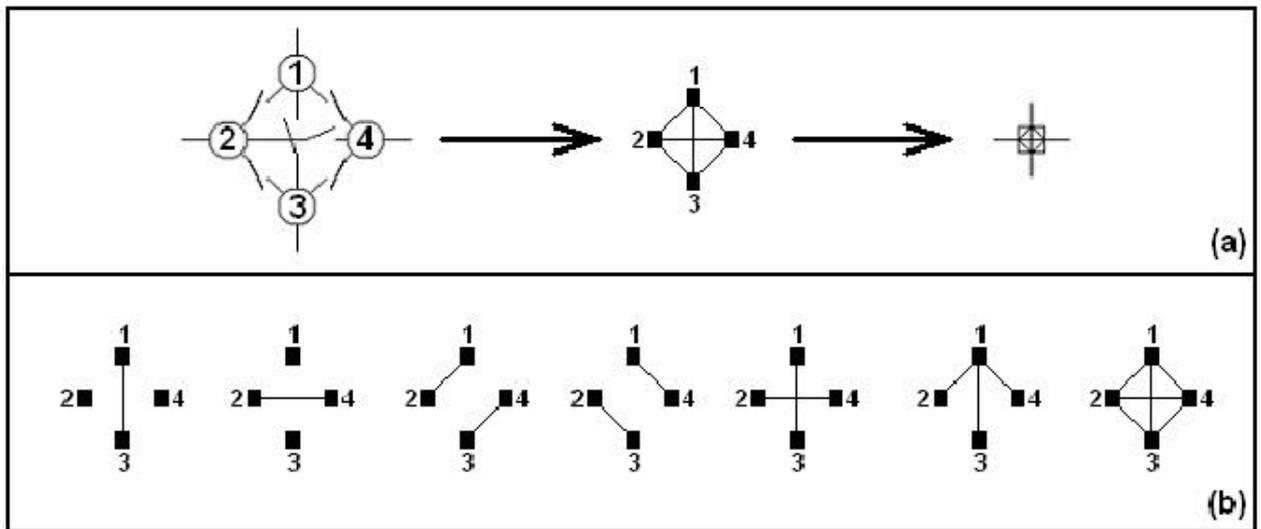
Neste sistema o LabVIEW apresenta vários VI's (Virtual Instrument) implementados com as funções básicas de controle/monitoração, podendo, pela facilidade de programação, criar novos VI's.

Em relação a custo, o ELVIS da NI (National Instruments) tem um valor expressivo de US\$ 7.000 FOB, desconsiderando os incentivos fiscais às universidades.

Na UNICAMP (proposta 1) implementou-se uma proposta inicial de laboratório remoto utilizando placas DAQs e instrumentos GPIB ligados a uma matriz de relés que podia ser roteada de acordo com o circuito em questão. Entretanto, apesar deste modelo ter grande importância para iniciar os estudos sobre o tema, apresentou alguns pontos negativos:

- Complexidade enorme para rotear os circuitos;
- Custo elevado devido à utilização dos equipamentos GPIB e placas de aquisição DAQs;

- As ligações dos relés induziam erros nos circuitos em análise;
- Necessidade de energização permanente da matriz de relés para o funcionamento;
- Inflexibilidade do sistema em introduzir novos elementos;



**FIGURA 10 – Modelo da Matriz apresentada pela Unicamp para rotear circuitos eletrônicos**

Os laboratórios remotos e/ou virtuais apresentados e pesquisados estão desenvolvendo meios para a elaboração de comunicação entre instrumentação inteligente, computadores e redes. Estes tipos de laboratórios geralmente utilizam sistemas existentes (DAQs, GPIB, etc.), gerando adaptações dispendiosas, falta de flexibilidade e conseqüentemente elevando o custo do sistema.

O desenvolvimento de metodologias para a criação dos experimentos remotos ainda é pouco explorada no mundo acadêmico, como indicado na FIGURA 11, porém é de grande importância para o sistema. Esta dissertação é parte de um projeto que pretende apresentar uma proposta para desenvolver metodologias flexíveis e amigáveis para criação de laboratórios remotos com o objetivo de facilitar a criação, ressaltar a teoria e agregar valores pedagógicos ligados ao ensino à distância.

<b>Laboratório</b>	<b>Custo</b>	<b>Modularidade</b>	<b>Hardware Dedicado</b>	<b>Rede</b>
<b>Singapura</b>	x	Não	Não	Sim
<b>MIT</b>	x	Não	Não	Sim
<b>Genova</b>	x	Sim	Não	Sim
<b>USP</b>	Alto	Sim	Não	Sim
<b>NI</b>	Alto	Sim	Não	Não
<b>Unicamp 1</b>	Alto	Não	Sim	Não
<b>Unicamp 2</b>	<b>Baixo</b>	<b>Sim</b>	<b>Sim</b>	<b>Sim</b>

**FIGURA 11 – Tabela Comparativa das Características dos Laboratórios Remotos/Virtuais com a nova proposta (UNICAMP 2)**

---

## CAPÍTULO 3

### ESTRUTURA GERAL E FUNCIONALIDADE DO HARDWARE

As fontes de inspiração inicial na criação do sistema proposto de laboratório remoto para o ensino de engenharia eletrônica foram: o custo, a falta de flexibilidade dos sistemas de hardware existentes e poucas inovações. Outro ponto a ser ressaltado é a reutilização dos experimentos laboratoriais existentes, como circuito, teoria, roteiro, etc.

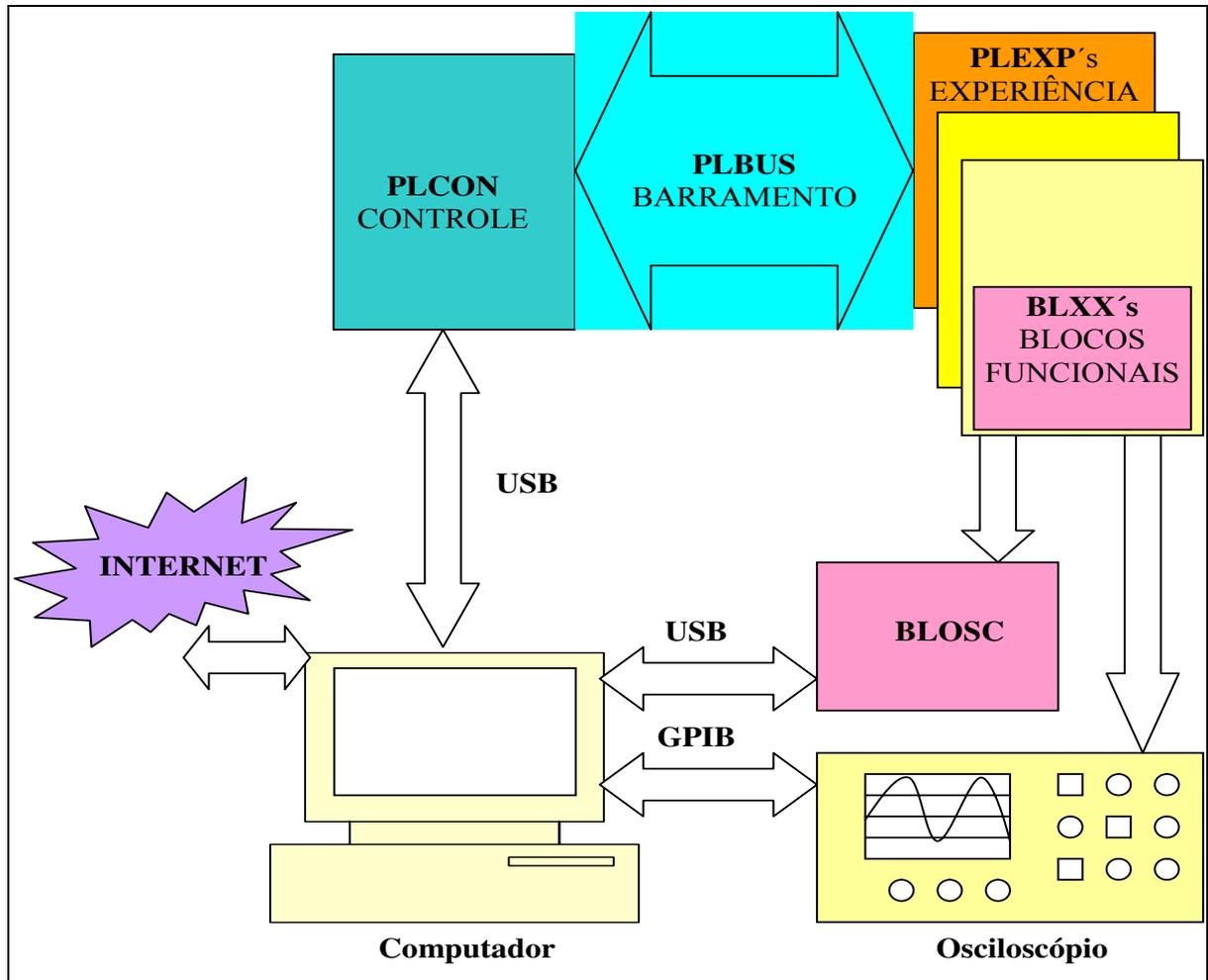
A proposta apresentada levou em consideração as definições requeridas pelo sistema como um todo, pesquisando e estudando as opções de tecnologias mais recentes e disponíveis no mercado, sempre objetivando o custo final do desenvolvimento.

#### 3.1 - INTRODUÇÃO

O sistema de hardware desenvolvido é constituído de quatro modelos de placas eletrônicas independentes:

- **Placa de Controle (PLCON)** : é a placa eletrônica capaz de interfacear com o Host via USB e atuar nos experimentos;
- **Placa de Experiência (PLEXP)** : são placas padrões com circuitos fixos que “automatizam” os experimentos convencionais do laboratório real;
- **Bloco Funcional (BLXXX)** : são pequenas placas eletrônicas com funções específicas, como relés (chaves), potenciômetros digitais, banco de resistores e instrumentos de análise e medição;
- **Placa de Barramentos (PLBUS)**: é a placa eletrônica que atua como interconexão entre os vários módulos de hardware, além de fornecer alimentação;

Todos estes componentes se comunicam conforme a arquitetura apresentada na FIGURA 12.



**FIGURA 12 – Arquitetura Geral do Laboratório Proposto, com as Placas de Controle, Barramento e Blocos Funcionais**

O funcionamento do sistema se inicia pela placa de controle geral, batizada de PLCON, recebendo os dados vindos do PC via USB, estes são interpretados através do protocolo desenvolvido para o microprocessador (PIC), que processa e disponibiliza os resultados nos barramentos.

Os resultados deste processamento são interpretados pela placa da experiência selecionada pelo PIC, que configura o circuito elétrico através dos blocos funcionais. Os blocos são acionados por circuitos digitais fixos nas placas de experiência.

Estes componentes desenvolvidos para automatizar os mais diversos tipos de experiências funcionam como chaves (relés), potenciômetros digitais, lista de resistores e terminais de instrumentos de medidas, podendo ser ampliado o número de funções dos mesmos, mantendo a mesma filosofia da autonomia, como por exemplo, um bloco conversor A/D.

Estas autonomias dos blocos permitem a modularidade do sistema, pois as chaves da placa de experiência 01 podem ser utilizadas na 02, 03..., 08 e vice-versa.

Finalizando o ciclo, tem os instrumentos de medidas baseado em computador medindo de forma real os resultados através dos terminais das placas de experiências. Estes dados podem ser obtidos na proposta atual via placa de aquisição GPIB ou pelo bloco instrumento (BLOSC) USB, a ser desenvolvido.

Nos subitens seguintes, será discutido mais detalhadamente cada elemento em separado do sistema proposto para o laboratório remoto.

### **3.2 – O HARDWARE DE CONTROLE GERAL - PLCON**

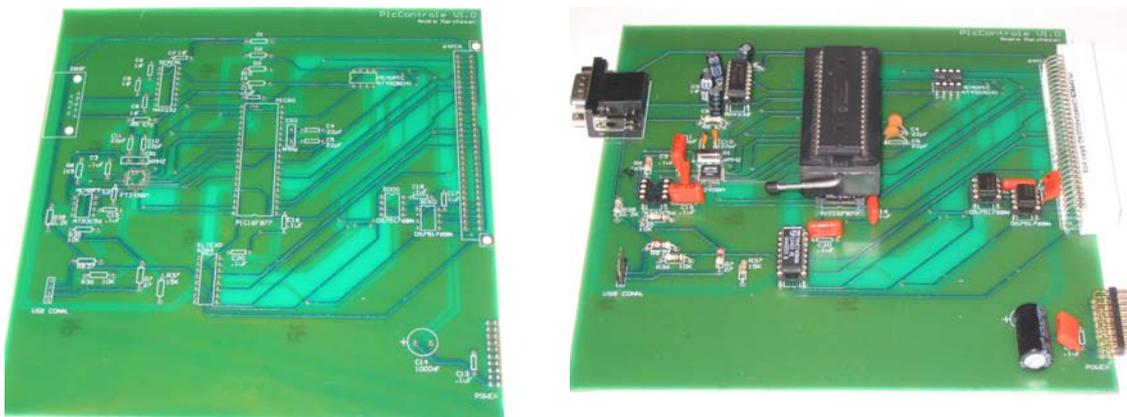
O hardware de controle, batizado de PLCON, é uma placa microcontrolada desenvolvida para se interligar todo o sistema (PC e experiências) através de diversos tipos de interface (USB/ SERIAL/ SPI).

Todos os tipos de interfaces utilizadas foram pesquisadas e estudadas, objetivando sempre o custo, a facilidade na utilização e a integração com o sistema desenvolvido.

Como mencionado a integração das diversas tecnologias utilizadas foi outro fator importante na decisão de algumas tecnologias, como por exemplo, a do protocolo SPI e o microprocessador PIC, uma vez que muitos componentes comerciais utilizam tal protocolo.

A elaboração do layout da placa foi desenvolvido no PROTEL 99SE, software de desenvolvimento de placas profissional com excelentes ferramentas e recursos, a partir do projeto eletrônico e de definições dos componentes utilizados no sistema, como por exemplo, dimensões e conexões entre placas.

A placa foi confeccionada em face dupla e furos metalizados, além de uma parte mecânica para encaixe no gabinete do sistema.



**FIGURA 13 – Placa de Controle (PLCON)**

### 3.2.1 – ESTRUTURA GERAL

A PLCON é constituída de seis sub-circuitos interligados. Cada circuito tem uma função específica definida conforme FIGURA 14.

<b>CIRCUITO</b>	<b>DESCRIÇÃO</b>	<b>COMPONENTE PRINCIPAL</b>
<b>USB/PIC</b>	Circuito que converte os dados vindos do PC através da comunicação USB em dados paralelos para entrar nas portas do microcontrolador (PIC) e vice-versa.	FT245BM (Ftdichip)
<b>PIC</b>	Circuito que processa e controla as entradas e saídas da PLCON.	PIC 16F877 (Microchip)
<b>PIC/SERIAL</b>	Circuito que permite programar e “Debugar” o PIC via serial	MAX232 (Maxxim)
<b>PIC/SPI</b>	Barramento que permite a comunicação com os componentes SPI existentes nas placas de experiências	
<b>SPI/RS485</b>	Converte os níveis de tensão do protocolo SPI para RS485 , eliminando ruídos externos provenientes do barramento único.	DS75176
<b>SERIAL/PARALELO</b>	Permite a conversão dos dados de saída via serial do PIC em paralelo para o barramento principal. (Shift Registers)	CD4094

**FIGURA 14 – Tabela Resumo dos Subsistemas da Placa de Controle (PLCON)**

O funcionamento básico da PLCON consiste em receber bytes de instruções e dados do PC via USB, disponibilizando-os paralelamente nas portas de entradas do PIC, processá-

los e colocar os resultados nas portas de saídas. Estas saídas podem ser SERIAL, PARALELA ou SPI.

A saída SERIAL é utilizada para debuggar o PIC, enviando os dados para as experiências diretamente ou para selecionar as experiências, porém neste último caso, passando antes pelo Shift Register (SR) da PLCON (SERIAL/PARALELO).

Já a PARALELA controla os STROBES dos Shift Registers (SR) de todas as experiências, pois a partir da seleção da experiência estes dados são interpretados apenas pela placa selecionada.

E finalizando tem as saídas do PIC comunicando via protocolo SPI. Este protocolo é muito utilizado para a comunicação entre componentes e sistemas. No caso específico, com o potenciômetro digital nas placas de experiência ou a memória na própria placa de controle.

A saída SPI ainda passa por um processo de conversão de nível para transmissão em modo diferencial antes do barramento geral de dados. Estes dados quando chegam até as experiências são invertidos novamente, evitando assim ruídos externos provenientes do chaveamento no barramento de dados.

O esquemático da placa de controle (PLCON) está disponível no APÊNDICE A.

### **3.2.2 – A USB (Universal Serial Bus)**

A necessidade de um novo método de conexão de periféricos ao PC, método este que deveria ser mais simples e eficiente, levou um consórcio de sete empresas (Intel, Compaq, Microsoft, etc) a desenvolver em 1995 o barramento USB.

### Características:

- Concebido para uma velocidade inicial de 12Mbps, atualmente 480Mbs com USB 2.0;
- Capacidade de suportar até 127 periféricos;
- Efetuar a distribuição de energia para periféricos de baixo consumo;
- Arquitetura plug-and-play, ou seja, não necessitam mais de efetuar configurações de recursos de hardware;
- O uso de conector universal externo que permite ao usuário instalar e remover periféricos sem sequer abrir o computador.

Atualmente tem mais de 500 milhões de dispositivos comunicando via USB, mostrando o domínio deste mercado no padrão de conectividade.

Quanto à organização das camadas de software necessárias para operar-se um dispositivo, o sistema USB HOST é composto por vários níveis de hardware e softwares. Um aplicativo requer o acesso a um periférico USB da forma padrão, como é feita para os periféricos comuns: chama funções da API (Application Programming Interface), ou seja, um conjunto de rotinas que um aplicativo utiliza para solicitar e conduzir serviços de baixo nível executados por um sistema operacional de computador.

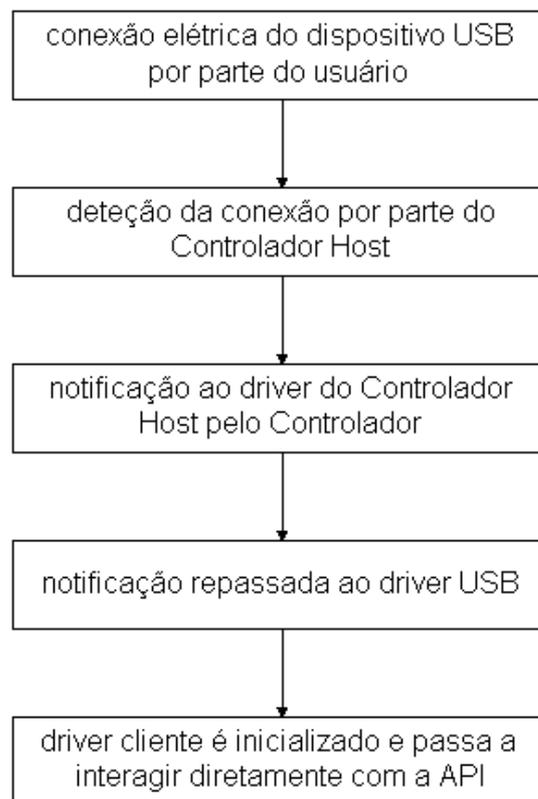
Num segundo estágio, a API chama rotinas do driver Cliente do periférico USB instalado. Este driver traduz os comandos da API para comandos USB. O driver Cliente é geralmente parte do sistema operacional ou vem instalável com o dispositivo USB.

A terceira camada de software indicada é o driver USB (USBD), que é aquela que dá ao sistema operacional o suporte ao USB.

A quarta camada de interesse é o driver do controlador HOST (HCD), que funciona a nível de Kernel do sistema operacional. O HCD provê o nível de software entre o hardware

do controlador HOST e o USB. É esta camada que realiza os acessos de I/O necessários para a operação do dispositivo USB. O HCD interpreta as chamadas do USB e constrói uma lista de estruturas, um descritor de transferências, uma fila principal e um buffer de dados para o controlador HOST.

Uma característica extremamente inovadora do USB é a possibilidade de conectar-se um novo dispositivo durante a execução do aplicativo. Neste caso, o controlador HOST detecta a conexão e envia uma mensagem ao HCD para avisá-lo do fato. Em seguida, o HCD faz a mesma notificação ao driver USB (USB). Este, então, inicializa o driver cliente do periférico conectado e, em seguida, torna-o operacional, de forma que o aplicativo já possa dispor de seus recursos. Tal seqüência está ilustrada a seguir.



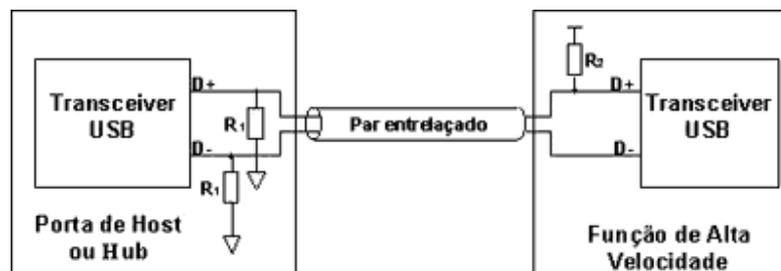
**FIGURA 15 – Seqüência de eventos desde a conexão da placa até a inicialização dos drivers**

O barramento físico é composto de um cabo com quatro fios: VBus, D+, D- e GND.

O fio VBus é o meio de fornecimento de alimentação para os dispositivos que necessitarem dela. Em um sistema USB, existem hubs e funções que possuem alimentação própria e hubs e funções que são alimentados pelo barramento através de VBus. VBus é nominalmente +5 V. Para aplicações de alta velocidade, os fios D+ e D- são entrelaçados.

Os dados são transmitidos através de D+ e D- por meio de diferenças de tensão entre eles.

A posição dos resistores de pull-up muda dependendo de tratar-se de alta ou baixa velocidade. Quando não existe função conectada ao hub, os resistores de pull-down fazem com que ambos D+ e D- fiquem abaixo de um valor de tensão de limiar para a detecção da presença do dispositivo. Se essa condição persistir por mais que 2,5 microssegundos, é caracterizada a desconexão do dispositivo. A conexão de um dispositivo é caracterizada pela situação oposta, ou seja, quando apenas uma das linhas é levada além da tensão de limiar e esta situação persiste por mais de 2,5 microssegundos.



**FIGURA 16 – Esquema Físico de Conexão do Barramento**

### 3.2.3 A USB e o FT245BM

Devido a todas as características apresentadas pela USB, optou-se por utilizar este protocolo de comunicação entre a PLCON e o PC (host).

Para comunicar via USB com o microprocessador da placa de controle, utilizou-se o chip FT245BM da FTDICHIIP como interpretador do protocolo, uma vez que o mesmo não tinha uma função direta.

O FT245BM permite um ótimo custo benefício na transmissão de dados entre um periférico e um host PC com velocidade de até 1 Megabyte por segundo.

Este chip, além de comunicar-se com o host (PC), disponibiliza os dados FIFO (8 bits) ao microprocessador no formato paralelo, agilizando assim seu processamento.

As entradas/saídas são controladas pelos pinos TXE# e RXF# no FT245BM e comandados pelo microprocessador.

#### Características do FT245BM:

- Apenas um único Chip para conversão USB/Paralelo;
- transferência de dados bi-direcional;
- Transfere até 1MB / s, utilizando o Driver D2XXL (DLL) for Windows;
- USB protocolo on-chip;
- 384 Byte Paralelo Tx buffer / 128 Byte Paralelo Rx Buffer;
- RX buffer ajustável (timeout);
- Integrado regulador 3.3V para USB I/O
- Power-On-Reset circuit
- Compatível USB 1.1 e USB 2.0

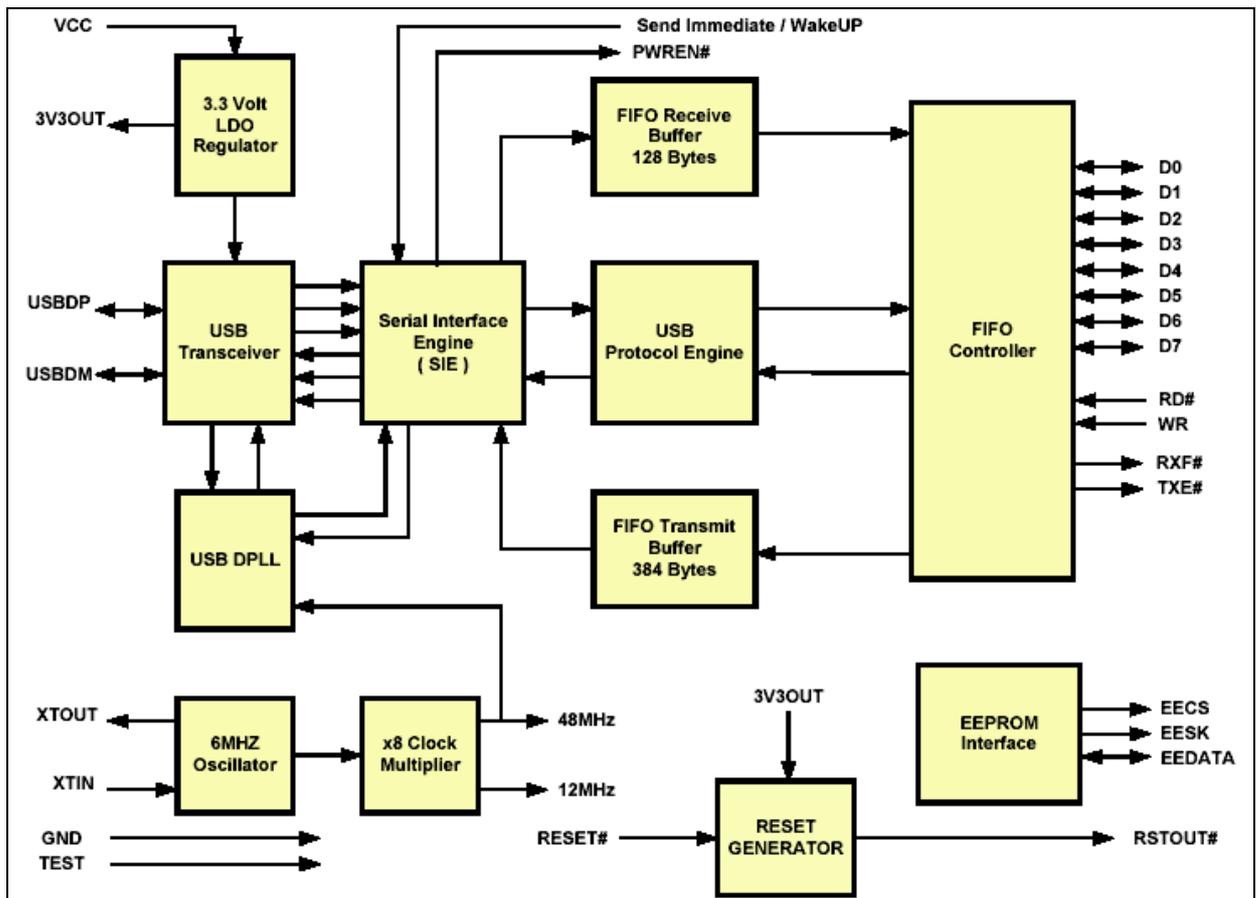


FIGURA 17 – Arquitetura do FT245BM

### 3.2.4 – MICROCONTROLADOR PIC 16F877

A PLCON tem como principal componente o PIC16F877. Este chip é um microcontrolador de 40 pinos, fabricado pela empresa MICROCHIP empregando tecnologia de fabricação CMOS. Dispõe de memória FLASH para programação o que permite a reprogramação sem a necessidade de utilizar luz ultravioleta. Necessita de poucos componentes externos, o que reduz custos e o consumo de energia, aumentando a confiabilidade.

Características do PIC16F877:

- 33 portas configuráveis como entrada ou saída.
- 4 interrupções disponíveis (TMRO, Externa, Mudança de Estado e EEPROM).
- Memória de programação FLASH.
- RAM de 368\*8 bytes.
- Memória EEPROM (não volátil) interna de 256\*8 bytes.
- Programação com 35 instruções.

Os microcontroladores PIC apresentam uma estrutura de máquina interna do tipo Havard, enquanto grande parte dos microcontroladores tradicionais apresenta uma arquitetura tipo Von Neumann. A diferença está na forma como dados e programas são processados pelo microcontrolador. Na arquitetura tradicional, tipo Von Neumann, existe apenas um barramento (bus) interno (geralmente de 8 bits), por onde passam as instruções e os dados. Já na arquitetura tipo Havard existem dois barramentos internos, sendo um de dados e outro de instruções [9]. No caso dos microcontroladores PIC, o barramento de dados é sempre de 8 bits e o de instruções pode ser de 12, 14 ou 16 bits, dependendo do microcontrolador.

Os PICs utilizam uma tecnologia chamada RISC, que significa Reduced Instruction Set Computer (Computador com Conjunto de Instruções Reduzido). Desta forma, o PIC16F877 possui cerca de 35 instruções, muito menos que os microcontroladores convencionais.

### **3.2.5 – BARRAMENTO DIFERENCIAL**

Os meios de transmissão estão sujeitos a diversos tipos de ruído e interferências com diversas origens.

O ruído vai afetar de modo decisivo a recepção dos sinais já que o receptor deve ter a capacidade de distinguir o sinal útil e filtrar todos os outros.

Para atenuação dos ruídos de transmissão dos dados entre o PIC e potenciômetros, componentes que utilizam o protocolo de comunicação SPI, foi desenvolvido uma transmissão diferencial.

Através do chip *DS75176BM* podemos transmitir e receber dados, em alta velocidade, no formato diferencial, diminuindo os efeitos de ruídos, além de funcionar como um driver de corrente para os componentes interligados, com a opção de deixar o chip no modo de alta impedância, comportando-se como um elemento não existente.

Na PLCON foi utilizado dois chips na configuração para transmissão diferencial, nas saídas do PIC para o protocolo SPI, sendo um na porta SDO e outra na SCL.

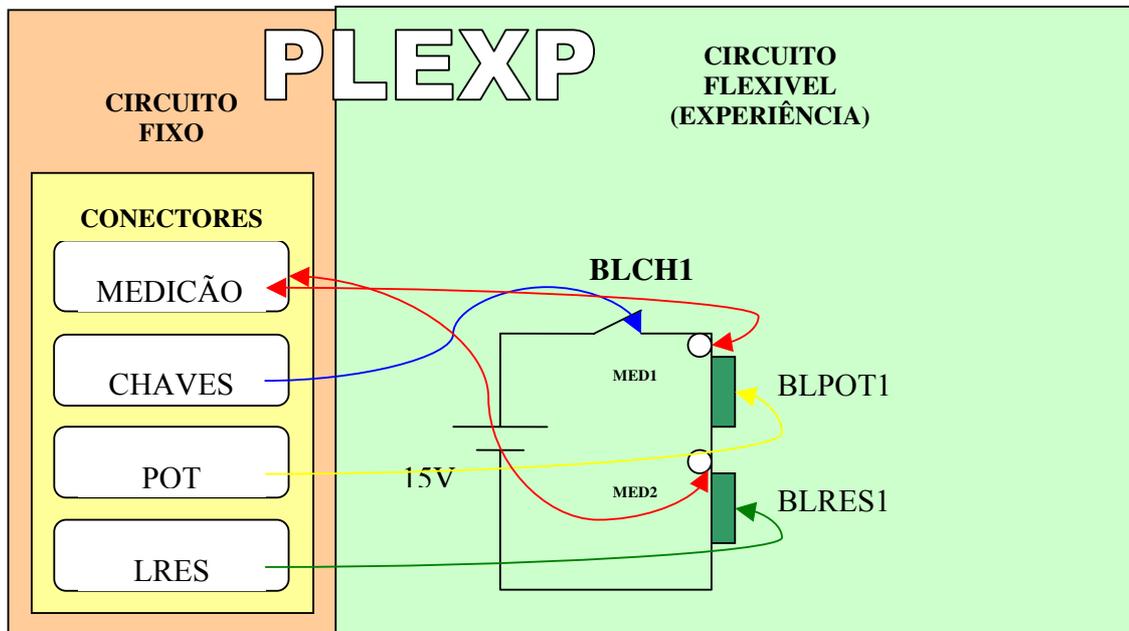
O funcionamento básico deste chip consiste na elevação da tensão para níveis normalizados e a partir da diferença entre tensões considerar nível lógico alto ou baixo.

### **3.3 – SISTEMA DE CONTROLE LOCAL - PLEXP**

As placas de controle local, seguindo o mesmo padrão de nomenclatura são denominadas PLEXP's e é nelas que ocorre a atuação direta, nas experiências em questão após os recebimentos dos comandos da PLCON.

As PLEXP's são placas com dois CIRCUITOS: um fixo, onde temos circuitos lógicos de controle e os pinos de conexão interno e externo; e o outro flexível, onde podemos

conectar vários tipos de experimentos, respeitando a filosofia adotada na elaboração das mesmas, conforme apresentado na FIGURA 18.

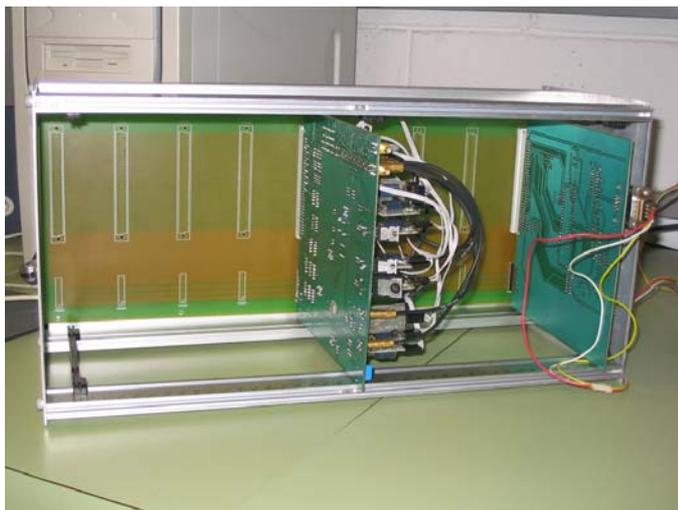


**FIGURA 18 – Diagrama Esquemático da Placa de Experiência (PLEXP)**

Os comandos de acionamento da PLEXP vêm da PLBUS após o processamento na PLCON, conforme descrito no item anterior. Estes dados chegam em paralelo, acionando as portas lógicas, shift register's (SR) e as chaves analógicas, disponibilizando os resultados em pinos de saídas e entradas do CIRCUITO fixo.

Através dos pinos de saídas e entradas podemos conectar, via cabos, os blocos funcionais como chaves, potenciômetros, lista de resistores e instrumentos de medição da experiência. Desta forma é possível automatizar as experiências localizadas no CIRCUITO flexível da PLEXP.

As PLEXP's são colocadas em um gabinete horizontal com conectores independentes para dados e potência, permitindo a conexão de até 8 experimentos. Entretanto esta disposição não se mostrou adequada devido à grande quantidade de conectores, ocupando um grande volume.



**FIGURA 19 – Gabinete horizontal com as placas: experiência e controle**

Todas as placas desenvolvidas para o sistema, foram elaboradas (PCB) utilizando o software PROTEL 99SE. Na PLEXP teve-se uma atenção especial devido às dimensões críticas para atender o CIRCUITO fixo e as possíveis experiências, sendo que esta placa foi referencia de medidas para o sistema. A placa física possui face dupla e furos metalizados, além de uma parte mecânica para encaixe no gabinete do sistema.

### 3.3.1 – ESTRUTURA GERAL

A PLEXP é constituída de:

#### *CIRCUITO FIXO:*

- 1- Circuitos de endereçamento da PLEXP através de portas lógicas;
- 2- Circuitos de multiplexação através dos Shift Registers (SR);
- 3- Circuito de seleção de pontos de medição;
- 4- Circuito de desacoplamento do osciloscópio de entrada/saída;
- 5- Conectores de entrada e saída.

#### *CIRCUITO FLEXIVEL:*

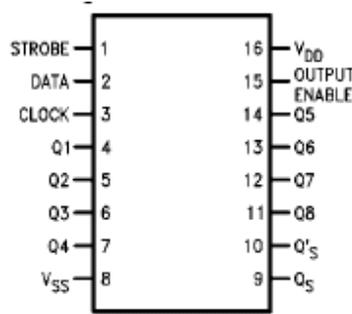
1. Circuito da experiência;
2. Conectores de entrada e saída.

O esquemático da placa de experiência (PLEXP) esta disponível no APÊNDICE B.

### 3.3.2 – O SHIFT REGISTER (SR)

A PLEXP possui vários componentes que devem ser controlados, portanto necessita de vários bits de controle e conseqüentemente de um número muito grande de barramentos para a comunicação. Entretanto utilizando os Shift Registers (CD4094 – SMD), é possível multiplexar os sinais de entrada, resolvendo o problema através de dois barramentos (dados e clock).

O CD4094 possui 8 pinos de saída e 03 de entrada (dados e controle). Os dados são deslocados para Q1, Q2...Q8, conforme ocorre o pulso de clock, desde que o chip esteja habilitado, conforme FIGURA 20.



**FIGURA 20 – Pin-Out do Shift Register com Entradas e Saídas (DIP)**

Na verdade estes dados são enviados serialmente e vão sendo deslocados e armazenados em latch's. Quando ocorre o strobe (habilita), os dados são disponibilizados na saída paralelamente.

Clock	Output Enable	Strobe	Data	Parallel Outputs		Serial Outputs	
				Q1	Q <sub>N</sub>	Q <sub>S</sub> (Note 1)	Q' <sub>Σ</sub>
	0	X	X	Hi-Z	Hi-Z	Q7	No Change
	0	X	X	Hi-Z	Hi-Z	No Change	Q7
	1	0	X	No Change	No Change	Q7	No Change
	1	1	0	0	Q <sub>N-1</sub>	Q7	No Change
	1	1	1	1	Q <sub>N-1</sub>	Q7	No Change
	1	1	1	No Change	No Change	No Change	Q7

**FIGURA 21 – Tabela Verdade do Componente - 4094**

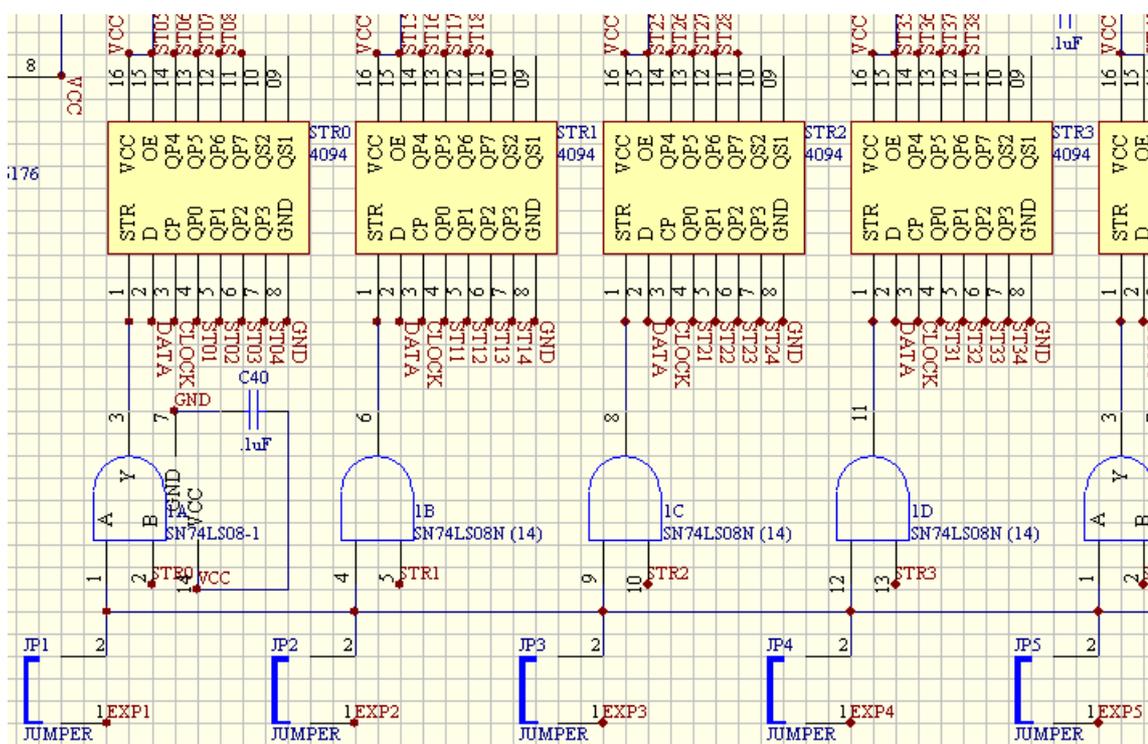
### 3.3.2 – SELEÇÃO DA PLEXP

Como pode haver 8 experiências sendo controlada pela PLCON, é necessário haver um controle de posicionamento da mesma no barramento para seleção.

As PLEXP's são idênticas, sendo que a única diferença esta no CIRCUITO das experiências, portanto é necessário identificar qual a posição da placa em operação para receber os comandos de acionamento.

A PLCON disponibiliza oito saídas de seleção no barramento. Assim no local reservado do barramento para receber a PLEXP, terá um sinal de seleção para habilitação da respectiva placa, caso o jumper esteja na posição correta.

Este jumper liga o sinal de seleção vindo do barramento nas entradas de 08 portas AND com duas entradas (74LS08N), portanto se PLEXP estiver selecionada temos em uma das entradas nível alto, e a outra entrada da porta é comum a todas PLEXP, assim tem-se a seleção da placa, conforme trecho do esquemático:



**FIGURA 22 – Trecho do Esquemático de Seleção da PLEXP**

### 3.3.3 – SELEÇÃO DO PONTO DE MEDIÇÃO

No CIRCUITO fixo temos 02 pontos de saída de sinais, estes pontos são levados até o osciloscópio. Estes pontos separam os dois canais do GND do equipamento de instrumentação. Esta separação ocorre por meio físico, ou seja, os conectores de entrada desacoplam o GND do Ch1 ou do Ch2. Estas saídas desacopladas são levadas a 03 chaves analógicas (AD7501) com 8 saídas cada, totalizando uma combinação de 24 pontos.

O chip AD7501, da Analog Devices, é uma chave analógica multiplexada de 8 canais de saída, com controle digital (03 bits), resistência típica de 170 Ohms (RON) e tensão de operação  $\pm 15V$ .

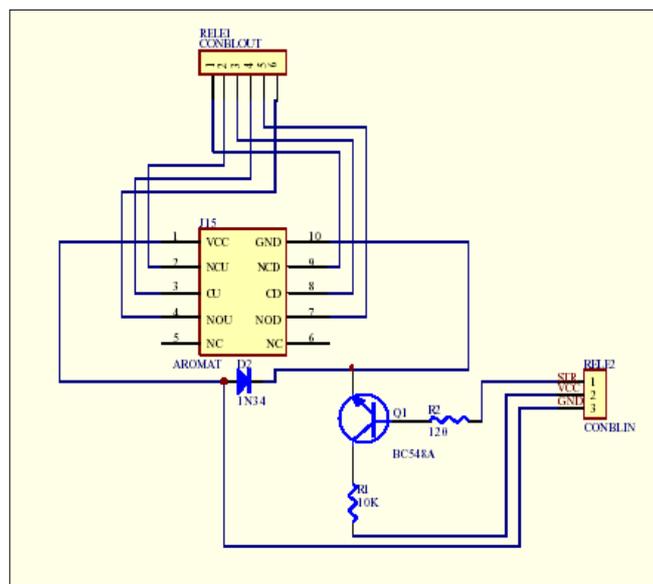


### 3.3.4 – ACIONANDO CHAVES

As chaves são fundamentais para o acionamento de componentes ou na alteração do lay-out do circuito experimental.

Os sinais de comandos para abertura ou fechamento das chaves (relés) são acionados pelos Shift Register's (SR). Este componente recebe informação do barramento provenientes da PLCON.

Através dos SR's é possível controlar até 23 chaves no mesmo experimento.



**FIGURA 24 – Circuito das Chaves e Conexão (PLEXP)**

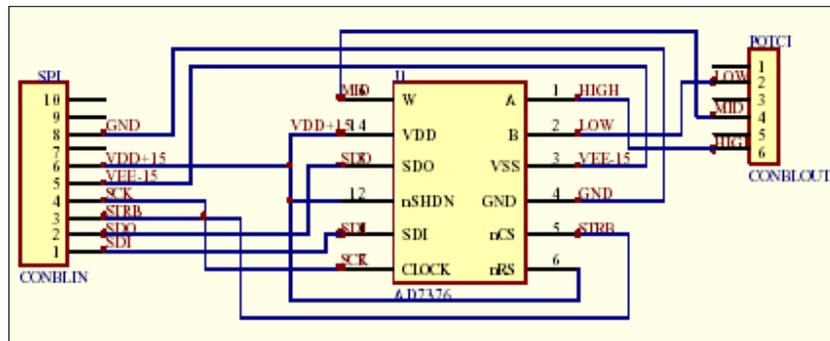
### 3.3.5 – SELEÇÃO DE RESISTORES

A idéia de poder ter um “almoxarifado” de resistores comerciais no experimento remoto, gerou a criação de um Bloco chamado BLRES.

O BLRES possui 03 chaves analógicas que operam com tensão de  $\pm 15V$  com 24 opções de seleção.

Para cada experimento temos a disponibilidade de 04 BLRES e assim obter até 96 valores diferentes de resistências.

Este Bloco, como as chaves são comandados por sinais enviados por cabos e conectores dedicados localizados no CIRCUITO fixo definido nas PLEXP's.



**FIGURA 25 – Circuito do Bloco - Lista de Resistores (BLRES)**

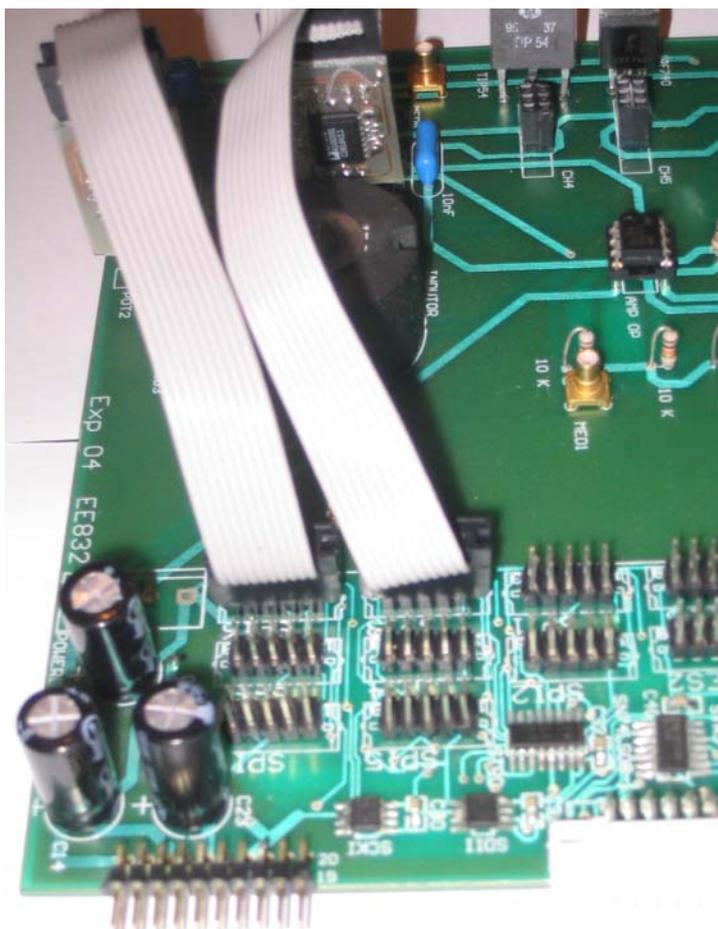
### 3.3.6 – ATUANDO NOS POTENCIOMETROS - PROTOCOLO SPI

As experiências de laboratório, em sua maioria, necessitam de potenciômetros para variação de efeitos eletrônicos, auxiliando nas observações teóricas.

O Bloco funcional potenciômetro (BLPOT), opera com  $\pm 15V$  e variação de 0 a  $10K\Omega$  em 128 passos, acionado digitalmente. Este bloco é composto de um único componente (CI AD7376 SMD) além dos conectores de entrada e saída, recebendo dados via SPI.

Basicamente todos os CI's, com esta finalidade, trabalham com este protocolo de 3 fios (Serial Data Input).

Como os demais Blocos Funcionais, os BLPOT's são controlados por cabos e conectores dedicados localizados no CIRCUITO fixo definido nas PLEXP's. Sendo possível utilizar até 08 potenciômetros no mesmo experimento.



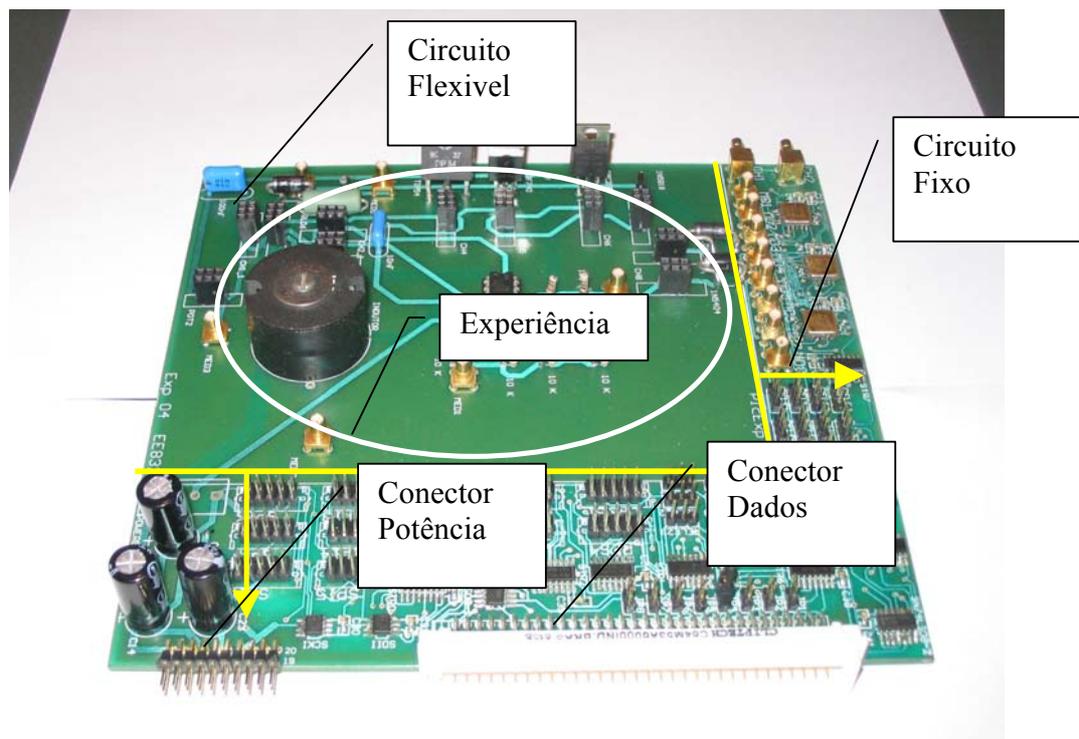
**FIGURA 26 – Conectores de Conexão para os Blocos Potenciômetros (BLOSC) no Circuito Fixo da Placa da Experiência (PLEXP)**

### 3.3.7 – Circuito da EXPERIÊNCIA

O espaço reservado para a elaboração das experiências não tem comunicação com o CIRCUITO fixo. Esta comunicação se dá por meio de cabos, tornando o espaço para o circuito experimental muito flexível e prático.

A experiência é montada de uma forma que o seu lay-out e características possam ser alterados por chaves, potenciômetros, resistores e pontos de medição. Desta forma é possível automatizar as experiências convencionais ou criar novas configurações.

Os componentes citados têm seus espaços reservados (footprint) que facilitam a confecção dos experimentos. Estes componentes são ligados à área da experiência via conectores específicos e estes conectados ao CIRCUITO fixo via cabos.



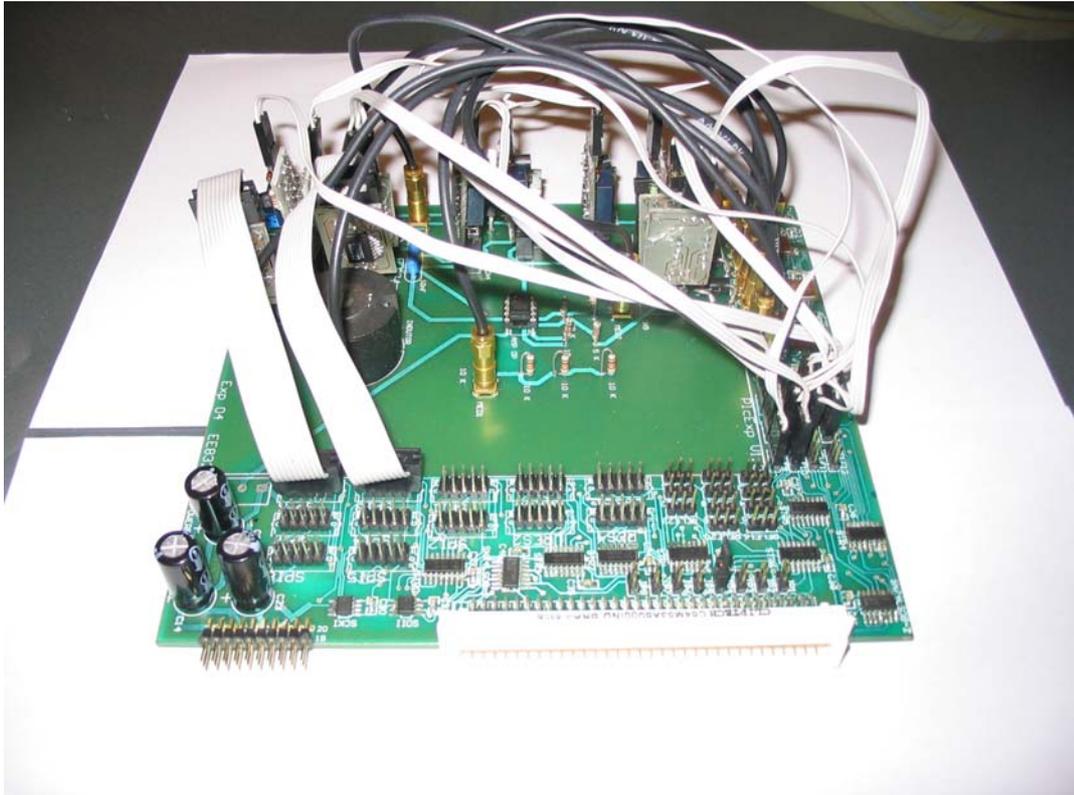
**FIGURA 27 – A Placa de Experiência (PLEXP)**

A grande flexibilidade deste sistema permite à mesma PLEXP receber diversas experiências diferentes, utilizados os blocos em outras posição ou situações. Uma sugestão para trabalhos futuros seria fazer uma PLEXP com um protoboard, podendo assim realizar experiências simples, ou montagens de teste.

### **3.4 – BLOCOS FUNCIONAIS**

As automatizações das experiências dependem de componentes autônomos controlados. Estes Blocos são pequenas placas com conectores de entrada e saída padronizados e características físicas e funcionais semelhantes.

Os conectores de entrada recebem via cabos os sinais e a alimentação, necessária ao funcionamento, diretamente do CIRCUITO Fixo da PLEXP. E os conectores de saída são plugados no CIRCUITO Flexível (experiências).



**FIGURA 28 – A Placa de Experiência Completa (PLEXP)**

Blocos desenvolvidos: chave, potenciômetro, lista de resistores e osciloscópio. Utilizando-se a mesma filosofia pode-se criar vários outros módulos práticos como conversor A/D ou D/A, gerador de sinais, etc...

Podemos ressaltar algumas características destes Blocos:

- Facilidade de confecção;
- Baixo custo;
- Grande flexibilidade de utilização, pois servem para várias experiências;
- Dimensões reduzidas;
- Não necessitam de alimentação independente, pois utilizam a placa da experiência;
- Facilidade na mudança das características dos componentes principais.

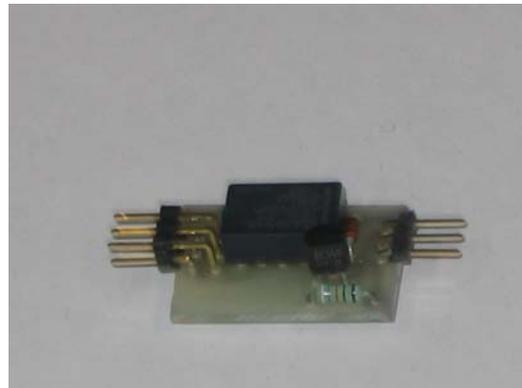
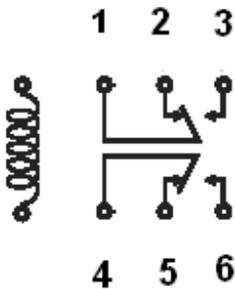
### 3.4.1 – BLOCO CHAVE – BLCH

O Bloco Chave, batizado de BLCH, é um dos mais importantes, pois são relés que funcionam como chaves controladas. Estes blocos permitem alterar a configuração dos circuitos experimentais, mudar os pontos de inspeção e adicionar componentes para análise.

O BLCH é uma placa reduzida (14mmx21mm) com um mini-relé (5V) acionado pelo circuito (APÊNDICE C) composto de transistor, diodo e resistor, além dos conectores de entrada e saída. Este circuito é alimentado com GND e 5V e acionado pelo Shift Register correspondente através do conector de entrada.

O componente principal da chave é o RELÉ TQ2-5V (NAIS) devido à baixa potência, dimensões reduzidas e não interferir no circuito do experimento.

Temos no conector de entrada 3 pinos : VCC, GND, e STR(sinal); e de saída o conector de 6 pinos, correspondendo exatamente à saída do relé, conforme FIGURA 29.



**FIGURA 29 – Bloco Chave - BLCH**

### 3.4.2 – BLOCO LISTA DE RESISTORES – BLRES

O Bloco Lista de Resistores, também chamado de BLRES, pretende funcionar como “*almoxarifado*” de resistências, ou seja, temos vários resistores comerciais interligados por chaves analógicas, e estas acionadas por lógica digital proveniente da PLEXP.

O Bloco BLRES consiste de um circuito composto de 03 componentes multiplexadores que operam com tensão de  $\pm 15V$  (7501SMD), permitindo a seleção de 24 posições distintas, podendo disponibilizar 24 resistores de valores diferentes no ponto de sua utilização.

Conector de entrada do Bloco:

**VDD / VEE / GND** – Alimentação do bloco;

**BIT 0 / BIT1 / BIT2** – Bits de seleção das chaves;

**CH1 / CH2 / CH3** – Bits de seleção dos componentes habilitados.

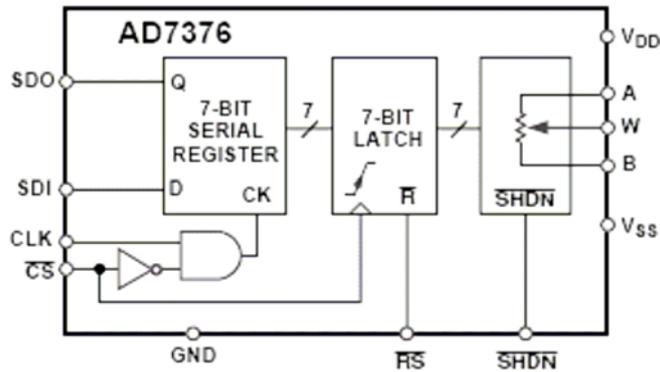
A conexão com a PLEXP (experimento), ou seja, a saída, consiste de um conector de 06 pinos fêmea, apesar da utilização de apenas 02. Desta forma é possível obter um maior apoio do bloco.

### 3.4.3 – BLOCO POTENCIOMETRO DIGITAL – BLPOT

A maioria dos experimentos necessita de variação de tensão ou corrente, portanto podemos utilizar nestes casos os Blocos Potenciômetro digitais, batizados de BLPOT's.

Este bloco contém um único componente, CI (AD7376 SMD), que varia de 0 a  $10K\Omega$  através de 128 passos e opera com níveis de tensão  $\pm 15V$ . O seu controle ocorre através do

protocolo de comunicação SPI, muito utilizado para este tipo de componentes, e disponível no PIC.



**FIGURA 30 – Diagrama Funcional do BLPOT**

O BLPOT também é uma placa pequena (14mmx21mm) contendo o CI AD7376 – SMD, conectores de entrada e saída, conforme descrito na FIGURA 31.

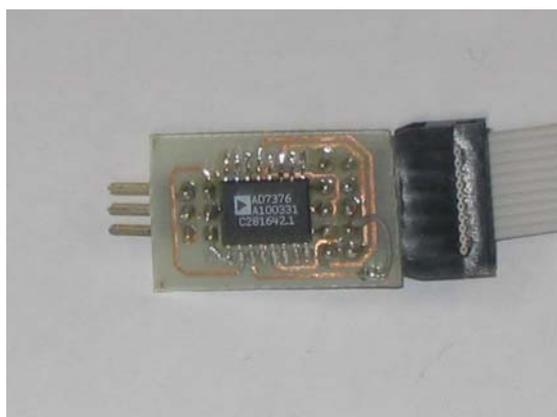
CONECTORES	
<b>ENTRADA (10 PINOS/MACHO – SIP 10)</b>	<b>SAIDA (06 PINOS/MACHO – SIP 6)</b>
SDO;SCK;SDI;VDD:VEE;GND;CS (STR)	A;B;W

**FIGURA 31 – Tabela de Descrição dos Pinos BLPOT**

Para comunicação com o BLPOT é necessário entrada de dados SDI, saída SDO, clock SCK e habilitação STR. SDI, SDO e SCK provêm direto do microcontrolador, que

também se comunica através do protocolo SPI. A habilitação STR vem das lógicas dos Shift Registers dedicados a esta operação.

Outro detalhe importante foi à utilização de transmissão diferencial de dados entre a saída/entrada do PIC e o Bloco BLPOT. Desta forma tem-se dois conversores (DS75176BN) na PLCON, um para o SCK e outro para o SDO e, repetindo o mesmo processo, dois inversores na PLEXP (DS75176BN). O uso desta forma de transmissão auxilia na diminuição dos ruídos nos barramentos de comunicação.



**FIGURA 32 – A Placa do BLPOT (AD7376)**

### **3.4.4 – NOVOS INSTRUMENTOS USB**

Utilizando a mesma filosofia apresentada para os demais Blocos Funcionais é possível, através da USB, implementar novos elementos: osciloscópio, gerador de funções, fonte programável, etc.

Como exemplo do osciloscópio, em desenvolvimento, para substituir o osciloscópio convencional e a comunicação GPIB existente no sistema. Desta forma os dados são transmitidos diretamente ao PC via USB.

O osciloscópio proposto apresenta dois canais, taxa de amostragem de 40ms, banda analógica de 10MHz e alimentação da própria comunicação USB.

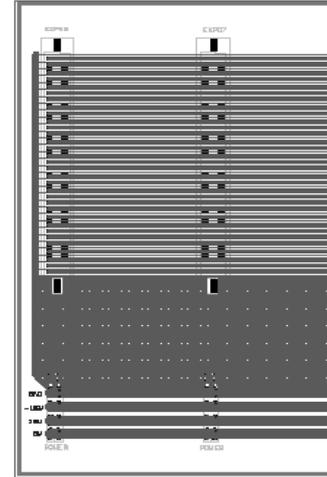
O custo inicial do bloco ficou em aproximadamente US\$ 100,00, sendo extremamente inferior ao sistema convencional de instrumentação (osciloscópio com saída GPIB, placas de aquisição, cabos, software de comunicação).

### **3.5– O BARRAMENTO DE DADOS E ALIMENTAÇÃO**

O barramento de dados consiste de 64 vias, sendo que 23 são de dados e os restantes compõem uma malha de terra. Com a intercalação de dados e GND, conforme FIGURA 32, pretendemos diminuir os ruídos gerados pelo chaveamento constante.

Independente do barramento de dados, porém na mesma placa, tem-se o barramento de alimentação +5, +15, -15 V e GND, com largura de 8mm cada para as tensões, e o GND fecha com a malha terra da placa.

Existem 04 conectores externos para receber a alimentação e 08 conectores DIN (macho 180°) 64 pinos interno para receber as experiências.



**FIGURA 33 – Placa e PCB do Barramento**

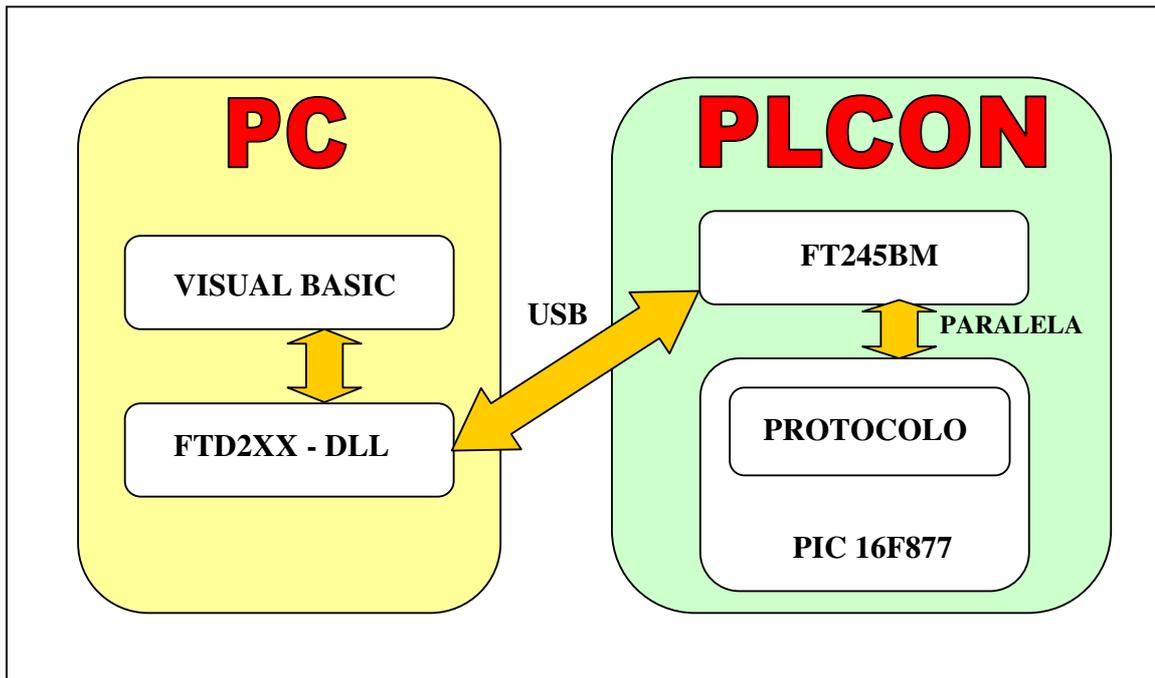
## **CAPÍTULO 4**

### **ASPECTOS DE SOFTWARE**

A idéia de flexibilidade do sistema através de uma placa de controle inteligente gerou a necessidade de se integrar tecnologias e elaborar um protocolo de comunicação dedicado entre o PC (host) e a PLCON.

A comunicação PC(host) com o chip FT245BM utilizou a DLL – FT2DXX desenvolvida pelo próprio fabricante do componente. Esta DLL pode ser incorporada por várias linguagens de programação que permitam anexar DLL's como ferramentas. Neste trabalho foi testado em Visual Basic e LabView.

No PIC foi desenvolvido em C, utilizando o compilador Hitech, o protocolo de comunicação para interpretar os códigos gerados pelo FT245BM.



**FIGURA 34 – Arquitetura do Software**

As vantagens da utilização do PIC 16F877 da Microchip já foram discutidas no CAPÍTULO anterior, juntamente com o FT245BM. Pretende-se descrever, nesta etapa, o protocolo desenvolvido bem como suas características e o funcionamento da DLL utilizada.

#### **4.1 – TRABALHANDO COM A DLL - HOST**

O FTD2XX é um software aplicativo de interface Windows para comunicação com FT232 USB UART ou FT245 USB FIFO. Este aplicativo é baseado na utilização da FTD2XX.DLL (dynamic link library), facilitando assim a comunicação com diversas linguagens de programação abertas a esta tecnologia.

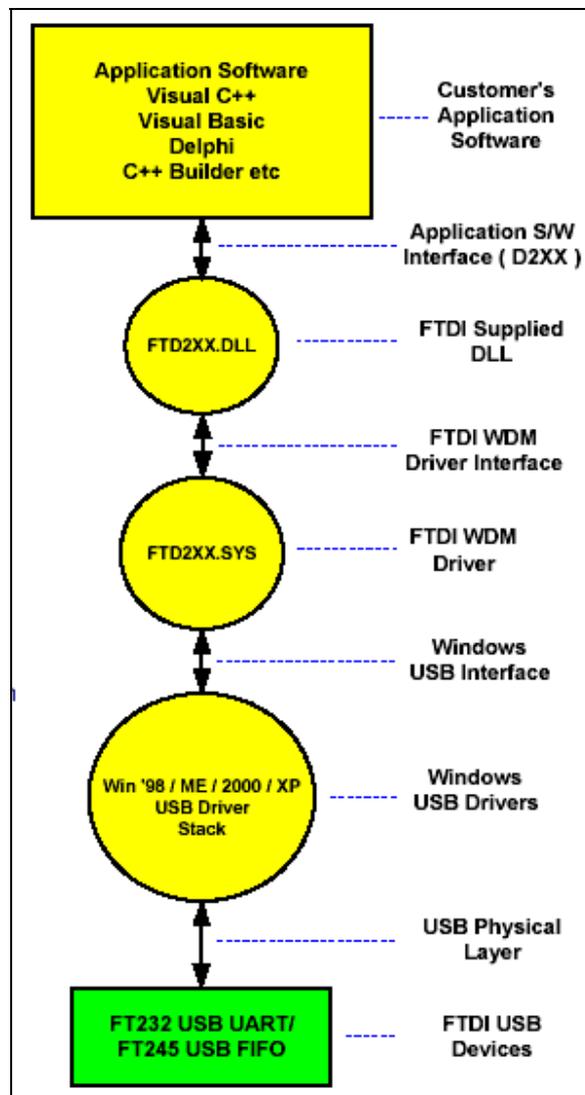
A seqüência básica para utilização deste driver segue as seguintes funções:

[FT\\_ListDevices](#) retorna a informação do componente FTDI conectado. Antes de acessar o dispositivo, o mesmo deve ser aberto. [FT\\_Open](#) abre e retorna o handle que será utilizada por todas as funções de interface. Após a abertura com sucesso, os I/O podem ser acessados através das funções [FT\\_Read](#) e [FT\\_Write](#). E para finalizar as operações básicas é necessário fechar o dispositivo usando [FT\\_Close](#).

Funções básicas:

<a href="#"><u>FT_ListDevices</u></a>	Lista os dispositivos conectados
<a href="#"><u>FT_Open</u></a>	Abre o dispositivo e retorna um handle que será utilizado nas outras funções
<a href="#"><u>FT_OpenEx</u></a>	Abre o dispositivo desejado e retorna um handle. O nome do dispositivo pode ser um número serial ou descrição.
<a href="#"><u>FT_Close</u></a>	Fecha o dispositivo aberto
<a href="#"><u>FT_Read</u></a>	Lê os dados do dispositivo
<a href="#"><u>FT_Write</u></a>	Escreve no dispositivo
<a href="#"><u>FT_SetBaudRate</u></a>	Ajusta o baud rate do dispositivo
<a href="#"><u>FT_SetDataCharacteristics</u></a>	Ajusta as características dos dados, como o número de bits, stop bits e parity
<a href="#"><u>FT_SetFlowControl</u></a>	Ajusta o controle de fluxo do dispositivo
<a href="#"><u>FT_SetDTR</u></a>	Ajusta o sinal de controle do terminal de leitura (DTR)
<a href="#"><u>FT_ClrDTR</u></a>	Limpa o sinal de controle do terminal de leitura (DTR)
<a href="#"><u>FT_SetRTS</u></a>	Ajusta o Request para enviar o sinal de controle (RTS)
<a href="#"><u>FT_ClrRTS</u></a>	Limpa o Request para enviar o sinal de controle (RTS)
<a href="#"><u>FT_GetModemStatus</u></a>	Obtém o status do modem do dispositivo
<a href="#"><u>FT_SetChars</u></a>	Ajusta um especial caractere para o dispositivo
<a href="#"><u>FT_Purge</u></a>	Limpa a transmissão e recepção do buffer do dispositivo
<a href="#"><u>FT_SetTimeouts</u></a>	Ajusta o timeouts de leitura e escrita do dispositivo

<a href="#">FT_GetQueueStatus</a>	Obtém o número de caractere recebido
<a href="#">FT_SetBreakOn</a>	Ajusta a condição de BREAK
<a href="#">FT_SetBreakOff</a>	Inicializa a condição de BREAK
<a href="#">FT_GetStatus</a>	Obtém o status do dispositivo incluindo o número de caracteres recebidos, enviados e o evento
<a href="#">FT_SetEventNotification</a>	Ajusta a condição de notificação do Chip
<a href="#">FT_ResetDevice</a>	Envia um comando de Reset para o dispositivo



## FIGURA 35 – Arquitetura do FT2XX

### 4.2 – O PROTOCOLO DE COMUNICAÇÃO - PIC

O protocolo de comunicação implementado no PIC foi desenvolvido na linguagem de programação C utilizando o compilador da Hitech. (programa – **APÊNDICE D**).

As palavras do protocolo são compostas por 5 bytes, sendo dois bytes iniciais de cabeçalho, um byte de comando e dois bytes de dados.

#### Formato básico do protocolo

<b>S</b>	<b>M</b>	<b>Comando</b>	<b>Dado (8 bits)</b>	<b>Dado estendido (8 bits)</b>
----------	----------	----------------	----------------------	--------------------------------

Byte 1 cabeçalho : S

Byte 2 cabeçalho : M

Byte 3 comando :

- S para potenciômetro SPI
- R para relés
- B para banco de resistores
- M para ponto de medição
- E para seleção da experiência

#### Ponto de medição (5 bytes)

<b>S</b>	<b>M</b>	<b>M</b>	<b>Bit8</b>	<b>Bit7</b>	<b>Bit6</b>	<b>Bit5</b>	<b>Bit4</b>	<b>Bit3</b>	<b>Bit2</b>	<b>Bit1</b>
----------	----------	----------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------

Byte 1 cabeçalho : S

<b>x</b>	<b>Bit 9</b>							
----------	----------	----------	----------	----------	----------	----------	----------	--------------

Byte 2 cabeçalho : M

Byte 3 comando : M (comando para ponto de medição)

Byte 4 dados :

Dados bit1 a bit8

Byte 5 dados :

Dado bit9

b1,b2,b3 seleciona pontos 1 a 8 do CH1

b4,b5,b6 seleciona pontos 1 a 8 do CH2

b7,b8,b9 seleciona pontos 1 a 8 da referencia

#### Potenciômetro Digital SPI ( 5 bytes)

S	M	S	x	x	x	X	x	E2	E1	E0
---	---	---	---	---	---	---	---	----	----	----

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

Byte 1 cabeçalho : S

Byte 2 cabeçalho : M

Byte 3 comando : S (comando para potenciômetro SPI)

Byte 4 dado :

Bits E2, E1, E0 : seleciona o endereço do potenciômetro (1 a 8)

Byte 5 dado :

Byte com dado contendo o valor a ser escrito no potenciômetro digital

#### Banco de Resistores (4 bytes)

S	M	B	B1	B0	Ch2	Ch1	Ch0	D2	D1	D0
---	---	---	----	----	-----	-----	-----	----	----	----

Byte 1 cabeçalho : S

Byte 2 cabeçalho : M

Byte 3 comando : B (comando para banco de resistores)

Byte 4 dados

**Seleção do banco de resistor**

Banco Selecionado	B1	B0
1	0	0
2	0	1
3	1	0
4	1	1

**Seleção do Bloco**

Bloco selecionado	Ch2	Ch1	Ch0
1	0	0	1
2	0	1	0
3	1	0	0

**Seleção do resistor**

Resistor selecionado	D2	D1	D0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

**Banco de Resistores (4 bytes)**

S	M	R	X	x	C0	R4	R3	R2	R1	R0
---	---	---	---	---	----	----	----	----	----	----

Byte 1 cabeçalho : S

Byte 2 cabeçalho : M

Byte 3 comando : R (comando para relés)

Byte 4 dados :

Bit C0 se 1 liga o relé, se 0 desliga o relé.

R0 a R4 endereça o relê que será acionado de 1 a 23

#### Seleção de experimento (4 bytes)

S	M	E	X	x	x	X	x	D2	D1	D0
---	---	---	---	---	---	---	---	----	----	----

Byte 1 cabeçalho : S

Byte 2 cabeçalho : M

Byte 3 comando : E (comando para seleção do experimento)

Byte 4 dados :

Bits D2, D1, D0 selecionam a experiência (1 a 7)

### 4.3 – TESTANDO COM VISUAL BASIC

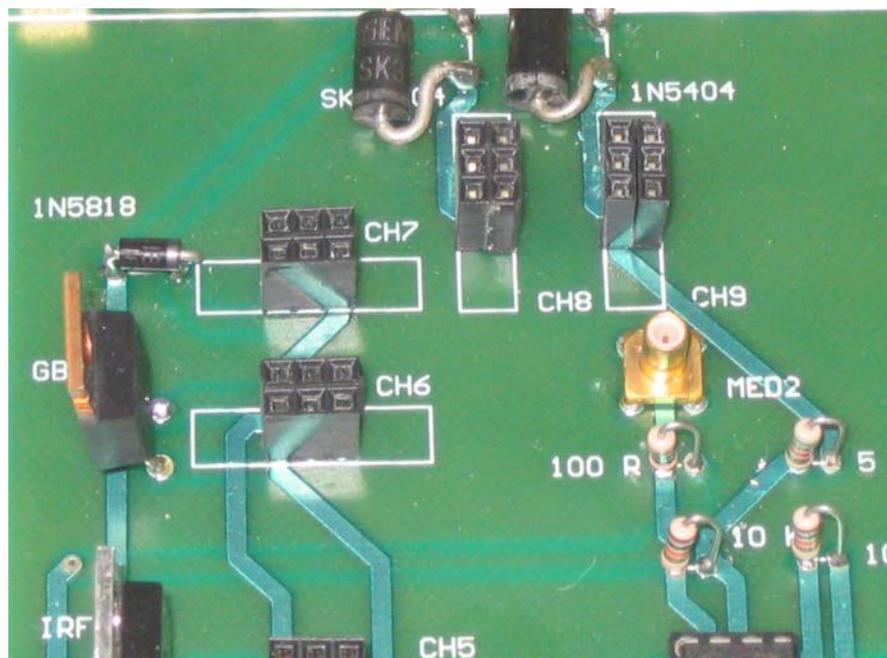
A linguagem de programação Visual Basic, como C, Delphi, LabView, permite incorporar as funções DLL's e ActiveX's, facilitando a interface com outras tecnologias.

Seguindo esta linha utilizou-se um programa básico desenvolvido em Visual Basic (**APÊNDICE E**) para testar o funcionamento da FT2XX.DLL- FT245BM - PIC rodando em PC simples com ambiente Windows ligado à rede da Unicamp.

O Programa desenvolvido em Visual Basic pretende apenas validar os testes iniciais do sistema. O objetivo principal deste trabalho não está focado no desenvolvimento de software, sendo este tópico assunto de outro trabalho. O software final está sendo desenvolvido no DEMIC/UNICAMP, levando em consideração cuidados especiais pedagógicos existentes na interface gráfica com o usuário, pois é por meio dela que ocorrerão todas as atividades do aprendiz no laboratório remoto, buscando tornar o desenvolvimento da atividade uma experiência mais próxima possível da realizada presencialmente no modo real.

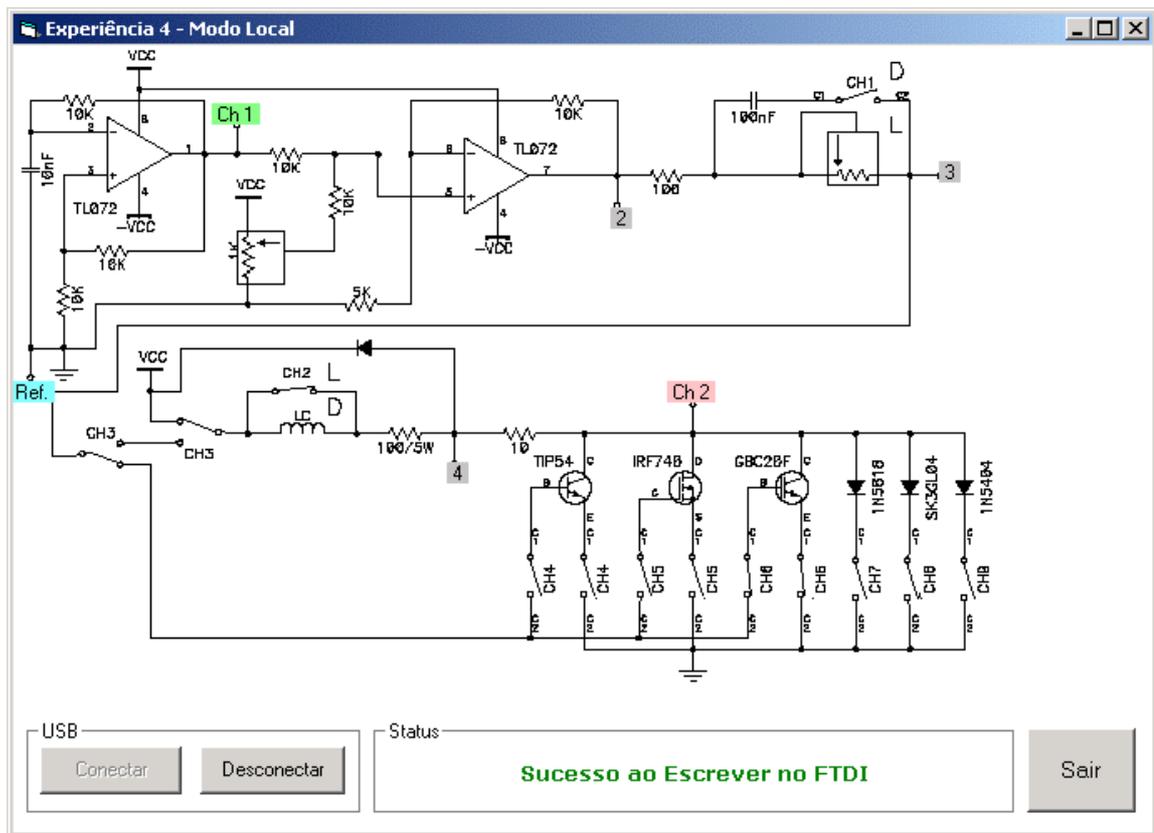


No CIRCUITO reservado para a implementação das experiências genéricas na **PLEXP** foi implementado o esquemático (FIGURA 36). Nos pontos de medição, nas chaves (CH's) e nos potenciômetros, utilizados na experiência, foram instalados conectores dedicados para o acoplamento dos módulos **BLCH**, **BLPOT** e medição, como pode ser observado na FIGURA 37 e 38.



**FIGURA 37 – Detalhe dos Conectores dos Módulos – BLCH**

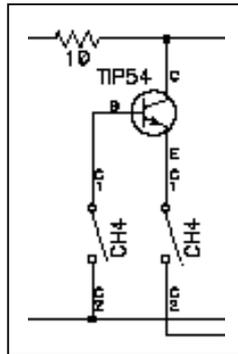




**FIGURA 39 – Software de Controle Implementado em Visual Basic para Realização da Experiência Proposta**

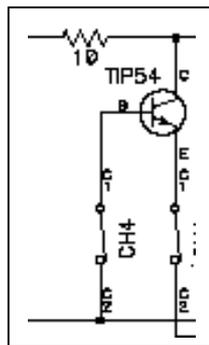
As chaves e os potenciômetros da experiência real são representados no desenho esquemático do circuito eletrônico e ao serem clicados funcionam como se o aprendiz estivesse atuando diretamente sobre o experimento convencional, ou seja os componentes serão fisicamente acionados, isto é, chaves serão ligadas ou desligadas, potenciômetros terão seu valor alterado. Para cada ação do usuário um retorno visual é apresentado, o que propicia uma representação fiel do estado do circuito que está sendo experimentado.

A FIGURA 40 mostra o detalhe de uma chave de seleção na condição de desliga. Ao clicar no desenho será enviado um comando que fecha a chave no circuito controlado pela placa de controle.



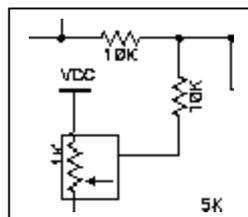
**FIGURA 40 – Detalhe da Interface Gráfica com Chave Aberta**

A seguir a interface gráfica altera o desenho da chave (FIGURA 41) para que ela represente a condição de chave fechada.



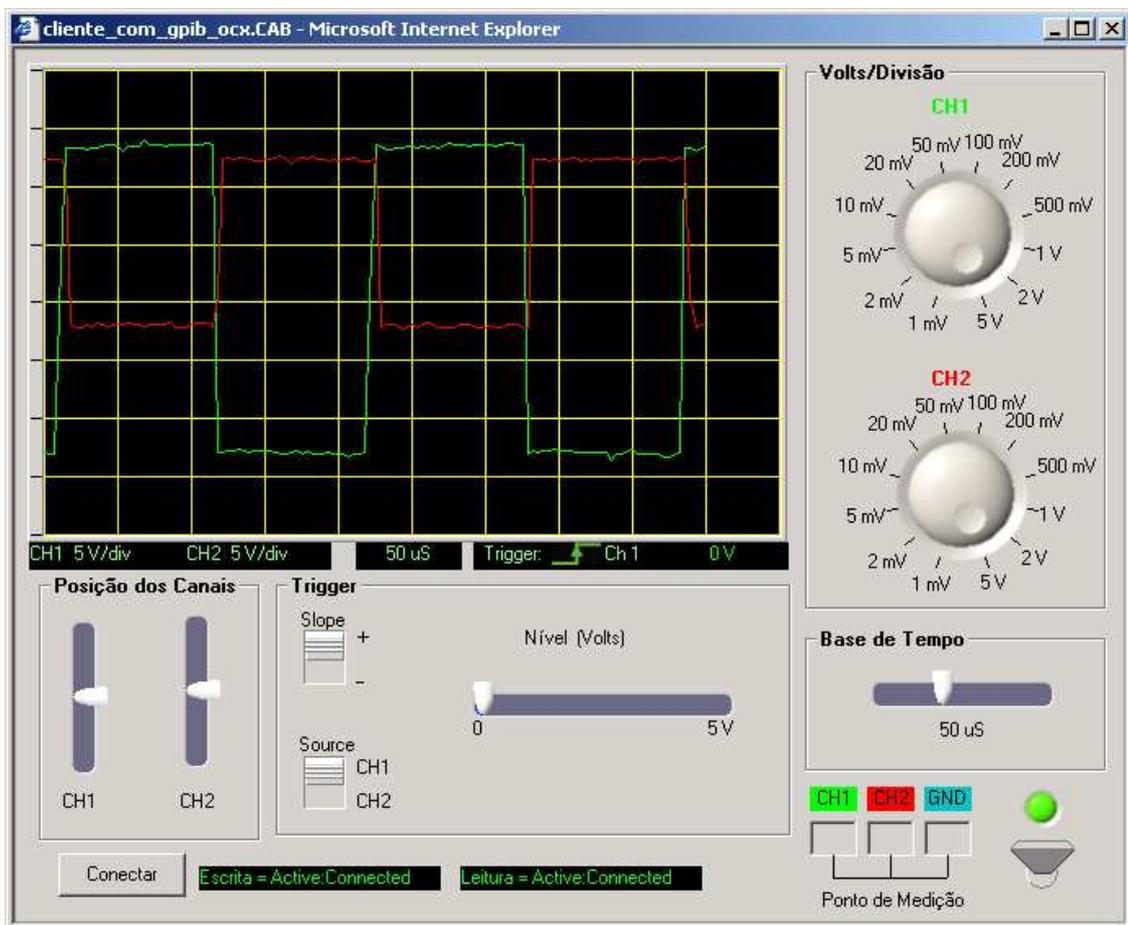
**FIGURA 41 – Detalhe da Interface Gráfica com Chave Ligada**

Para alterar o valor do potenciômetro digital o aluno deve clicar sobre o desenho e arrastar o cursor móvel que indicará a variação da resistência (FIGURA 42), a seguir é enviado pela rede o comando que irá atuar no potenciômetro correspondente da placa de controle.



**FIGURA 42 – Detalhe da Interface Gráfica com Potenciômetro**

Paralelamente a este desenvolvimento, gerou-se, utilizando Visual Basic, um osciloscópio remoto, rodando no mesmo PC em outra janela. O osciloscópio real foi conectado à placa da experiência (PLEXP) através dos cabos de entrada comum e a saída GPIB ao PC com a devida placa de comunicação. Esta metodologia foi aplicada para obter os resultados pelo trabalho, pois a implementação do osciloscópio, utilizando a comunicação USB não foi concluída.



**FIGURA 43 – Osciloscópio Desenvolvido em Visual Basic Conectado via GPIB**

O osciloscópio remoto criado para testes não contempla todas as funções necessárias de um osciloscópio real. Entretanto este desenvolvimento foi adequado para avaliar o sistema proposto, sendo seu desenvolvimento completado em trabalhos futuros.

## **5.1 – APLICAÇÃO EXPERIMENTAL**

Utilizando o mesmo roteiro da experiência convencional (APÊNDICE C), o que é uma grande vantagem do sistema proposto, pretende-se demonstrar os resultados obtidos através de observações. Não se pretende discutir eletrônica nos resultados.

## **5.2 – OBJETIVO DO EXPERIMENTO**

Caracterização dos dispositivos semicondutores rápidos de potência, através de levantamento de curvas, medições e comparações entre os componentes analisados. (diodos, transistor bipolar, MOSFET e IGBT)

## **5.3 - MATERIAL NECESSÁRIO**

- Placa de controle PLCON;
- Placa de experiência PLEXP;
- Módulos BLPOT e BLCH;
- Cabos (controle e medição);
- Osciloscópio digital (2 canais) com comunicação GPIB;
- Fonte de alimentação +/- 15V e +5V, ajustável (isolada da rede);
- Computador Pessoal com porta USB.

## 5.4 - ROTEIRO DA EXPERIÊNCIA E RESULTADOS OBTIDOS

- Verifique as tensões requeridas pelo sistema e ligue a fonte;
- Conecte o Sistema ao PC através da USB;
- Execute o software EXP4.EXE;
- No software EXP4, abra o osciloscópio.

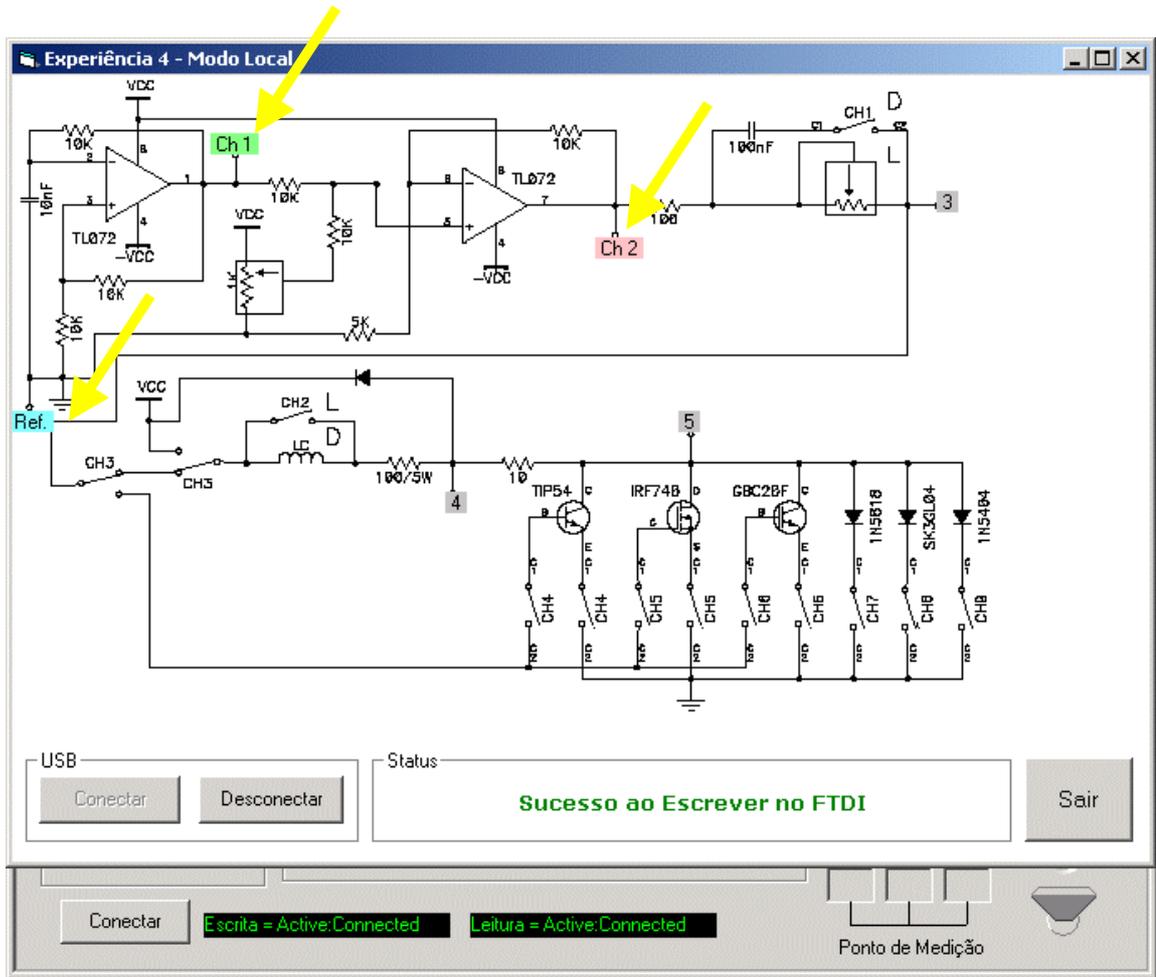
### *Observações:*

1. Para abrir ou fechar as chaves clique sobre as mesmas;
2. Para selecionar o canal desejado no ponto de medição pretendido clique sobre o mesmo (caixa de escolha).

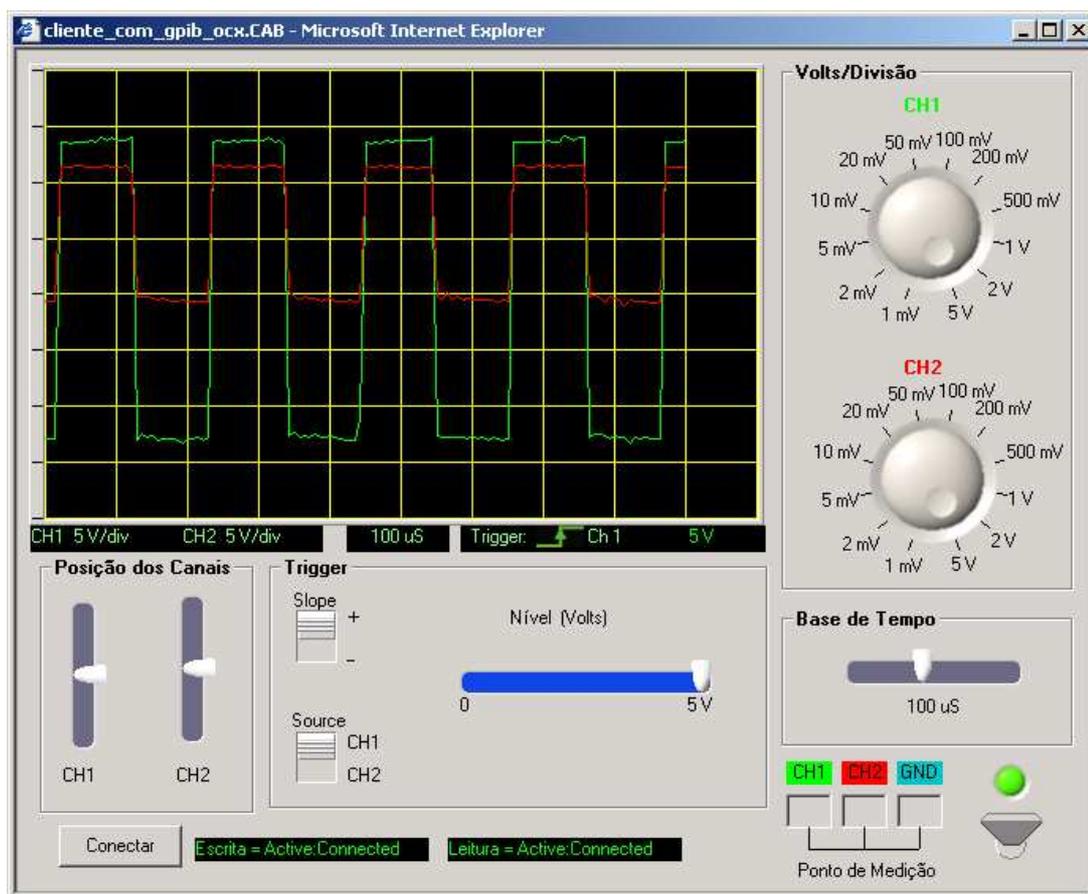
### 5.4.1 - Oscilador de onda quadrada

Para todas as medições nesta experiência, conecte ambos os terminais de terra das pontas de prova na placa de testes, a fim de minimizar o efeito das capacitâncias das pontas de prova.

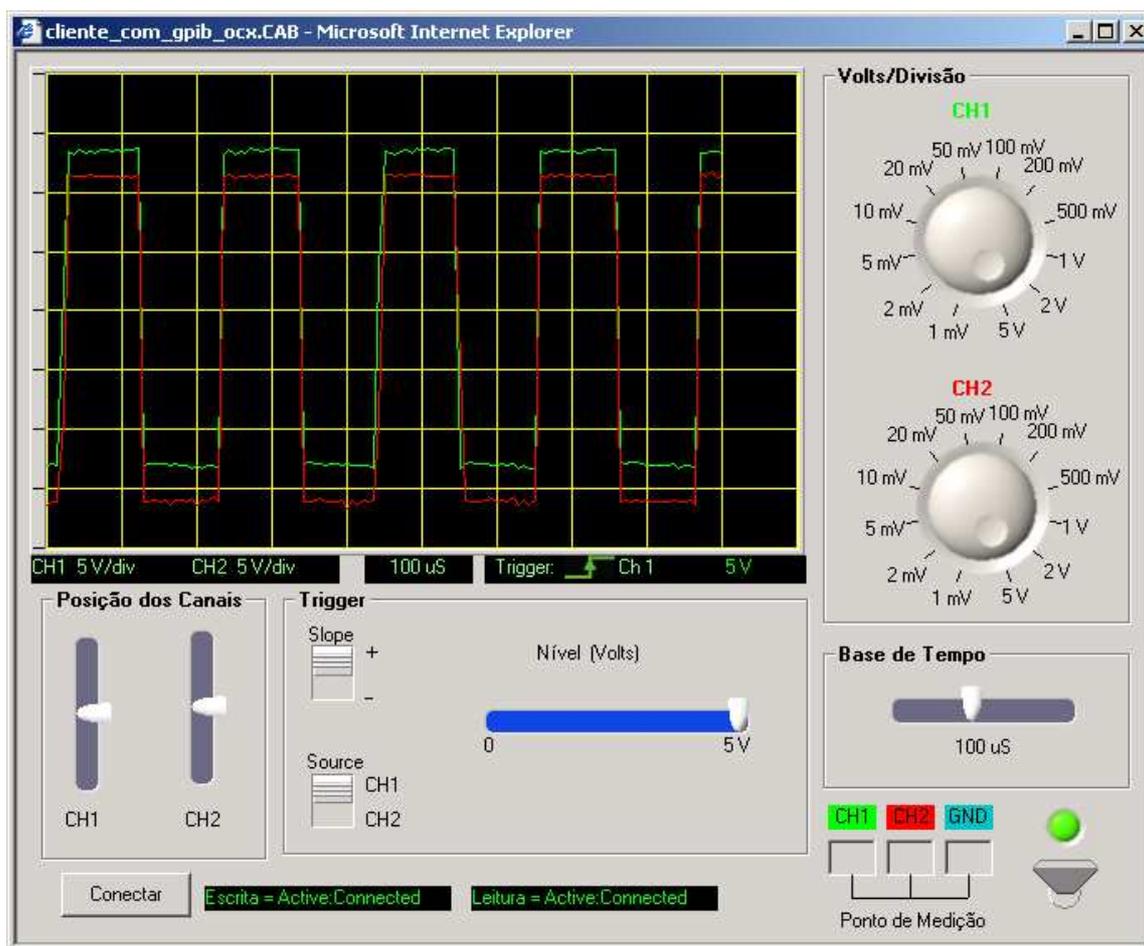
- a) Coloque todas as chaves na posição D. Alimente o circuito com uma fonte de +/- 15V
- b) Observe os pontos 1 e 2 (em relação ao 0) (**FIGURA 44**). Descreva o funcionamento do oscilador. Varie o TRIMPOT e identifique seu efeito sobre as formas de onda (**FIGURA 45 e 46**)



**FIGURA 44 – Configuração do Esquemático para Análise do Oscilador de Onda Quadrada.**



**FIGURA 45 – Resultado obtido com o Potenciômetro (BLPOT1) em VCC**



**FIGURA 46 – Resultado obtido com o Potenciômetro (BLPOT1) em GND**

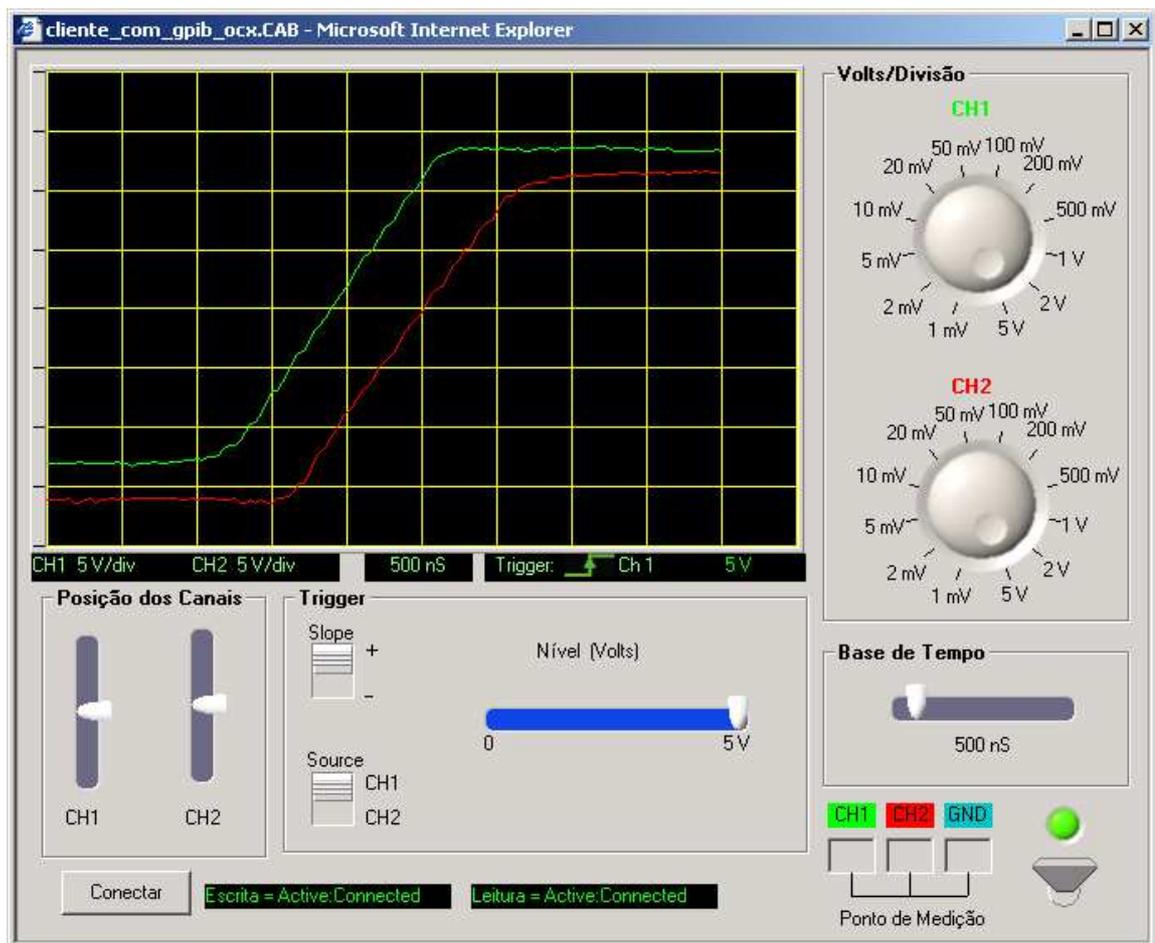
### *Observações:*

Neste item pode-se observar o funcionamento adequado dos pontos de medição (seleção), do potenciômetro (BLPOT), o tempo de resposta e qualidade das curvas obtidas.

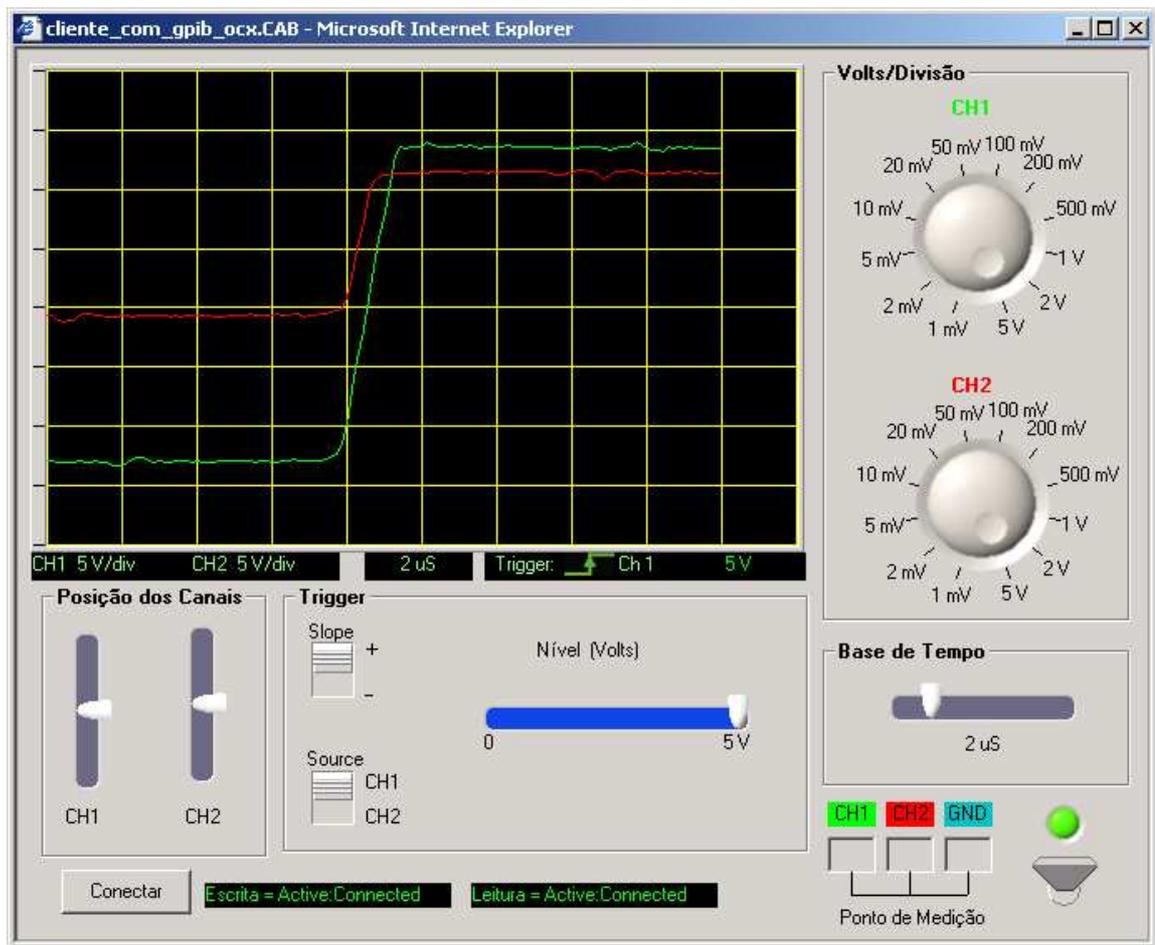
As alterações realizadas e mostradas no osciloscópio remoto foram extremamente rápidas e precisas.

Experimentalmente foi observado a soma de tensão ao ponto de inspeção 2 (curva vermelha) gerada pela variação do potenciômetro do circuito.

c) Para uma variação simétrica (em relação ao zero) da tensão no ponto 2, meça os tempos de subida e de descida, bem como os atrasos nos cruzamentos com zero (**FIGURA 47 e 48**).



**FIGURA 47 – Resultado obtido em relação ao tempo de subida**



**FIGURA 48 – Resultado obtido em relação ao tempo de descida**

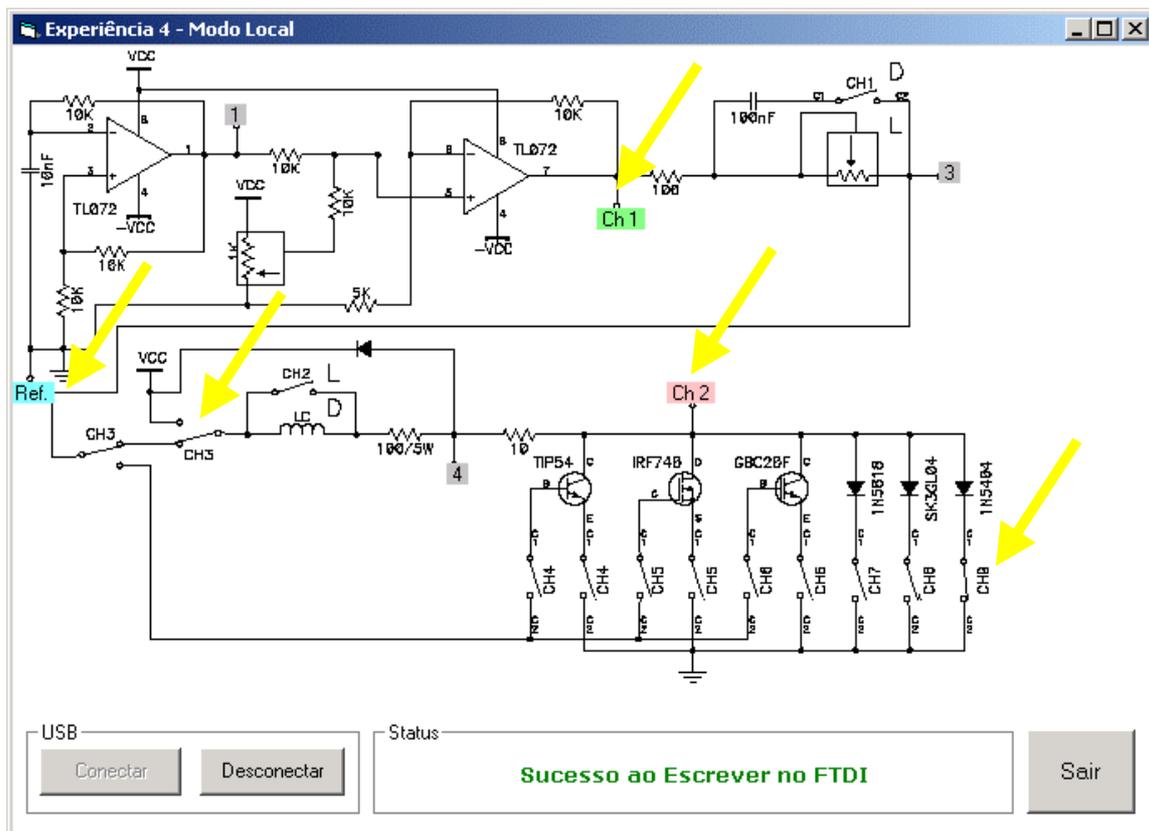
**Observações:**

Neste item pode-se observar a necessidade de funções de medição como soma, subtração, e inversão, além de cursores de marcação ao osciloscópio remoto, pois não foi possível realizar as medições solicitadas pelo roteiro da experiência de uma forma satisfatória.

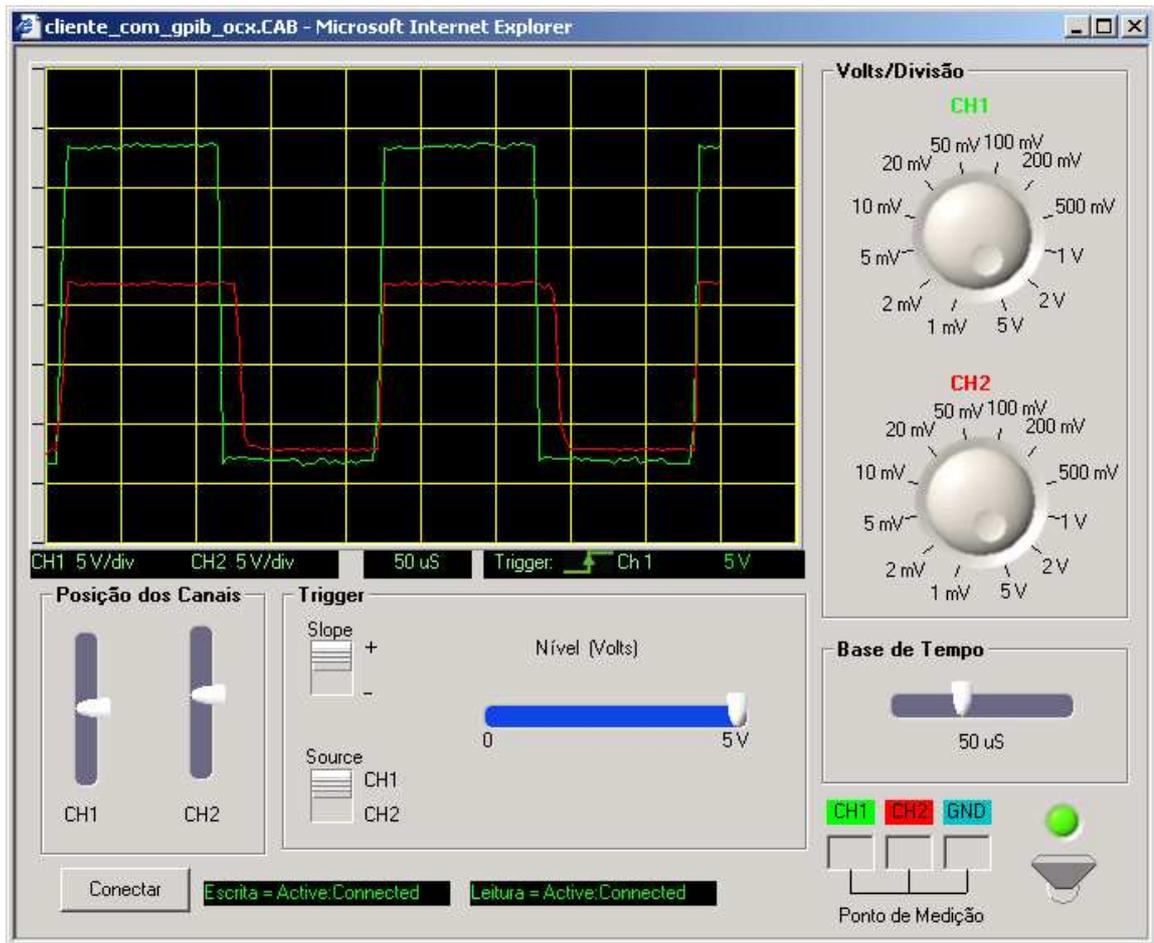
### 5.4.2 Teste do diodo 1N5404

a) Coloque a chave CH3 na posição "DIODOS". Coloque o POTENCIÔMETRO na posição de *máxima* resistência. Ligue a chave CH9 (selecione o diodo retificador 1N5404 para o teste)

b) Observando os pontos 2 e 5 em relação ao ponto 0, observe e meça, caso existam, os atrasos nas comutações do diodo. Faça a medida considerando o atraso entre os cruzamentos com o zero de ambos os sinais. (**FIGURA 49 e 50**)



**FIGURA 49 – Configuração do Esquemático Solicitado pelo item 5.1.3.21b**



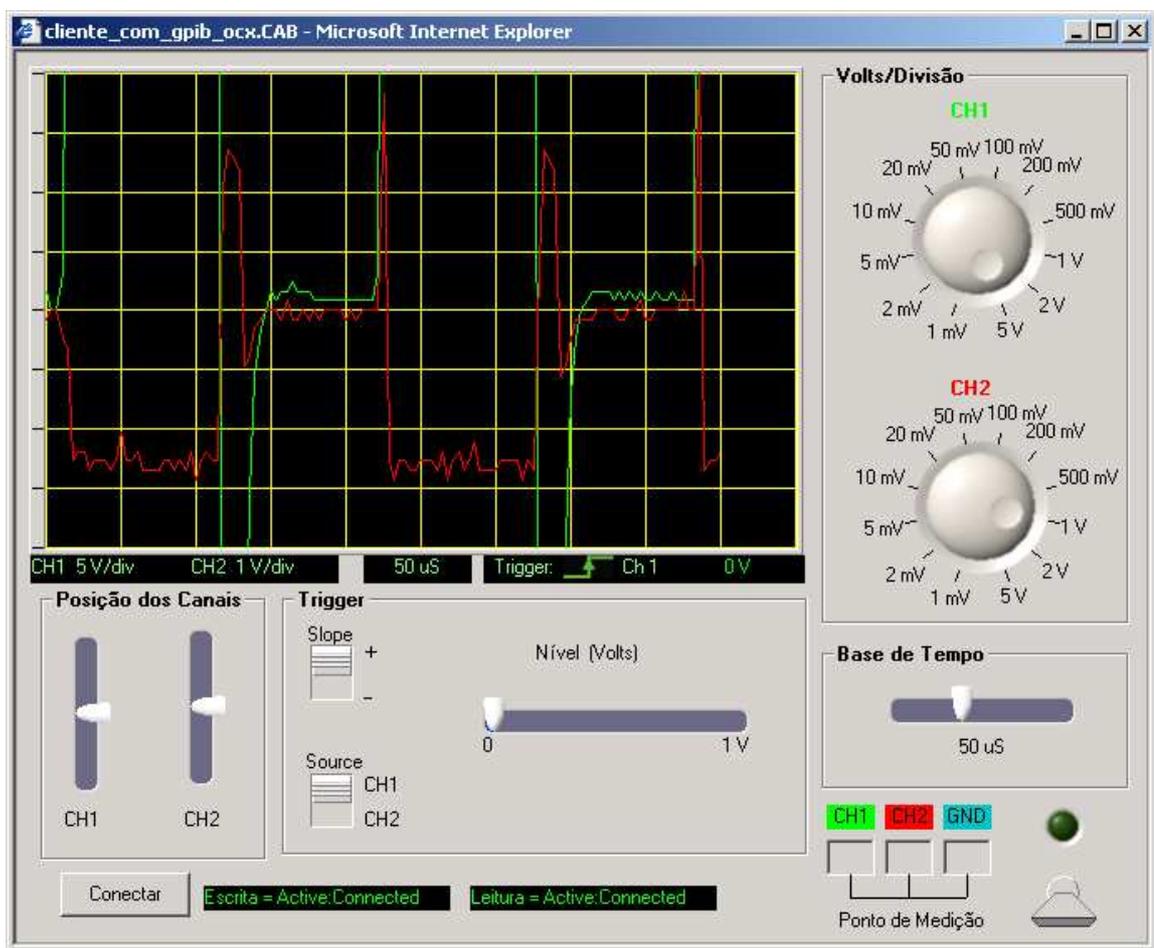
**FIGURA 50 – Resultado Obtido utilizando o Diodo 1N5404 (CH 9)**

**Observações:**

Neste item pode-se verificar o perfeito funcionamento do sistema como: a seleção do componente pela chave (BLCH), a seleção dos pontos de inspeção, tempo de resposta e curvas obtidas. Outro ponto importante observado nesta etapa foi a facilidade de conseguir “trigger” nas curvas.

Experimentalmente foi observado o funcionamento do diodo. Cortando a tensão positiva, a queda de aproximadamente 0,7V sobre o mesmo e o atraso na entrada, em condução.

c) Colocando o "terra" das pontas de prova no ponto 5, observe o ponto 4 (corrente pelo diodo) e 0 (inverso da tensão sobre o diodo). Observando as comutações, comente e justifique as formas de onda. Procure associar os fenômenos observados nos itens "b" e "c" para explicar os atrasos. (FIGURAS 51 e 52)

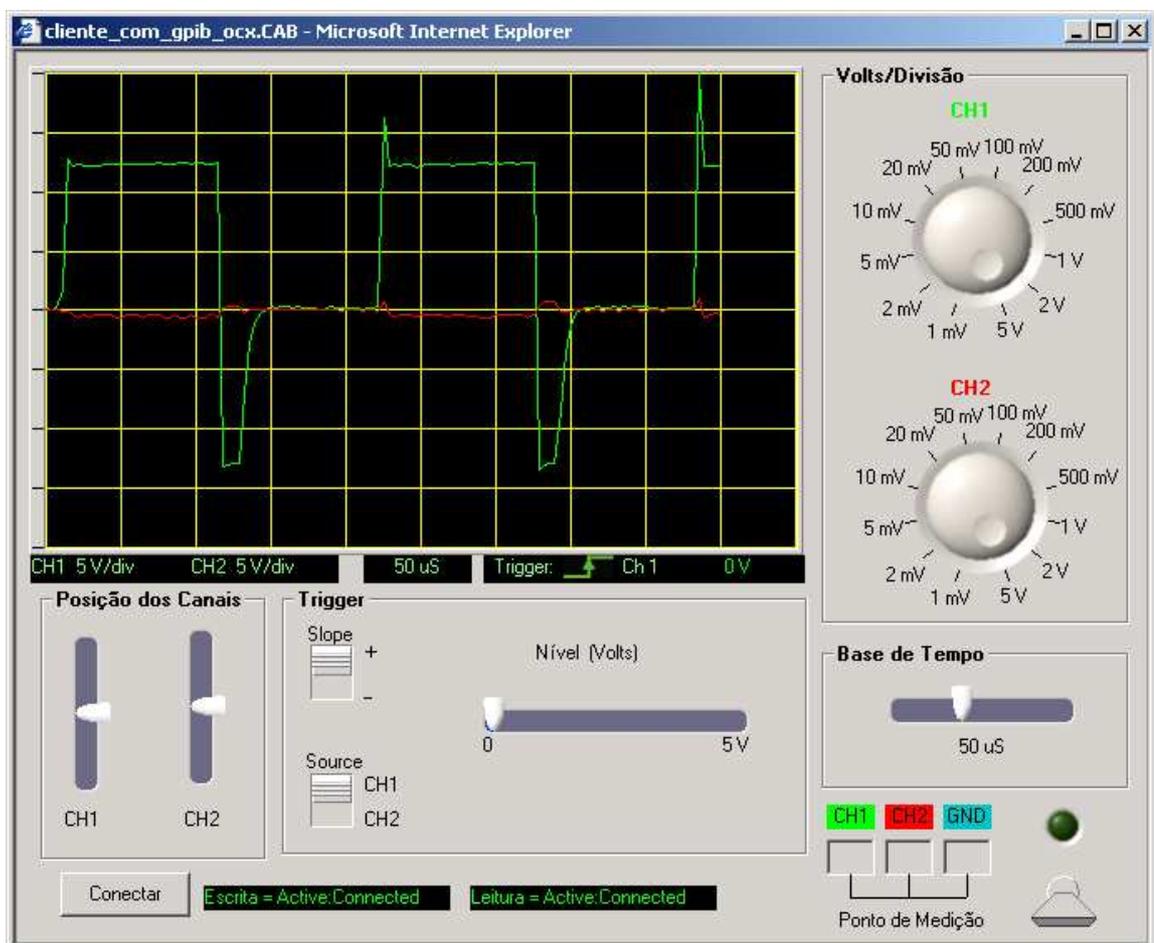


**FIGURA 51 – Resultado Obtido na medição de corrente sobre o Diodo 1N5404**

**Observações:**

As curvas obtidas apresentaram enormes distorções. A falta de isolamento da medição com a conexão da USB/PC e a resistência de  $170\ \Omega$  da chave analógica (7501) selecionadora dos pontos de medição foram responsáveis por estas distorções.

Para trabalhos futuros pode-se utilizar relés para a seleção dos pontos de medição e prever uma isolamento simples entre o PC e a conexão USB.



**FIGURA 52 – Resultado Obtido na medição de corrente sobre o Diodo 1N5404 com aterramento**

**Observações:**

Esta curva foi obtida isolando a USB do circuito, desconectando o cabo a PLCON, e aterrando o ponto de medição 5 (GND).

Experimentalmente foi observado no corte, a corrente de recombinação reversa, o pico reverso limitado pela proteção do opamp contra curto circuito.

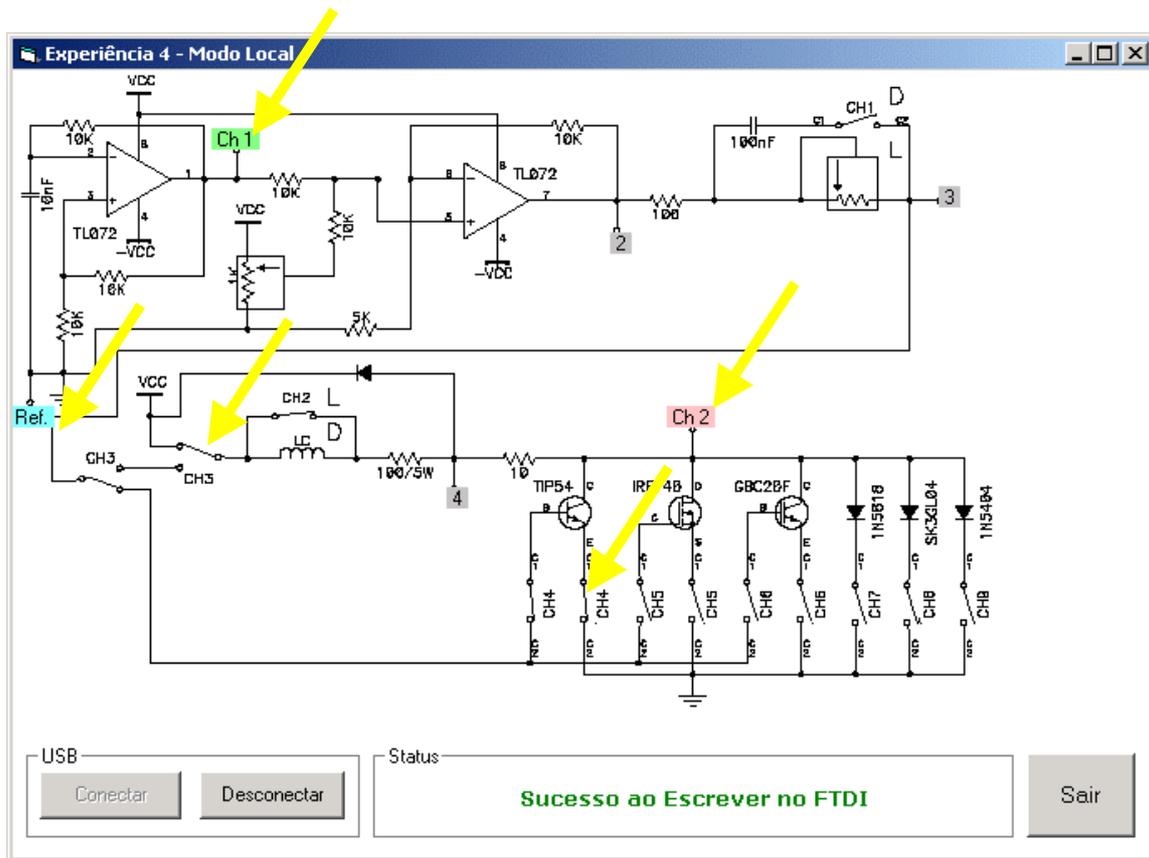
O patamar da corrente, na entrada em condução, se deve ao fato de que a tensão no ponto 5 (diodo) variar conjuntamente com o ponto 2 (saída do oscilador), produzindo uma diferença de tensão que é praticamente constante, levando a uma corrente constante que descarrega a capacitância da junção, até levar o diodo à condução.

**Nota:**

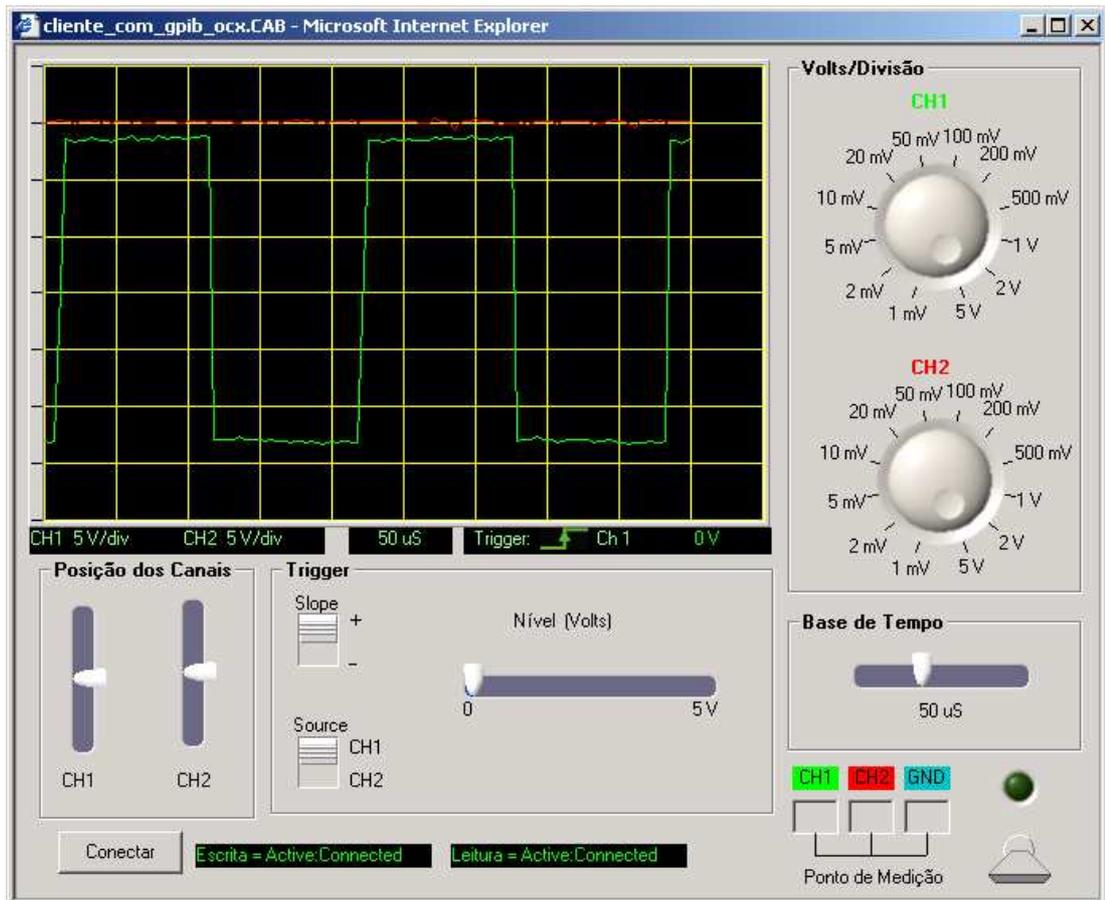
O roteiro da experiência segue os mesmos procedimentos para a análise dos outros diodos realizando medições e comparações em relação a velocidade e características.

**5.4.3 Teste do transistor TIP54**

a) Desligue todas as chaves. Passe a chave CH3 para a posição "TRANSISTORES".  
**(FIGURA 53)**



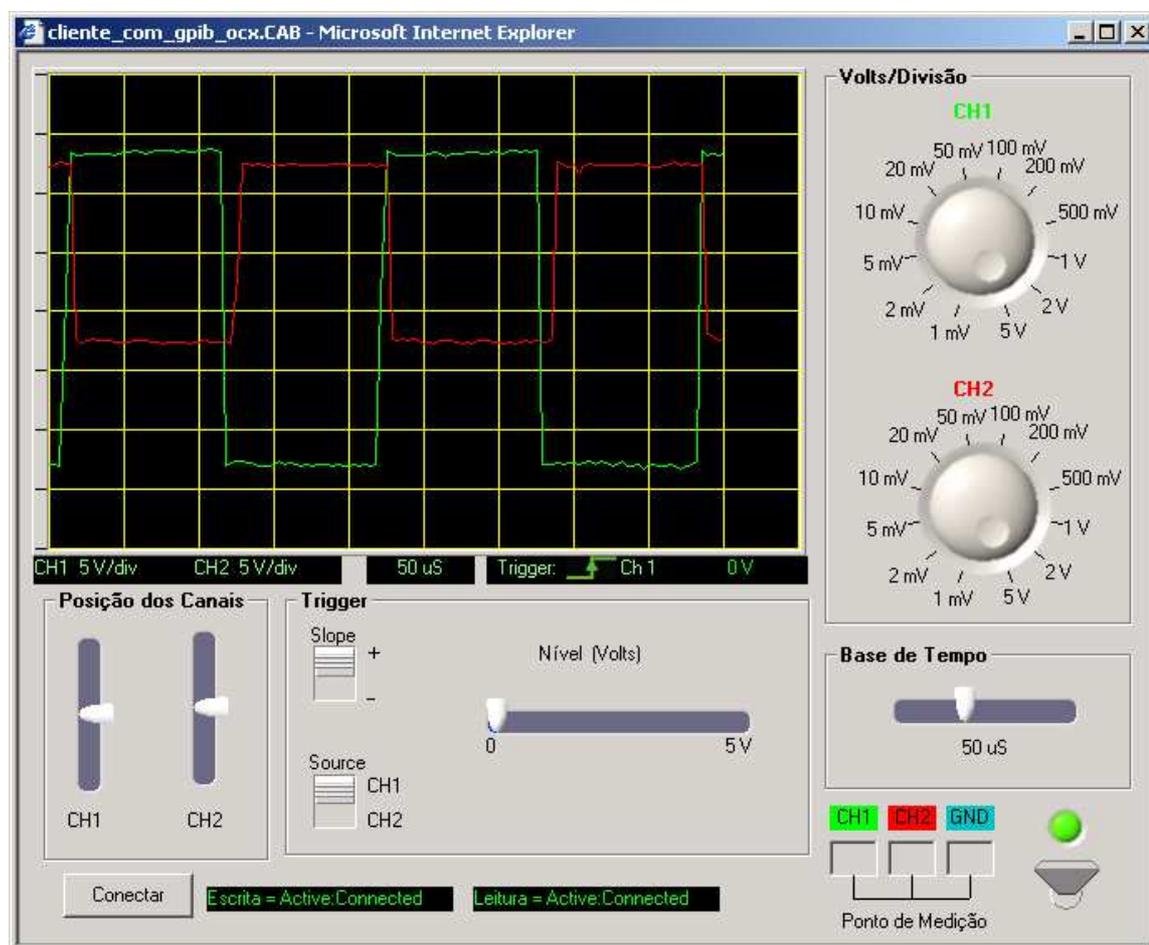
**FIGURA 53 – Configuração do Esquemático Solicitado pelo item 5.1.3.3a**



**FIGURA 54 – Resultado Obtido com CH3 na posição Transistor, sem CH4**

b) Ligue CH4, que conecta o TBP TIP54 ao circuito, com o POTENCIÔMETRO na posição de resistência máxima, observe os pontos 1 (tensão de comando) e 5 (tensão Vce ou Vds). Considere que o 2o AmpOp faz parte do circuito de acionamento e os eventuais atrasos por ele introduzidos (medidos anteriormente) ocorrem para todos os transistores analisados. **(FIGURA 55)**

c) Observe, anote e comente os atrasos para ligar e desligar o transistor. Meça os atrasos medindo o intervalo entre o início da variação da tensão de comando e o início da variação na tensão do transistor.



**FIGURA 55 – Resultado obtido com a resistência máxima (TIP 54)**

**Observações:**

Neste item pode se observar novamente o funcionamento adequado: dos pontos de medição (seleção), das chaves selecionadoras (BLCH), o tempo de resposta e qualidade das curvas obtidas.

Experimentalmente foi observado um grande atraso para o desligamento o TBP, a entrada em condução muito mais rápida e uma diminuição da tensão negativa de comando aceleram a entrada em condução.

**Nota:**

A seqüência do roteiro da experiência repete os mesmos procedimentos dos diodos para os transistores realizando medições e comparações em relação a velocidade e características do outros tipos.

Esta facilidade de comutar chaves (BLCH) faz os circuitos experimentais serem extremamente versáteis, pois é possível adicionar componentes e alterar circuitos na mesma placa (PLEXP), como exemplo deste experimento, onde iniciou-se estudando apenas o sinal de entrada, e depois, independentemente, cada componente (diodos/transistores).

---

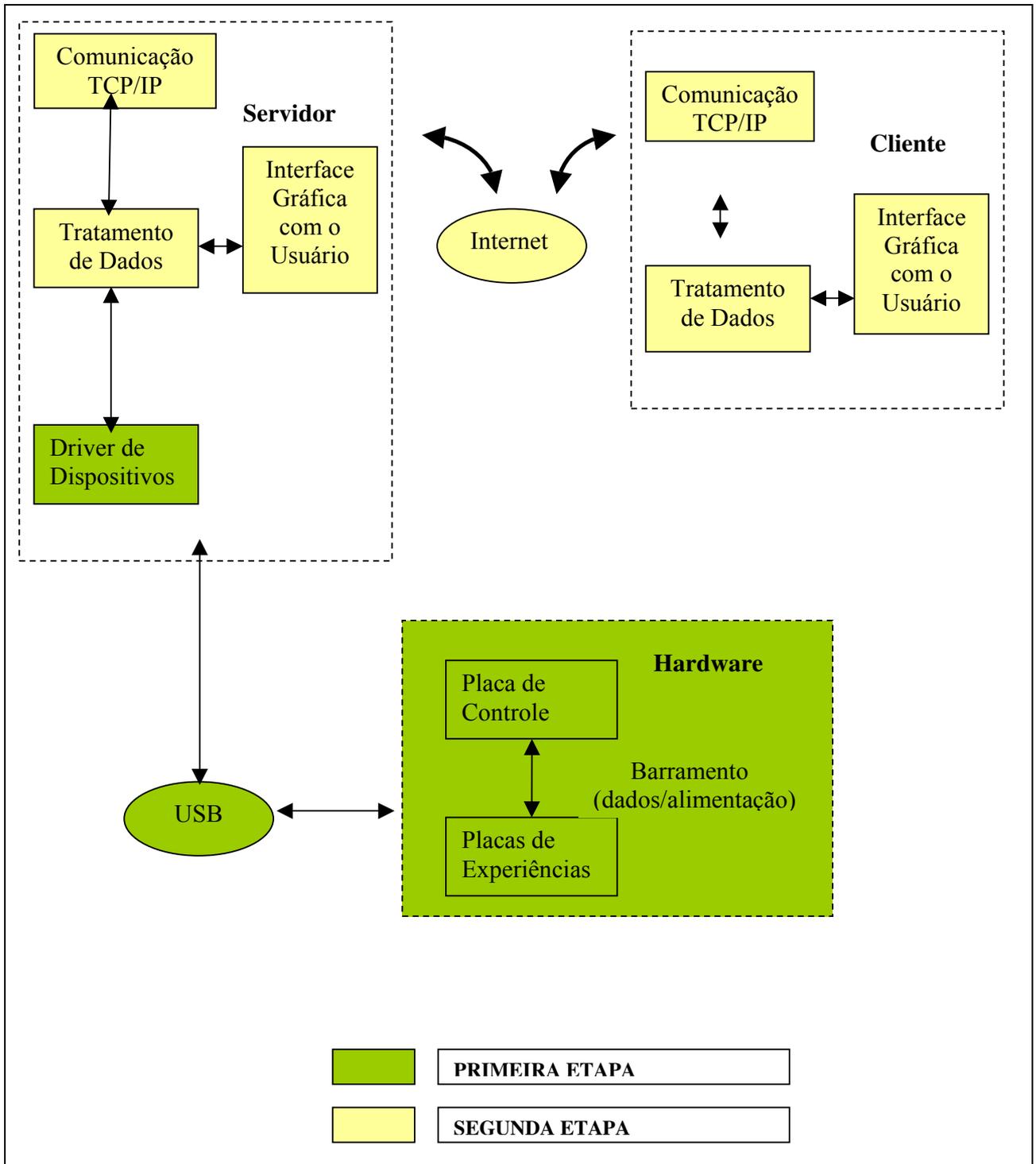
## **CAPÍTULO 6**

### **CONCLUSÃO**

O custo de material, componentes e acessórios para o primeiro protótipo do laboratório remoto ficou na ordem de R\$ 1.500,00. Deste valor, mais de 50% refere-se ao serviço de fabricação de 03 conjuntos das PLCON, PLEXP e Barramento, portanto pode-se estimar o valor total final R\$ 800,00 para 01 conjunto e acrescer para cada experiência o valor de R\$ 180,00 com os componentes do experimento. Estes valores são extremamente reduzidos em relação ao laboratório convencional, pois a experiência pode estar disponíveis por um período maior, atendendo mais alunos.

Outra característica observada, posteriormente ao início do trabalho, foi a disponibilidade do aluno ao laboratório remoto de uma forma presencial, ou seja no próprio PC onde instalados os componentes do sistema. Esta idéia poderia ser estendida para diversas máquinas com experimentos distintos. Além da possibilidade do professor realizar a experiência de uma forma dirigida, onde os alunos acompanhariam a execução dos procedimentos do experimento.

Analisando o projeto de uma forma ampla, conforme FIGURA 56, os resultados obtidos nesta primeira etapa do projeto global foram muito bons, mostrando-se altamente viável o sistema proposto. A segunda etapa, segue no desenvolvimento de ambientes amigáveis (software) para os educadores criarem os experimentos remotos, configurar a rede (WEB) e controlar e acompanhar seus alunos.



**FIGURA 56 – Arquitetura Completa do Sistema do Laboratório Remoto**

---

Este trabalho apresentou, de forma sintética, o conceito de laboratório remoto, juntamente com a implementação de um sistema físico (hardware) utilizando tecnologias baseadas em instrumentação remota para a implementação de um sistema completo de desenvolvimento de experimentos de laboratórios de eletrônica, propondo como apoio ao processo de ensino/aprendizado de engenharia elétrica.

Além disso, foi possível elaborar e implementar um Laboratório Remoto com praticamente qualquer experimento geral ou específico, utilizando uma infra-estrutura de custo relativamente baixo, permitindo a utilização dos procedimentos experimentais já existentes nos laboratórios convencionais como livros-texto, ilustrações, entre outros meios.

Finalmente esta dissertação contribuiu no desenvolvimento de pesquisas que levem à elaboração de sistema de apoio ao ensino/aprendizado, auxiliando os educadores frente a velocidade necessária aos dias atuais.

---

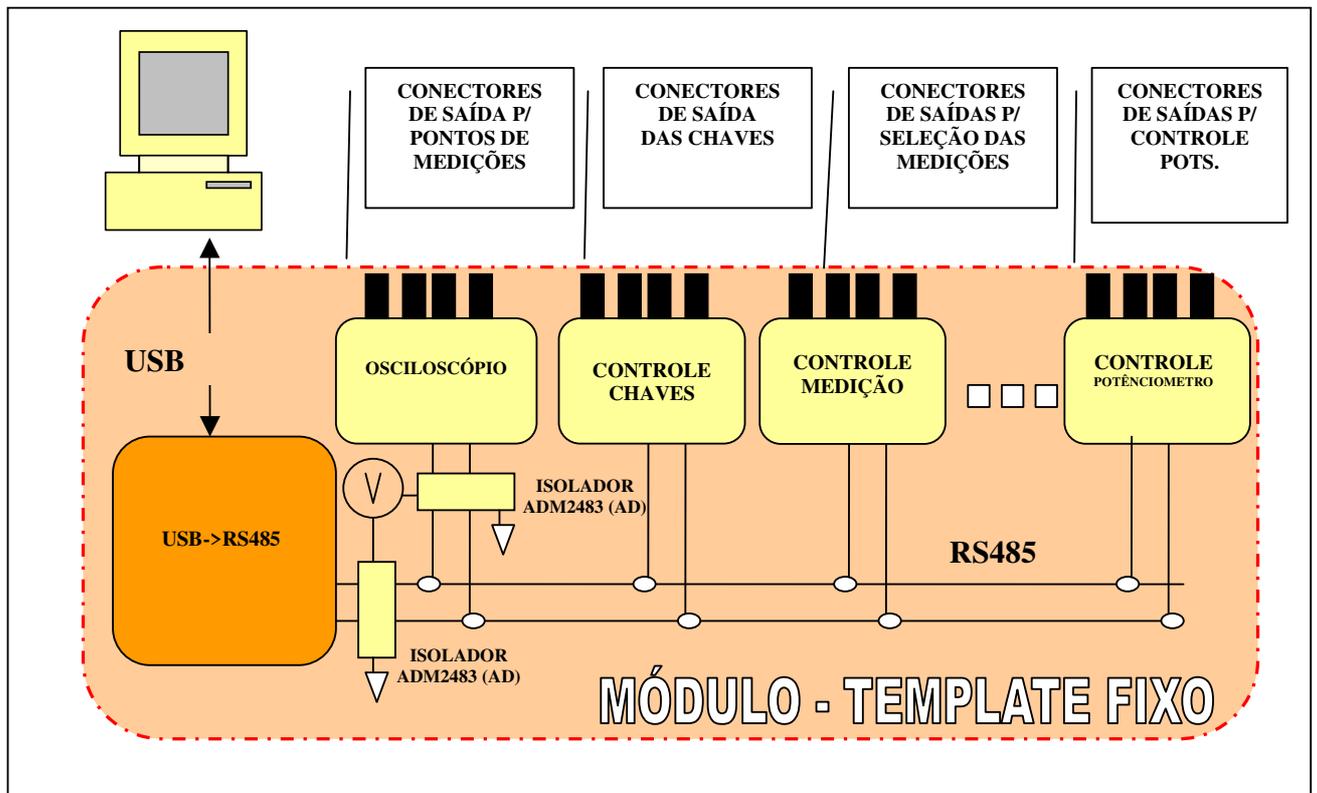
## SUGESTÕES PARA TRABALHOS FUTUROS

Todos os desenvolvimentos de novas propostas e tecnologias são realizados em etapas, com amadurecimento de idéias e conhecimentos, portanto foi de suma importância este primeiro desenvolvimento (protótipo).

Ficaram claras as melhorias que devem ser implementadas no sistema atual, como: melhoria dos conectores, cabos, suportes, footprints (marcadores de espaço para os blocos funcionais) , espaço no gabinete e acessórios em geral. Também as correções como: substituição das chaves analógicas 7501 por relés na seleção dos pontos de medição e isolamento da comunicação USB, como discutido no item RESULTADOS. E o possível emprego de novas tecnologias ainda neste protótipo, como o PIC com comunicação direta USB 2.0 e outros componentes mais recentes. Entretanto o mais importante para trabalhos futuros baseados neste desenvolvimento seriam as idéias surgidas ao longo deste projeto e descritas a seguir:

- Elaboração de um módulo compacto (500x500x200mm) contendo: conexão USB e fonte interna própria;
- Eliminação do barramento de experiência e, portanto cada módulo realizaria apenas uma experiência por vez, porém poderia ter mais módulo;
- Este módulo poderia ter acesso remoto ou local, portanto o aluno poderia utilizar o mesmo de uma forma convencional ou remotamente;
- Os experimentos poderiam ser realizados de uma forma dirigida, o professor realizaria o experimento com acompanhamento via internet e/ou localmente pelo aluno;
- Neste módulo haveria espaço para a conexão das placas de experiência, ou seja, o CIRCUITO fixo ficaria ao redor da placa do experimento;

- O módulo teria: controle para chaves, potenciômetros, resistores, osciloscópio, gerador de sinais, fonte ajustável e sinalizadores (LED's), interconectados por apenas 02 fios (RS485) controlados pela USB.
- Os blocos funcionais teriam suas dimensões reduzidas;
- Teria uma placa de experiência com protoboard acoplado para realização de experiências simples.



**FIGURA 57 – Arquitetura de Comunicação do Novo Módulo Proposto**

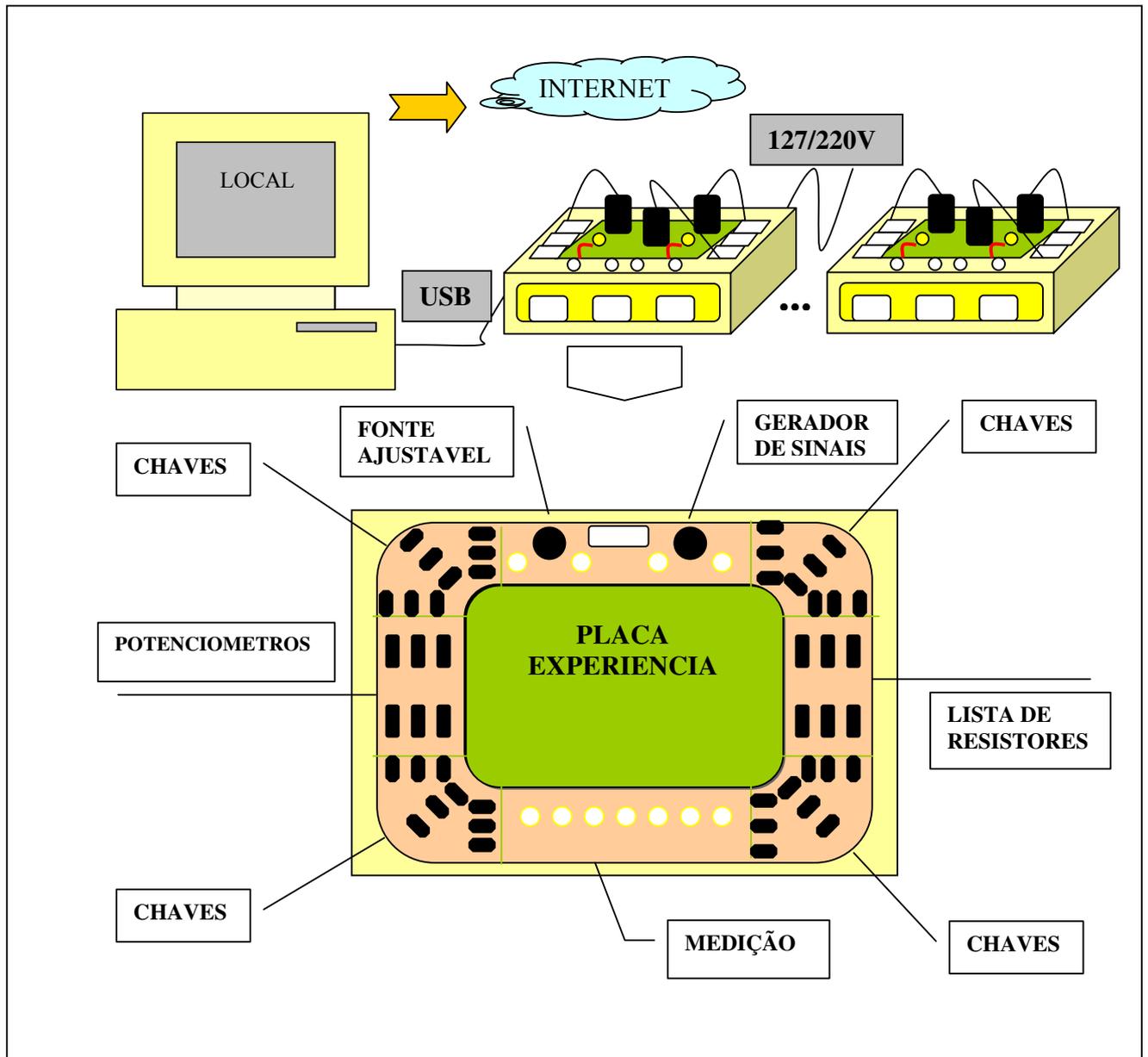


FIGURA 58 – Esquemático da Nova Proposta Sugerida

---

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] K. Jeppson, P. Lundgren, J. del Alamo, J. Hardison, D. Zych, “*iLab*”, Abril de 2004  
<http://weblab.mit.edu/>
- [2] Bagnasco Andrea, Scapolla Anna Marina, “*ISILab*”, Abril de 2004  
bagnasco, [scapolla}@dibe.unige.it](mailto:scapolla@dibe.unige.it)
- [3] **Sensores Integráveis e Microssistemas** – USP, junho de 2004,  
<http://sim.lme.usp.br>
- [4] Chella, Marco Túlio (2002) “*Ambiente de Robótica para Aplicações Educacionais com SuperLogo*”, Dissertação de Mestrado - Faculdade de Engenharia Elétrica e Computação – UNICAMP.
- [5] D’Abreu v.v.j., Chella M. T. (2003) “*Ambiente em Telerobótica em EaD*”, XIII Simpósio Brasileiro de Informática na Educação – SBC2003, Campinas SP.
- [6] Borges, Adriana Paula (2002) “*Instrumentação Virtual Aplicada a um Laboratório com Acesso Pela Internet*”, Dissertação de Mestrado – Escola Politécnica - USP.
- [7] Ferreira, Elnatan Chagas . (1998) “*Sensores e Condicionamento de Sinais*”, Faculdade de Engenharia Elétrica e Computação – DEMIC – UNICAMP,  
<http://www.demic.fee.unicamp/~elnatan>

- [8] Microchip Technology Incorporated. (2002) “*Datasheet PIC16C745*”  
<http://www.microchip.com/1010/pline/picmicro/category/perictrl/14kbytes/devices/16c745>
- [9] Souza, D.J. (2000) “*Desbravando o PIC*”, 3 ed. São Paulo: Érica, 200 p.
- [10] Aulanet (2003) Aulanet, <http://asgard.les.inf.puc-rio.br/aulanet/> - Janeiro de 2004
- [11] Casini M., Prattichizzo D. e Vicino A. (2002) “*Automatic Control Telelab: un Laboratorio Remoto per E-learning*”, IEEE
- [12] Hua, J. e Ganz, A. (2003) “*A new model for remote laboratory education based on next generation interactive technologies*”,  
<http://researchers.conferencexp.net/Lists/Research%20Papers2/Attachments/5/aseeivlab.pdf>
- [13] Lucena C., Fuks H. “*A Educação na Era da Internet. Clube do Futuro*”, Rio de Janeiro
- [14] *National Instruments*, <http://www.ni.com/>
- [15] *Teleduc* (2003) Teleduc, <http://teleduc.nied.unicamp.br/~teleduc/> Janeiro de 2004.
- [16] Stegawski, M. e Schaumann, R. (1998) “*A New Virtual Instrumentation-Based Experimenting Environment for Undergraduate Laboratories with Application in Research and Manufacturing*”, IEEE
- [17] *Future Technology Devices Intl Ltd. FTDI Chip (FT245BM and FTD2XX)*,  
<http://ftdichip.com>

[18] *Universal Serial Bus (USB)*,

<http://www.usb.org/home>

[19] *Writing DLLs and calling DLLs from LabVIEW :*

Rongen, Heinz, *Forschungszentrum Julich, Germany*

[20] *Tutorial LabVIEW*. Available in the Software version 6i

[21] Hauck, Lane, (2002) “*Embedded System Conference*”

[22] Overstreet, Jamahl and Tzes, Anthony (1999) “*An Internet-Based Real Time Control Engineering Laboratory*”, IEEE

[23] Schafer, Tilman , Seigneur, Jean e Donnelly Alexis (1999) “*PEARL – A Generic Architecture for Live Experiments in a Remote Laboratory*” Trinity College Dublin

[24] Gustavsson, Ingvar (2002) “*Remote Laboratory Experiments in Electrical Engineering Education*”, IEEE

[25] Miller Floyd, (2001) “*Virtual- Reality in Microelectronics Laboratory Instruction*”, IEEE

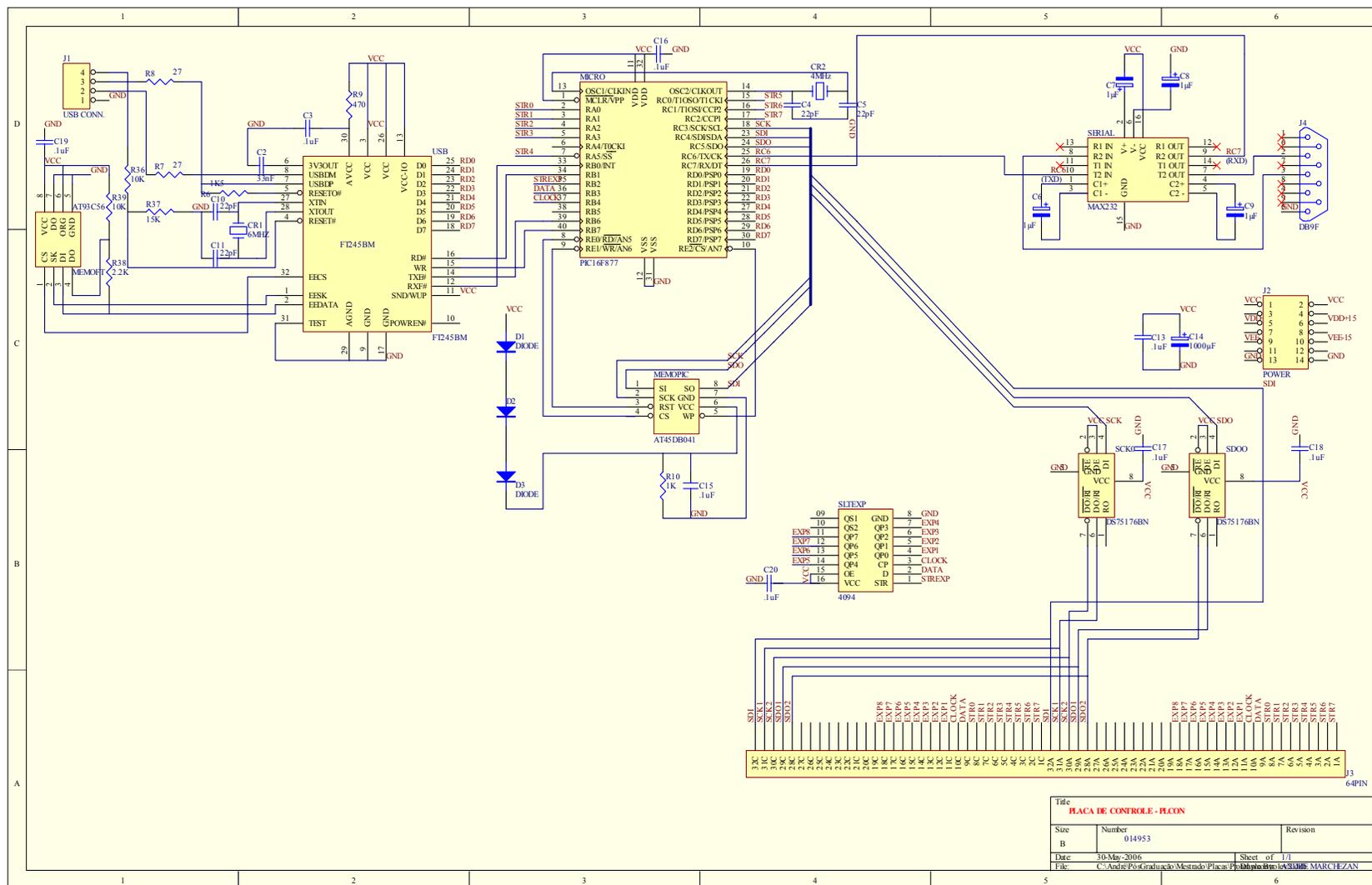
[26] Wen, Huan and Tzeng, (2000) “*Design of a Virtual Laboratory for Teaching Electric Machinery*”, IEEE

[27] Gomez, F. , Cervera, M. e Martinez, J. (2000) “*A World Wide Web Based Architecture for the Implementation of a Virtual Laboratory*”, IEEE

- 
- [28] Meliopoulos, A. E Cokkinides, G. (2000) “***Role of Laboratory Education in Power Engineering : Is the Laboratory Feasible?***”, IEEE
- [29] Rohrig, Cristof e Joheim, Andreas (1999) “***The Virtual Lab for Controlling Real Experiments via Internet***”, IEEE
- [30] Shor, Molly (2000) “***Remote- Access Engineering Education Laboratories: Who, What, When, Where, Why, and How?***”, IEEE
- [31] Aktan, Burcin e Bohus Carisa (1996) “***Distance Learning Applied to Control Engineering Laboratories***”, IEEE

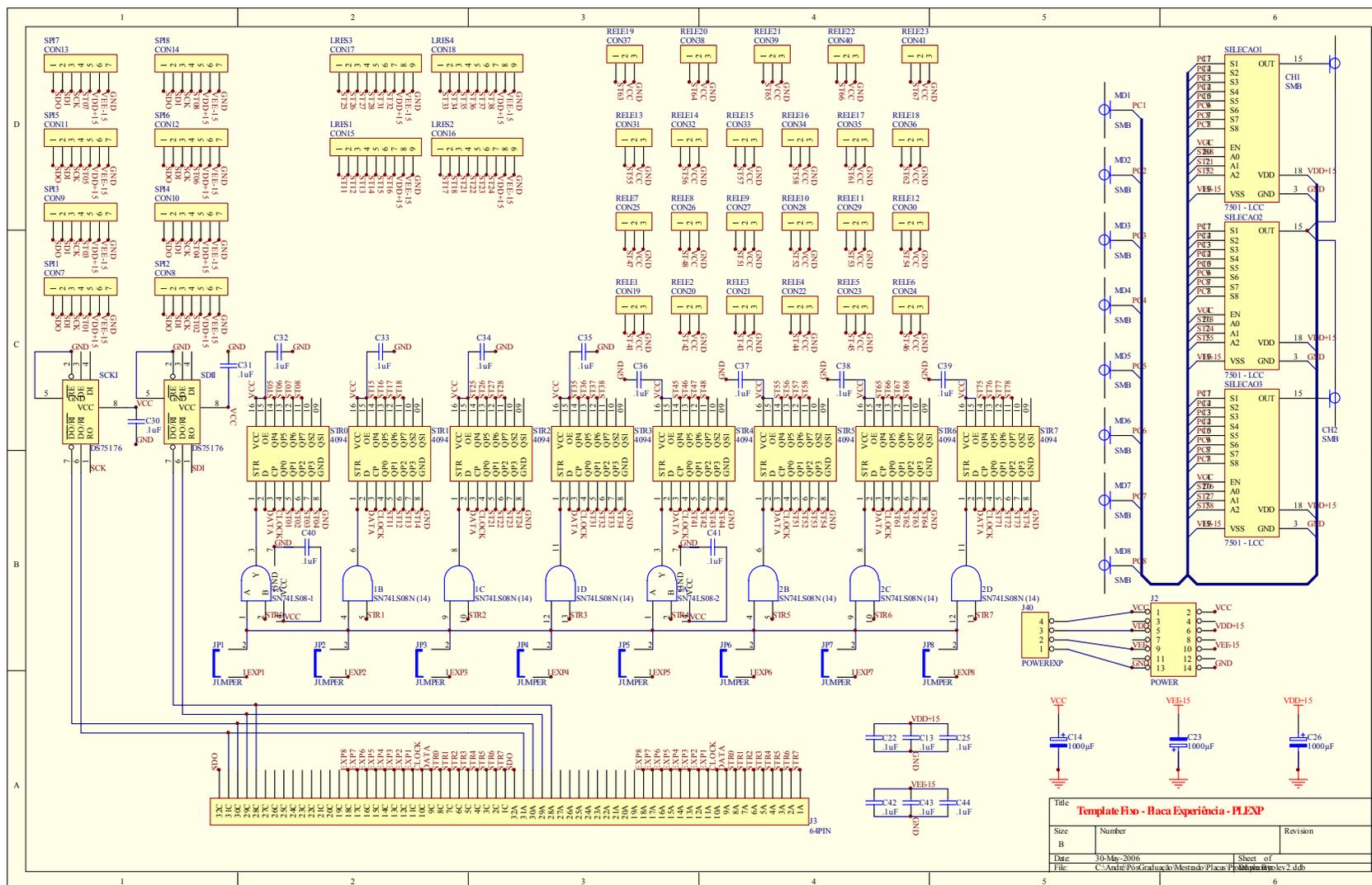
# APÊNDICE A

## ESQUEMÁTICO E LAYOUT DA PLCON



# APÊNDICE B

## ESQUEMÁTICO E LAYOUT DA PLEXP



## APÊNDICE C

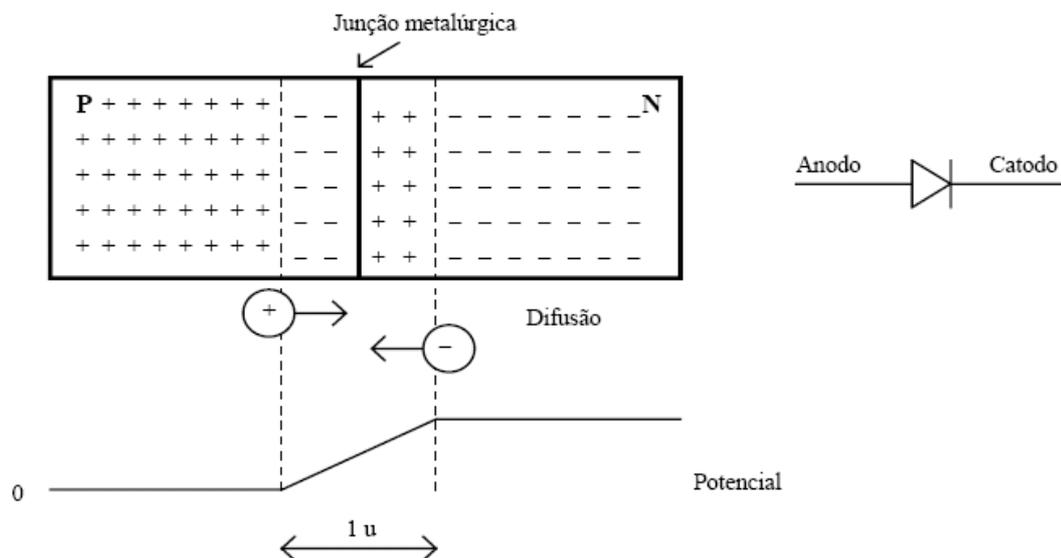
### ROTEIRO DA EXPERIÊNCIA 4 – DISCIPLINA EE832

FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO - UNICAMP  
EE 832 - LABORATÓRIO DE ELETRÔNICA INDUSTRIAL

#### EXPERIÊNCIA 4 - COMPONENTES SEMICONDUTORES RÁPIDOS DE POTÊNCIA

##### 4.1 Diodos de Potência

Um diodo semiconductor é uma estrutura P-N que, dentro de seus limites de tensão e de corrente, permite a passagem de corrente em um único sentido. Detalhes de funcionamento, em geral desprezados para diodos de sinal, podem ser significativos para componentes de maior potência, caracterizados por uma maior área (para permitir maiores correntes) e maior comprimento (a fim de suportar tensões mais elevadas). A figura 4.1 mostra, simplificada, a estrutura interna de um diodo.



**FIGURA 4.1 - Estrutura básica de um diodo semiconductor**

Aplicando-se uma tensão entre as regiões P e N, a diferença de potencial aparecerá na região de transição, uma vez que a resistência desta parte do semiconductor é muito maior que a do restante do componente (devido à concentração de portadores).

Quando se polariza reversamente um diodo, ou seja, se aplica uma tensão negativa no anodo (região P) e positiva no catodo (região N), mais portadores positivos (lacunas) migram para o lado N, e vice-versa, de modo que a largura da região de transição aumenta, elevando a barreira de potencial.

Por difusão ou efeito térmico, uma certa quantidade de portadores minoritários penetra na região de transição. São, então, acelerados pelo campo elétrico, indo até a outra região neutra do dispositivo. Esta corrente reversa independe da tensão reversa aplicada, variando, basicamente, com a temperatura.

Se o campo elétrico na região de transição for muito intenso, os portadores em trânsito obterão grande velocidade e, ao se chocarem com átomos da estrutura, produzirão novos portadores, os quais, também acelerados, produzirão um efeito de avalanche. Dado o aumento na 2 corrente, sem redução significativa na tensão na junção, produz-se um pico de potência que destrói o componente.

Uma polarização direta leva ao estreitamento da região de transição e à redução da barreira de potencial. Quando a tensão aplicada superar o valor natural da barreira, cerca de 0,7V para diodos de Si, os portadores negativos do lado N serão atraídos pelo potencial positivo do anodo e vice-versa, levando o componente à condução.

Na verdade, a estrutura interna de um diodo de potência é um pouco diferente desta apresentada. Existe uma região N intermediária, com baixa dopagem. O papel desta região é permitir ao componente suportar tensões mais elevadas, pois tornará menor o campo elétrico na região de transição (que será mais larga, para manter o equilíbrio de carga).

Esta região de pequena densidade de dopante dará ao diodo uma significativa característica resistiva quando em condução, a qual se torna mais significativa quanto maior for a tensão suportável pelo componente. As camadas que fazem os contatos externos são altamente dopadas, a fim de fazer com

que se obtenha um contato com característica ôhmica e não semicondutor (como se verá adiante nos diodos Schottky).

O contorno arredondado entre as regiões de anodo e catodo tem como função criar campos elétricos mais suaves (evitando o efeito de pontas).

No estado bloqueado, pode-se analisar a região de transição como um capacitor, cuja carga é aquela presente na própria região de transição.

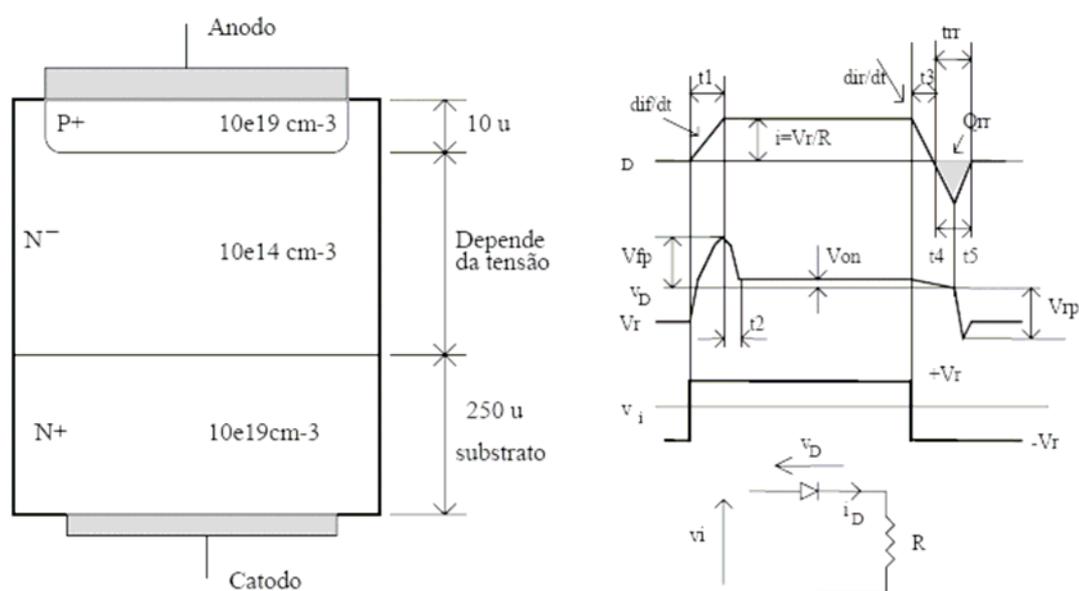
Na condução não existe tal carga, no entanto, devido à alta dopagem da camada P+, por difusão, existe uma penetração de lacunas na região N-. Além disso, à medida que cresce a corrente, mais lacunas são injetadas na região N-, fazendo com que elétrons venham da região N+ para manter a neutralidade de carga. Desta forma, cria-se uma carga espacial no catodo, a qual terá que ser removida (ou se recombinar) para permitir a passagem para o estado bloqueado do diodo. O comportamento dinâmico de um diodo de potência é, na verdade, muito diferente do de uma chave ideal, como se pode observar na figura 4.2. Suponha-se que se aplica uma tensão  $v_i$  ao diodo, alimentando uma carga resistiva (cargas diferentes poderão alterar alguns aspectos da forma de onda).

Durante  $t_1$ , remove-se a carga acumulada na região de transição. Como ainda não houve significativa injeção de portadores, a resistência da região N- é elevada, produzindo um pico de tensão. Indutâncias parasitas do componente e das conexões também colaboram com a sobretensão. Durante  $t_2$  tem-se a chegada dos portadores e a redução da tensão para cerca de 1V.

Estes tempos são, tipicamente, da ordem de centenas de ns. No desligamento, a carga espacial presente na região N- deve ser removida antes que se possa reiniciar a formação da barreira de potencial na junção. Enquanto houver portadores transitando, o diodo se mantém em condução. A redução em Von se deve à diminuição da queda ôhmica. Quando a corrente atinge seu pico negativo é que foi retirado o excesso de portadores, iniciando-se, então, o bloqueio do diodo. A taxa de variação da corrente, associada às indutâncias do circuito, provoca uma sobretensão negativa.

Diodos rápidos possuem  $t_{rr}$  da ordem de, no máximo, poucos microsegundos, enquanto nos diodos normais é de dezenas ou centenas de microsegundos.

O retorno da corrente a zero, após o bloqueio, devido à sua elevada derivada e ao fato de, neste momento, o diodo já estar desligado, é uma fonte importante de sobretensões produzidas por indutâncias parasitas associadas aos componentes por onde circula tal corrente. A fim de minimizar este fenômeno foram desenvolvidos os diodos “soft-recovery”, nos quais esta variação de corrente é suavizada, reduzindo os picos de tensão gerados.



**FIGURA 4.2 - Estrutura típica de diodo de potência e formas de onda típicas de comutação**

## 4.2 Diodos Schottky

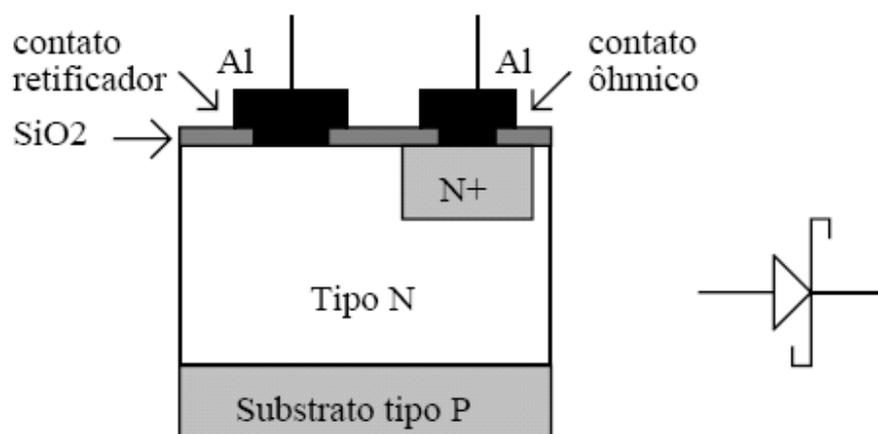
Quando é feita uma junção entre um terminal metálico e um material semiconductor, o contato tem, tipicamente, um comportamento ôhmico, ou seja, a resistência do contato governa o fluxo da corrente. Quando este contato é feito entre um metal e uma região semicondutora com densidade de dopante relativamente baixa, o efeito dominante deixa de ser o resistivo, passando a haver também um efeito retificador.

Um diodo Schottky é formado colocando-se um filme metálico em contato direto com um semicondutor, como indicado na figura 4.3. O metal é usualmente depositado sobre um material tipo N, por causa da maior mobilidade dos portadores neste tipo de material. A parte metálica será o anodo e o semicondutor, o catodo.

Numa deposição de Al (3 elétrons na última camada), os elétrons do semicondutor tipo N migrarão para o metal, criando uma região de transição na junção.

Note-se que apenas elétrons (portadores majoritários em ambos materiais) estão em trânsito. O seu chaveamento é muito mais rápido do que o dos diodos bipolares, uma vez que não existe carga espacial armazenada no material tipo N, sendo necessário apenas refazer a barreira de potencial (tipicamente de 0,3V). A região N tem uma dopagem relativamente alta, a fim de reduzir as perdas de condução, com isso, a máxima tensão suportável por estes diodos é de cerca de 100V.

A aplicação deste tipo de diodos ocorre principalmente em fontes de baixa tensão, nas quais as quedas sobre os retificadores são significativas.



**FIGURA 4.3 - Diodo Schottky construído através de técnica de CIs.**

### 4.3 Transistor Bipolar de Potência (TBP)

Os TBP foram os primeiros componentes controláveis no disparo e no desligamento capazes de manobrar níveis relativamente elevados de potência.

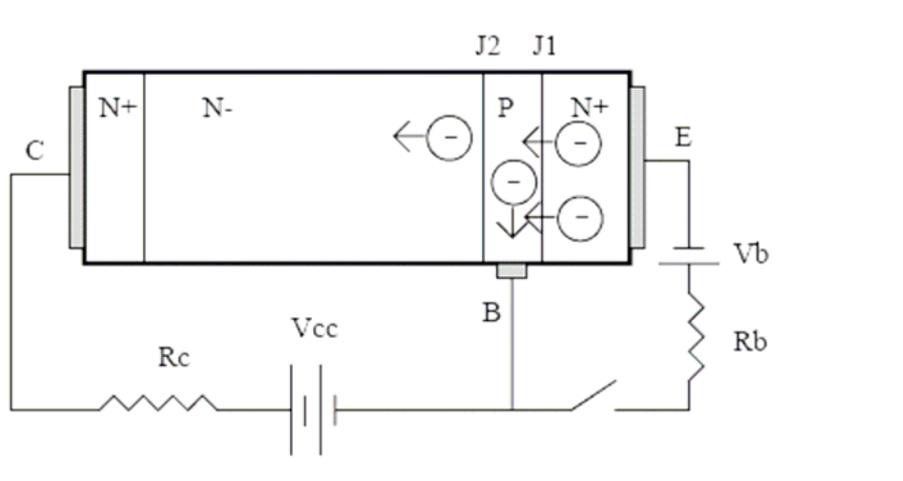
Embora a tecnologia dos tiristores já fosse relativamente bem dominada, o impedimento para a utilização dos transistores em aplicações de potência era a baixa velocidade de comutação, ou seja, a "lentidão" com que o componente, ao ser ligado ou desligado, atravessava a sua região ativa, o que implicava em excessiva dissipação de potência sobre o mesmo. Quando as melhorias tecnológicas permitiram realizar tal operação em tempos da ordem de poucos microssegundos, os TBP começaram a ser aplicados em conversores eletrônicos de potência.

Posteriormente, com as evoluções tecnológicas que permitiram o desenvolvimento de outros tipos de transistores, como os MOSFETs e IGBTs (que serão vistos adiante), os TBPs perderam competitividade e hoje praticamente não são mais utilizados.

Apresentamos na seqüência algumas características de tais componentes a título de informação.

#### 4.3.1 Princípio de funcionamento

A figura 4.4 mostra a estrutura básica de um transistor bipolar.



**FIGURA 4.4 - Estrutura básica de transistor bipolar**

A operação normal de um transistor é feita com a junção J1 (B-E) diretamente polarizada, e com J2 (B-C) reversamente polarizada.

No caso NPN, os elétrons são atraídos do emissor pelo potencial positivo da base. Esta camada central é suficientemente fina para que a maior parte dos portadores tenha energia cinética suficiente para atravessá-la, chegando à região de transição de J2, sendo, então, atraídos pelo potencial positivo do coletor. O controle de  $V_{be}$  determina a corrente de base,  $I_b$ , que, por sua vez, se relaciona com  $I_c$  pelo ganho de corrente do dispositivo.

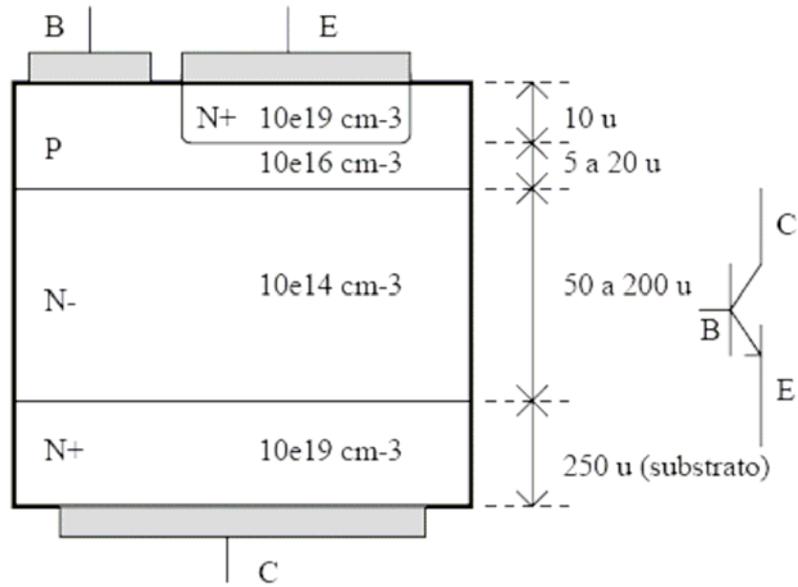
Na realidade, a estrutura interna dos TBPs é diferente. Para suportar tensões elevadas, existe uma camada intermediária do coletor, com baixa dopagem, a qual define a tensão de bloqueio do componente. A figura 4.5. mostra uma estrutura típica de um transistor bipolar de potência. As bordas arredondadas da região de emissor permitem uma homogeneização do campo elétrico, necessária à manutenção de ligeiras polarizações reversas entre base e emissor. O TBP não sustenta tensão no sentido oposto porque a alta dopagem do emissor provoca a ruptura de J1 em baixas tensões (5 a 20V).

O uso preferencial de TBP tipo NPN se deve às menores perdas em relação aos PNP, o que ocorre por causa da maior mobilidade dos elétrons em relação às lacunas, reduzindo, principalmente, os tempos de comutação do componente.

#### **4.3.2 Área de Operação Segura (AOS)**

A AOS representa a região do plano  $V_{ce} \times I_c$  dentro da qual o TBP pode operar sem se danificar. A figura 4.7 mostra uma forma típica de AOS.

À medida que a corrente se apresenta em pulsos (não-repetitivos) a área se expande. Para pulsos repetitivos deve-se analisar o comportamento térmico do componente para se saber se é possível utilizá-lo numa dada aplicação, uma vez que a AOS, por ser definida para um único pulso, é uma restrição mais branda. Esta análise térmica é feita com base no ciclo de trabalho a que o dispositivo está sujeito, aos valores de tensão e corrente e à impedância térmica do transistor, a qual é fornecida pelo fabricante.



**FIGURA 4.5 - Estrutura interna de TBP e seu símbolo**

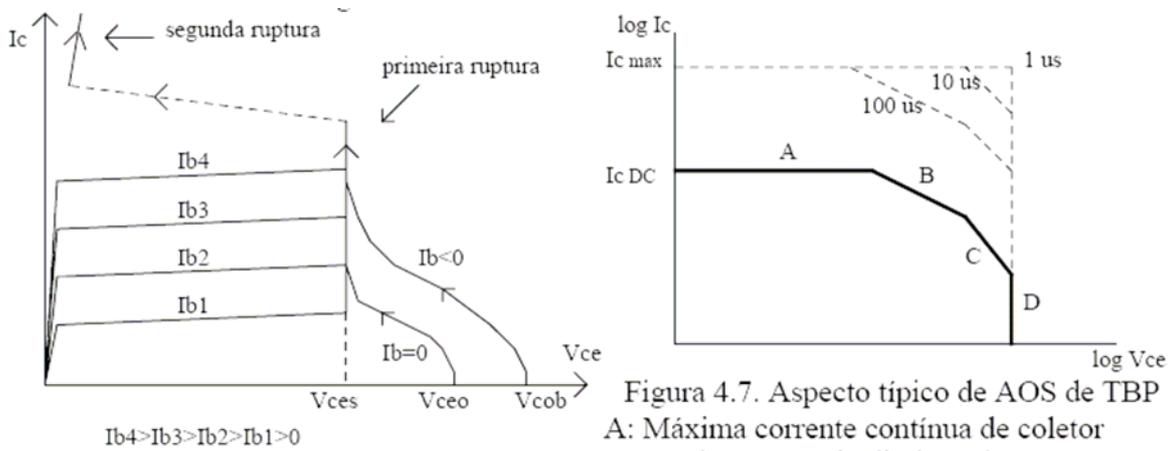


Figura 4.6 Característica estática de transistor bipolar.

Figura 4.7. Aspecto típico de AOS de TBP  
 A: Máxima corrente contínua de coletor  
 B: Máxima potência dissipável  
 C: Limite de segunda ruptura  
 D: Máxima tensão Vce

**FIGURA 4.6 -Característica estática de transistor**

### 4.3.3 Região de quase-saturação

Consideremos o circuito mostrado na figura 4.8, e as curvas estáticas do TBP ali indicadas.

Quando  $I_c$  cresce,  $V_{ce}$  diminui, dada a maior queda de tensão sobre  $R$ . À medida que  $V_{ce}$  se reduz, caminha-se no sentido da saturação.

Os TBP apresentam uma região chamada de quase-saturação gerada, principalmente, pela presença da camada N- do coletor.

À semelhança da carga espacial armazenada nos diodos, nos transistores bipolares também ocorre armazenamento de carga. A figura 4.9 mostra a distribuição de carga estática no interior do transistor para as diferentes regiões de operação.

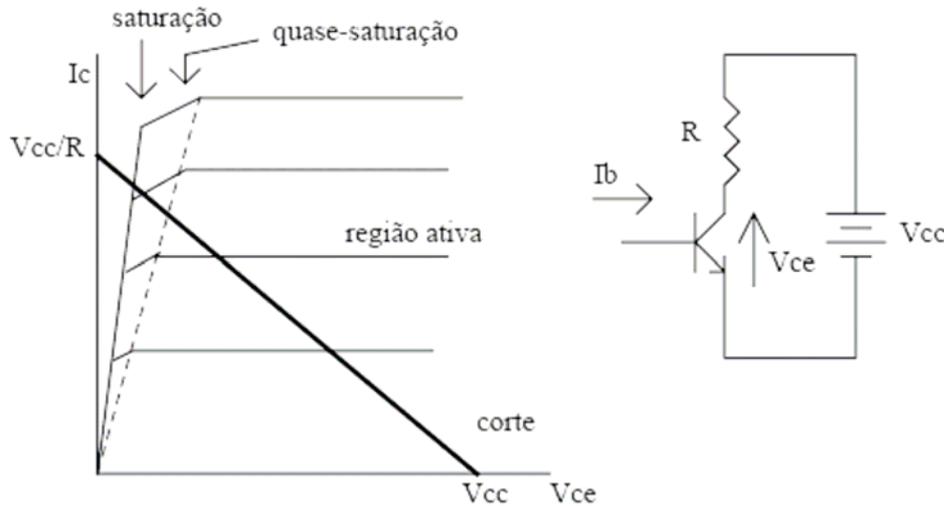
Na região ativa,  $J_2$  está reversamente polarizada e ocorre uma acumulação de elétrons na região da base. Quando se aproxima da saturação,  $J_2$  fica diretamente polarizada, atraindo lacunas da base para o coletor. Tais lacunas associam-se a elétrons vindos do emissor e que estão migrando pelo componente, criando uma carga espacial que penetra a região N-. Isto representa um "alargamento" da região da base, implicando na redução do ganho do transistor. Tal situação caracteriza a chamada quase-saturação. Quando esta distribuição de carga espacial ocupa toda a região N- chega-se, efetivamente, à saturação.

É claro que no desligamento toda esta carga terá que ser removida antes do efetivo bloqueio do TBP, o que sinaliza a importância do ótimo circuito de acionamento de base para que o TBP possa operar numa situação que minimize a tempo de desligamento e a dissipação de potência (associada ao valor de  $V_{ce}$ ). O tempo necessário para a extração desta carga é chamado "tempo de armazenamento".

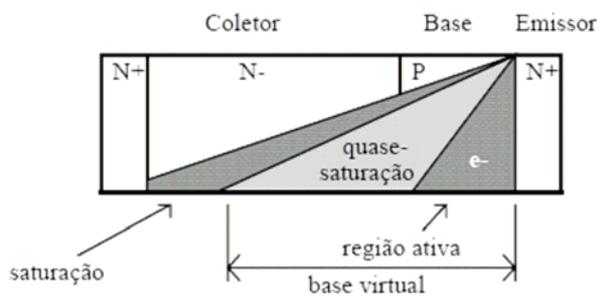
### 4.3.4 Forma da corrente de base

Um ponto básico para obter rápidos chaveamentos é utilizar uma corrente de base adequada, como mostra a figura 4.10.

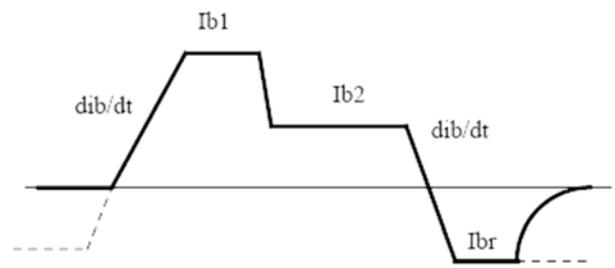
As transições devem ser rápidas, para reduzir os tempo de atraso. Um valor elevado  $I_{b1}$  permite uma redução no tempo de crescimento da corrente. Quando em condução,  $I_{b2}$  deve ter tal valor que faça o TBP operar na região de quase-saturação. No desligamento, deve-se prover uma corrente negativa, acelerando assim a retirada dos portadores armazenados.



**Figura 4.8 - Região de quase-saturação do TBP**



**Figura 4.9 Distribuição da carga estática acumulada no TBP**



**Figura 4.10 Forma de onda de corrente de base recomendada para acionamento de TBP.**

## 4.4 MOSFET

### 4.4.1 Princípio de funcionamento (canal N)

O terminal de gate é isolado do semiconductor por SiO<sub>2</sub>. A junção PN- define um diodo entre Source (Fonte) e Drain (Dreno), o qual conduz quando  $V_{ds} < 0$ . A operação como transistor ocorre quando  $V_{ds} > 0$ . A figura 4.11 mostra a estrutura básica do transistor.

Quando uma tensão  $V_{gs} > 0$  é aplicada, o potencial positivo no gate (porta) repele as lacunas na região P, deixando uma carga negativa, mas sem portadores livres. Quando esta tensão atinge um certo limiar ( $V_{th}$ ), elétrons livres (gerados por efeito térmico) presentes na região P, são atraídos e formam um canal N dentro da região P, pelo qual torna-se possível a passagem de corrente entre D e S. Elevando  $V_{gs}$ , mais portadores são atraídos, ampliando o canal, reduzindo sua resistência, permitindo o aumento de da corrente de dreno,  $I_d$ . Este comportamento caracteriza a chamada "região resistiva".

A passagem de  $I_d$  pelo canal produz uma queda de tensão que leva ao seu afinilamento, ou seja, o canal é mais largo na fronteira com a região N<sup>+</sup> do que quando se liga à região N<sup>-</sup>. Um aumento de  $I_d$  leva a uma maior queda de tensão no canal e a um maior afinilamento, o que conduziria ao seu colapso e à extinção da corrente! Obviamente o fenômeno tende a um ponto de equilíbrio, no qual a corrente  $I_d$  se mantém constante para qualquer  $V_{ds}$ , caracterizando a região ativa do MOSFET. A figura 4.12 mostra a característica estática do MOSFET, Uma pequena corrente de gate é necessária apenas para carregar e descarregar as capacitâncias de entrada do transistor. A resistência de entrada é da ordem de  $10^{12}$  ohms.

Estes transistores, em geral, são de canal N por apresentarem menores perdas e maior velocidade de comutação, devido à maior mobilidade dos elétrons em relação às lacunas.

A máxima tensão  $V_{ds}$  é determinada pela ruptura do diodo reverso. Os MOSFETs não apresentam segunda ruptura uma vez que a resistência do canal aumenta com o crescimento de  $I_d$ .

Este fato facilita a associação em paralelo destes componentes.

A tensão  $V_{gs}$  é limitada a algumas dezenas de Volts, por causa da capacidade de isolamento da camada de  $SiO_2$ .

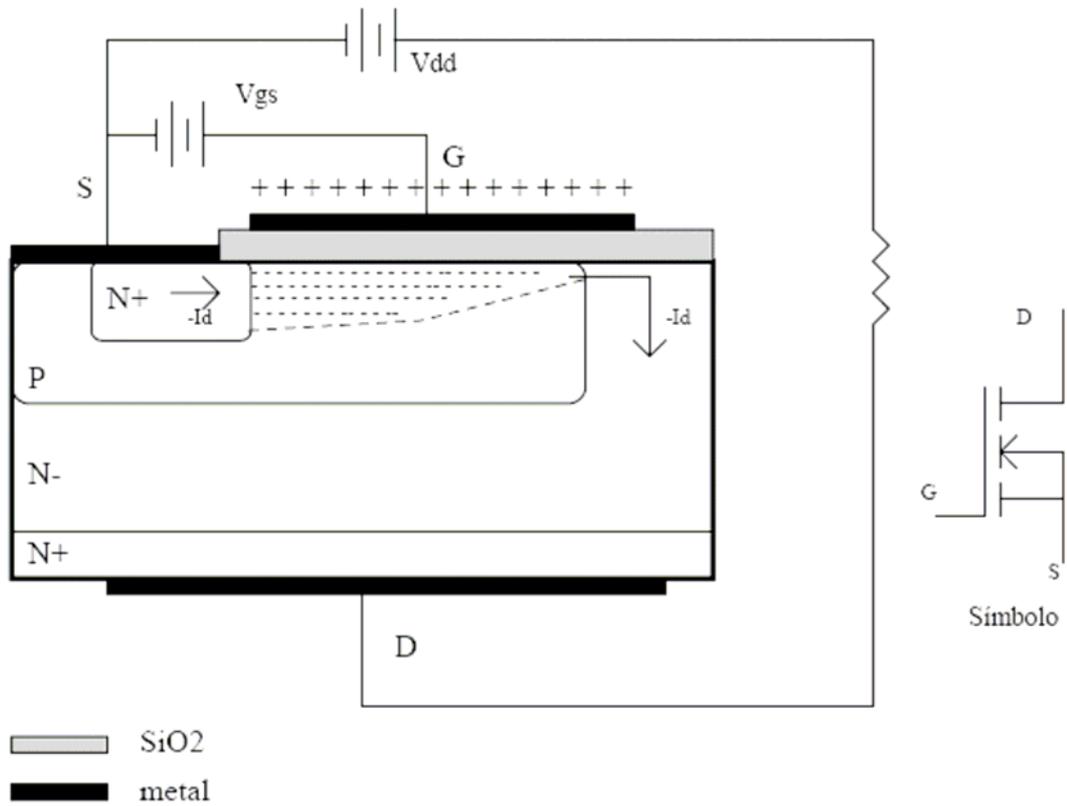


Figura 4.11. Estrutura básica de transistor MOSFET.

### 4.4.2 Área de Operação Segura

A figura 4.13 mostra a AOS dos MOSFET. Para tensões elevadas ela é mais ampla que para um TBP equivalente, uma vez que não existe o fenômeno de segunda ruptura. Para baixas tensões, entretanto, tem-se a limitação da resistência de condução.

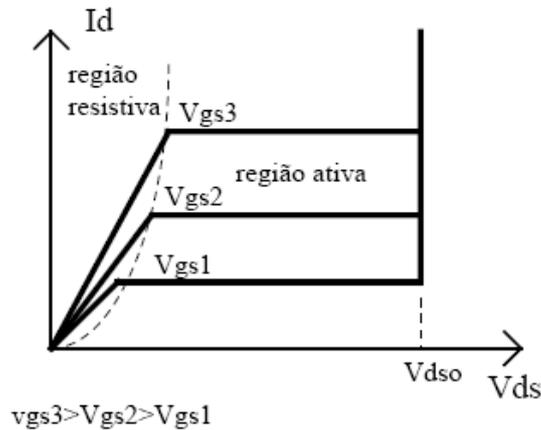


Figura 4.12. Característica estática do MOSFET.

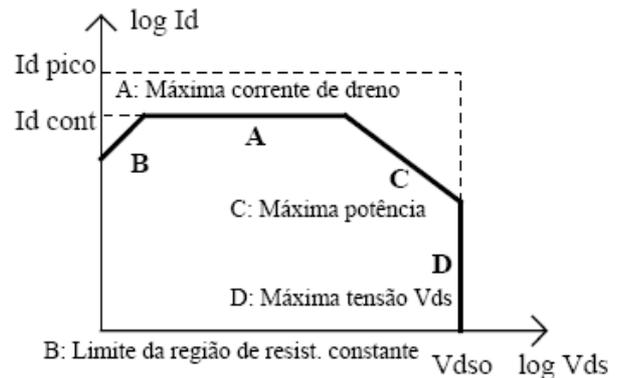


Figura 4.13. AOS para MOSFET.

### 4.4.3 Característica de chaveamento - carga indutiva

#### a) Entrada em condução (figura 4.14)

Ao ser aplicada a tensão de acionamento ( $V_{gg}$ ), a capacitância de entrada começa a se carregar, com a corrente limitada por  $R_g$ . Quando se atinge a tensão limiar de condução ( $V_{th}$ ), após  $t_d$ , começa a crescer a corrente de dreno. Enquanto  $I_d < I_o$ ,  $D_f$  se mantém em condução e  $V_{ds} = V_{dd}$ . Quando  $I_d = I_o$ ,  $D_f$  desliga e  $V_{ds}$  cai. Durante a redução de  $V_{ds}$  ocorre um aparente aumento da capacitância de entrada ( $C_{iss}$ ) do transistor, fazendo com que a variação de  $V_{gs}$  se torne muito mais lenta (em virtude do "aumento" da capacitância). Isto se mantém até que  $V_{ds}$  caia, quando, então, a tensão  $V_{gs}$  volta a aumentar, até atingir  $V_{gg}$ .

Na verdade, o que ocorre é que, enquanto  $V_{ds}$  se mantém elevado, a capacitância que drena corrente do circuito de acionamento é apenas  $C_{gs}$ . Quando  $V_{ds}$  diminui, a capacitância dentre dreno e source se descarrega, o mesmo ocorrendo com a capacitância entre gate e dreno. A descarga desta última

capacitância se dá desviando a corrente do circuito de acionamento, reduzindo a velocidade do processo de carga de  $C_{gs}$ , o que ocorre até que  $C_{gd}$  esteja descarregado.

**b) Desligamento**

O processo de desligamento é semelhante ao apresentado, mas na ordem inversa. O uso de uma tensão  $V_{gg}$  negativa apressa o desligamento, pois acelera a descarga da capacitância de entrada.

Como os MOSFETs não apresentam portadores minoritários estocados, ou seja, num MOSFET canal N todos os portadores em trânsito são elétrons.

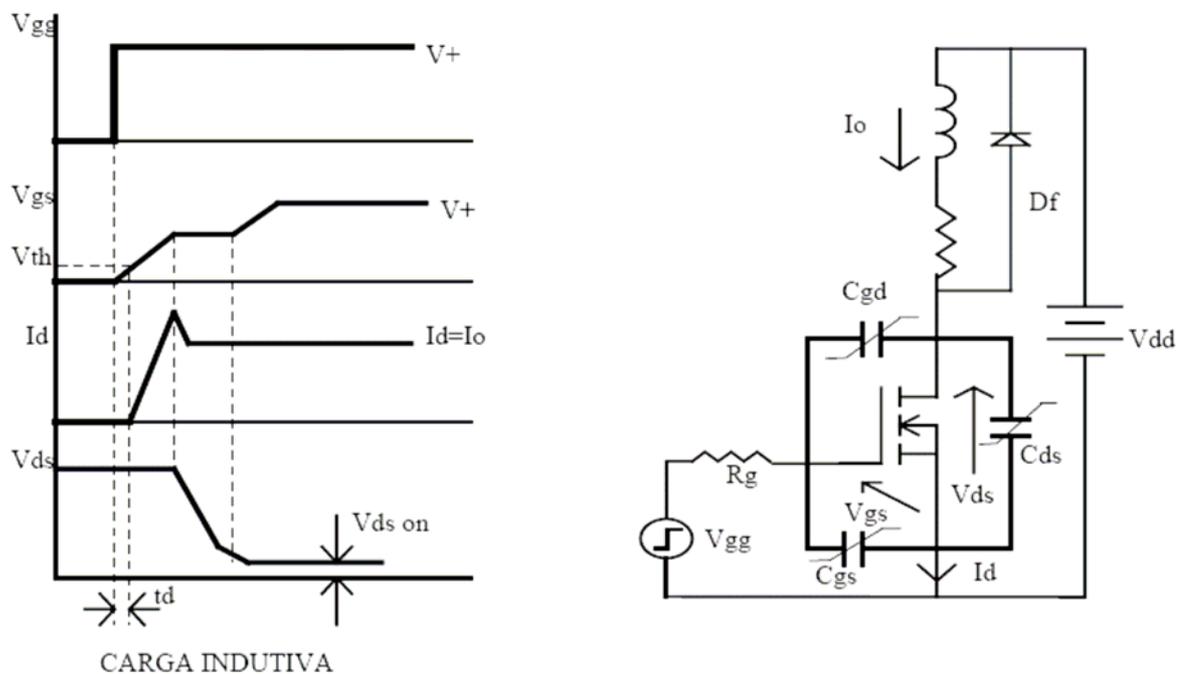


Figura 4.14 Formas de onda na entrada em condução de MOSFET com carga indutiva.

**4.4.4 Circuitos amaciadores (ou de ajuda à comutação) - "snubber"**

O papel dos circuitos amaciadores é garantir a operação do transistor dentro da AOS, especialmente durante o chaveamento de cargas indutivas. Embora ilustrado aqui para os MOSFETs, estes circuitos podem ser utilizados também para os demais tipos de transistores e também para os diodos.

**a) Desligamento** - Objetivo: atrasar o crescimento de  $V_{ds}$  e desviar  $I_d$  (figura 4.15) Quando  $V_{ce}$  começa a crescer, o capacitor  $C_s$  começa a se carregar (via  $D_s$ ), desviando parcialmente a corrente, reduzindo  $I_d$ .  $D_f$  só conduzirá quando  $V_{ds} > V_{cc}$ .

Quando o transistor ligar o capacitor se descarregará por ele, com a corrente limitada por  $R_s$ . A energia acumulada em  $C_s$  será, então, dissipada sobre  $R_s$ .

Sejam as formas de onda mostradas na figura 4.16. Considerando que  $I_d$  caia linearmente e que  $I_o$  é constante, a corrente por  $C_s$  cresce linearmente. Fazendo-se com que  $C_s$  complete sua carga quando  $I_d=0$ , o pico de potência se reduzirá a menos de 1/4 do seu valor sem circuito amaciador.

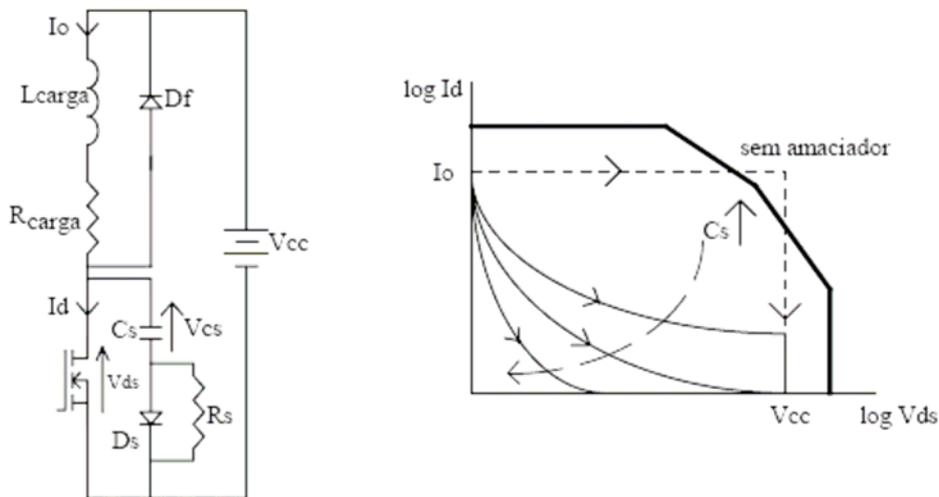


Figura 4.15. Circuito amaciador de desligamento e trajetórias na AOS

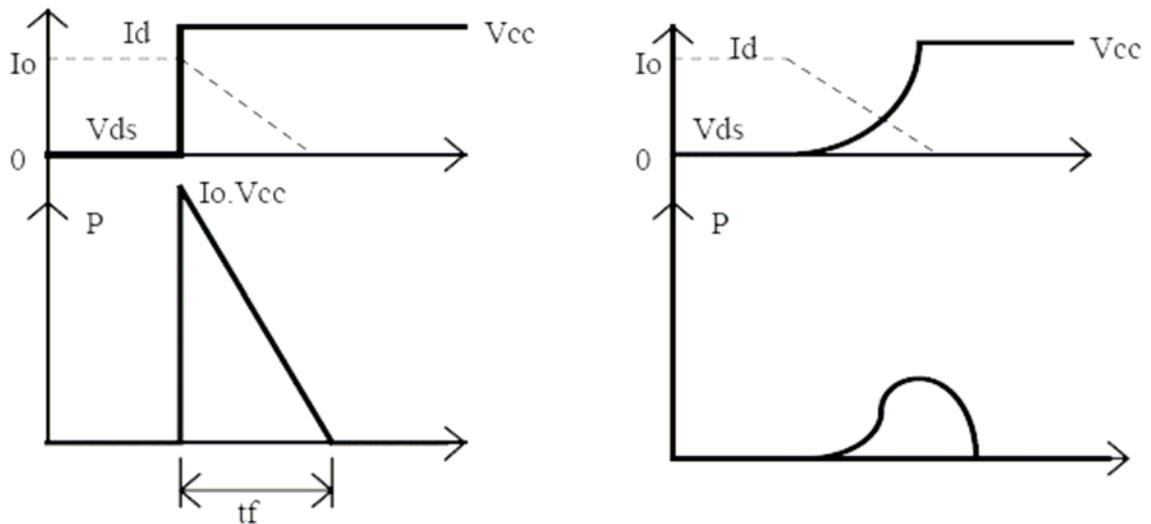


Figura 4.16. Formas de onda no desligamento sem e com o circuito amaciador.

O valor de  $R_s$  deve ser tal que permita toda a descarga de  $C_s$  durante o mínimo tempo ligado do Transistor e, por outro lado, limite o pico de corrente em um valor inferior à máxima corrente de pico repetitiva do componente. Deve-se usar o maior  $R_s$  possível.

**b) Entrada em condução:** Objetivo: reduzir  $V_{ds}$  e atrasar o aumento de  $I_d$  (figura 4.17)

No circuito sem amaciador, após o disparo do transistor,  $I_d$  cresce, mas  $V_{ds}$  só se reduz quando  $D_f$  deixar de conduzir. A colocação de  $L_s$  provoca uma redução de  $V_{ds}$ , além de reduzir a taxa de crescimento de  $I_d$ .

Normalmente não se utiliza este tipo de circuito, considerando que  $L_s$  possui baixo valor e pode ser substituído pela própria indutância parasita do circuito.

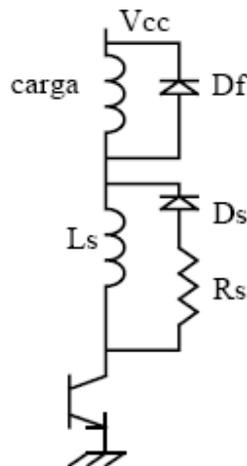


Figura 4.17. Circuito amaciador para entrada em condução.

#### 4.5 IGBT (Insulated Gate Bipolar Transistor)

O IGBT alia a facilidade de acionamento dos MOSFET com as pequenas perdas em condução dos TBP. Sua velocidade de chaveamento é semelhante à dos transistores bipolares.

##### 4.5.1 Princípio de funcionamento

A estrutura do IGBT é similar à do MOSFET, mas com a inclusão de uma camada P+ que forma o coletor do IGBT, como se vê na figura 4.18.

Em termos simplificados pode-se analisar o IGBT como um MOSFET no qual a região Ntem sua condutividade modulada pela injeção de portadores minoritários (lacunas), a partir da região P+, uma vez que J1 está diretamente polarizada. Esta maior condutividade produz uma menor queda de tensão em comparação a um MOSFET similar.

O controle de componente é análogo ao do MOSFET, ou seja, pela aplicação de uma polarização entre gate e emissor. Também para o IGBT o acionamento é feito por tensão.

A máxima tensão suportável é determinada pela junção J2 (polarização direta) e por J1 (polarização reversa). Como J1 divide 2 regiões muito dopadas, conclui-se que um IGBT não suporta tensões elevadas quando polarizado reversamente.

Os IGBTs apresentam um tiristor parasita. A construção do dispositivo deve ser tal que evite o acionamento deste tiristor, especialmente devido às capacitâncias associadas à região P, a qual relaciona-se à região do gate do tiristor parasita. Os modernos componentes não apresentam problemas relativos a este elemento indesejado.

#### **4.5.2 Características de chaveamento**

A entrada em condução é similar ao MOSFET, sendo um pouco mais lenta a queda da tensão  $V_{ce}$ , uma vez que isto depende da chegada dos portadores vindos da região P+.

Para o desligamento, no entanto, tais portadores devem ser retirados. Nos TBPs isto se dá pela drenagem dos portadores via base, o que não é possível nos IGBTs, devido ao acionamento isolado. A solução encontrada foi a inclusão de uma camada N+, na qual a taxa de recombinação é bastante mais elevada do que na região N-. Desta forma, as lacunas presentes em N+ recombinam-se com muita rapidez, fazendo com que, por difusão, as lacunas existentes na região N- refluem, apressando a extinção da carga acumulada na região N-, possibilitando o restabelecimento da barreira de potencial e o bloqueio do componente.

#### **4.5.3 IGBT rápido x IGBT de baixas perdas**

Existem atualmente no mercado 2 tipos de IGBTs: os rápidos e os de baixas perdas. Este fato já indica que não existem IGBTs rápidos e com baixas perdas. A diferença está relacionada exatamente ao comportamento no desligamento descrito anteriormente. Os IGBTs de baixas perdas utilizam dopagem e materiais que reduzem as perdas em condução, no entanto, implicam num maior tempo de desligamento. Para os dispositivos rápidos, não se consegue reduzir significativamente a queda de tensão sobre o componente quando em condução.

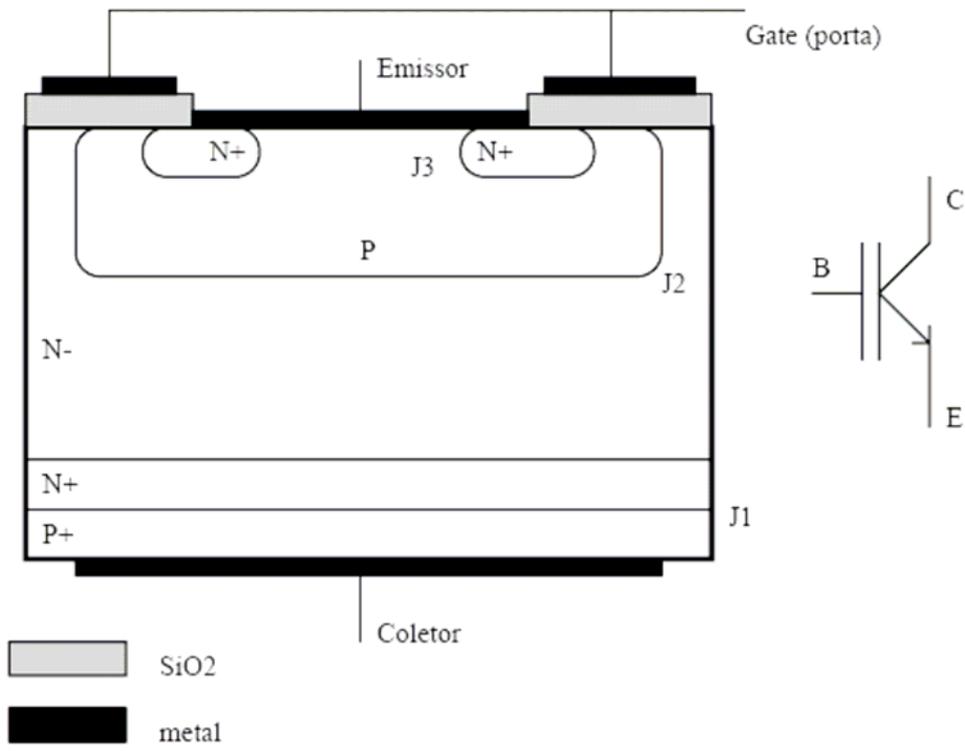


Figura 4.18. Estrutura básica de IGBT.

## 4.6 - Alguns Critérios de Seleção

Um primeiro critério é o dos limites de tensão e de corrente. Os MOSFET possuem uma faixa mais reduzida de valores, ficando, tipicamente entre: 100V/200A e 1000V/20A.

Já os TBP e IGBT atingem potências mais elevadas, indo até 1200V/500A.

Como o acionamento do IGBT é muito mais fácil do que o do TBP, seu uso tem sido crescente, em detrimento dos TBP.

Outro importante critério para a seleção refere-se às perdas de potência no componente. Assim, em aplicações em alta frequência (acima de 50kHz) devem ser utilizados MOSFETs. Em frequências mais baixas, qualquer dos 3 componentes pode responder satisfatoriamente. No entanto, as perdas em condução dos TBPs e dos IGBTs são sensivelmente menores que as dos MOSFET.

Como regra básica: em alta frequência: MOSFET em média/baixa frequência: IGBT

## 4.7 Parte Experimental

O circuito de teste está mostrado na figura 4.19

### 4.7.1 Material Necessário

- Osciloscópio digital (2 canais)
- Impressora acoplável ao osciloscópio
- Circuito de teste
- Fonte de alimentação +/- 15V; 0,5A, ajustável (isolada da rede).

### 4.7.2 Oscilador de onda quadrada

Para todas as medições nesta experiência, conecte ambos os terminais de terra das pontas de prova na placa de testes, a fim de minimizar o efeito das capacitâncias das pontas de prova.

- a) Coloque todas as chaves na posição D. Alimente o circuito com uma fonte de +/- 15V
- b) Observe os pontos 1 e 2 (em relação ao 0). Descreva o funcionamento do oscilador. Varie o TRIMPOT e identifique seu efeito sobre as formas de onda
- c) Para uma variação simétrica (em relação ao zero) da tensão no ponto 2, meça os tempos de subida e de descida, bem como os atrasos nos cruzamentos com zero.

#### 4.7.3 Testes dos diodos

- a) Coloque a chave CH3 na posição "DIODOS". Coloque o POTENCIÔMETRO na posição de máxima resistência. Ligue a chave CH9 (selecione o diodo retificador 1N5404 para o teste)
- b) Observando os pontos 2 e 5 em relação ao ponto 0, observe e meça, caso existam, os atrasos nas comutações do diodo. Faça a medida considerando o atraso entre os cruzamentos com o zero de ambos os sinais.
- c) Colocando o "terra" das pontas de prova no ponto 5, observe o ponto 4 (corrente pelo diodo) e 0 (inverso da tensão sobre o diodo). Observando as comutações, comente e justifique as formas de onda. Procure associar os fenômenos observados nos itens "b" e "c" para explicar os atrasos
- d) Varie o POTENCIÔMETRO e veja o efeito sobre as formas de onda.
- e) Estime a carga de recombinação reversa do diodo ( $Q_{rr}$ ) com o potenciômetro na posição de mínima resistência.
- f) Abra a chave CH9 e feche CH8. Repita os itens "b", "c", "e" para o diodo rápido (SK3GL04)
- g) Abra a chave CH8 e feche CH7. Repita os itens "b", "c", "e" para o diodo schottky (1N5818)
- h) No relatório, comente as diferenças observadas entre os diferentes tipos de diodos EE832 – Lab. de Eletrônica Industrial 2002 FEEC – UNICAMP 13
- i) Para o diodo rápido (CH8), ligue a chave CH2 (a qual conecta uma indutância em série com o diodo). Observe e comente o efeito sobre as formas de onda da corrente e da tensão no diodo.

#### 4.7.4 Testes dos transistores

- a) Desligue todas as chaves. Passe a chave CH3 para a posição "TRANSISTORES". Ligue CH4, que conecta o TBP TIP54 ao circuito.

- b) Com o POTENCIÔMETRO na posição de resistência máxima, observe os pontos 1 (tensão de comando) e 5 (tensão  $V_{ce}$  ou  $V_{ds}$ ). Considere que o 2o AmpOp faz parte do circuito de acionamento e os eventuais atrasos por ele introduzidos (medidos anteriormente) ocorrem para todos os transistores analisados.
- c) Observe, anote e comente os atrasos para ligar e desligar o transistor. Meça os atrasos medindo o intervalo entre o início da variação da tensão de comando e o início da variação na tensão do transistor.
- d) Varie o POTENCIÔMETRO e verifique o efeito sobre os atrasos observados no item anterior. Justifique.
- e) Com o POTENCIÔMETRO novamente na máxima resistência, ligue a chave CH1 (que conecta um capacitor em paralelo com o POTENCIÔMETRO). Verifique e comente seu efeito sobre os atrasos.
- f) Novamente com o POTENCIÔMETRO na máxima resistência, mas sem o capacitor (CH1 desligado), observe agora os pontos 2 e 5, varie o TRIMPOT e verifique o efeito da polarização negativa de base (gate) sobre os atrasos para ligar e desligar o transistor. Explique.
- Retorne para a situação inicial de uma onda simétrica.
- g) Desligue CH4 e ligue CH5 (conectando o MOSFET). Repita os itens, "c", "d", "e" e "f".
- h) Desligue CH5 e ligue CH6 (conectando o IGBT). Repita os itens "c", "d", "e" e "f".
- i) Para o MOSFET e o IGBT, observe os pontos 3 ( $V_{gs}$ ) e 5 ( $V_{ds}$ ), em ambas as transições. Varie o POTENCIÔMETRO e comente seu efeito. Analise comparativamente as capacitâncias de entrada dos transistores.
- j) Para o IGBT, observe os pontos 4 e 0 (em relação ao 5), o que mostra a corrente de coletor e a tensão  $V_{ce}$  (invertida). Anote e comente as formas de onda com carga resistiva (chave CH2 desligada) e com carga indutiva (chave CH2 ligada). Verifique em detalhe a entrada em condução do transistor.

Para a próxima aula, apresente o relatório contendo os resultados experimentais comentados.

Apresente os resultados do exercício preparatório para a 5ª experiência.

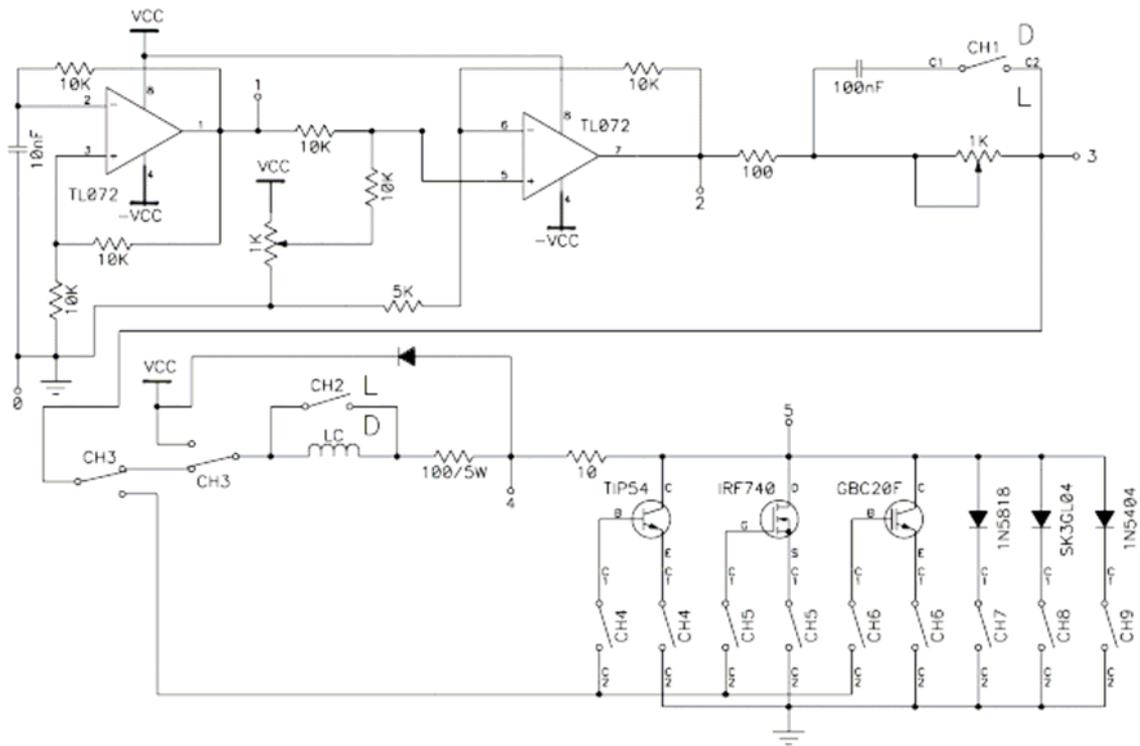


Figura 4.19 Circuito de teste.

---

## APÊNDICE D

### O SOFTWARE DO MICROCONTROLADOR (PIC)

```
// implementação de funções para FTDI, SPI e CD4094 que serão usados
// na placa de controle 04/12/2003
```

```
unsigned char ler_ftdi(void)
{
    TRISD = 0b11111111;
    TRISB = 0b0100001;

    while (RB0==1);

    RB1= 1;
    DelayMs(3);
    RB1=0;

    return (PORTD);

    RB1= 1;
}

void escrever_ftdi(unsigned char valor)
{
    TRISD = 0b00000000;
    TRISB = 0b0100001;
    do
    {
        RA1= 1;
        DelayMs(10);
        RB7= 0; // conectado em WR do FTDI
        DelayMs(3);
        RB7=1;
        PORTD = valor; // dado a ser escrito no FTDI
        DelayMs(3);
        RB7=0;
        RA1=0;
    } while(RB6==0);
```

```
void escreve_4094(int qual_4094,int d7,int d6,int d5,int d4,int d3,int d2,int d1,int d0)
{

int espera;

TRISA0 =0;
TRISA1 =0;
TRISA2 =0;
TRISA3 =0;
TRISA5 =0;
TRISC0 =0;
TRISC1 =0;
TRISC2 =0;
TRISB2 =0;
TRISB3 =0;
TRISB4 =0;

espera =5;

switch (qual_4094)
{
case 0:
{
RA0=0; // descida do Strobe no pino RA0
break;
}
case 1:
{
RA1=0; // descida do Strobe no pino RA1
break;
}
case 2:
{
RA2=0; // descida do Strobe no pino RA2
break;
}
case 3:
{
RA3=0; // descida do Strobe no pino RA3
break;
}
case 4:
{
```

```
    RA5=0; // descida do Strobe no pino RA5
    break;
}
case 5:
{
    RC0=0; // descida do Strobe no pino RC0
    break;
}
case 6:
{
    RC1=0; // descida do Strobe no pino RC1
    break;
}
case 7:
{
    RC2=0; // descida do Strobe no pino RC2
    break;
}
case 8:
{
    RB2=0; //descida do Strobe no pino RC2
    break;
}
}
DelayMs(espera);
```

```
    RB4=1; //subida do Clock
    DelayMs(espera);
    RB3=d7; //bit d0
    DelayMs(espera);
    RB4=0; //descida do Clock
    DelayMs(espera);
```

```
    RB4=1; //subida do Clock
    DelayMs(espera);
    RB3=d6; //bit d1
    DelayMs(espera);
    RB4=0; //descida do Clock
    DelayMs(espera);
```

```
    RB4=1; //subida do Clock
    DelayMs(espera);
    RB3=d5; //bit d2
    DelayMs(espera);
    RB4=0; //descida do Clock
    DelayMs(espera);
```

```

RB4=1; //subida do Clock
DelayMs(espera);
RB3=d4; //bit d3
DelayMs(espera);
RB4=0; //descida do Clock
DelayMs(espera);

```

```

RB4=1; //subida do Clock
DelayMs(espera);
RB3=d3; //bit d4
DelayMs(espera);
RB4=0; //descida do Clock
DelayMs(espera);

```

```

RB4=1; //subida do Clock
DelayMs(espera);
RB3=d2; //bit d5
DelayMs(espera);
RB4=0; //descida do Clock
DelayMs(espera);

```

```

RB4=1; //subida do Clock
DelayMs(espera);
RB3=d1; //bit d6
DelayMs(espera);
RB4=0; //descida do Clock
DelayMs(espera);

```

```

RB4=1; //subida do Clock
DelayMs(espera);
RB3=d0; //bit d7
DelayMs(espera);
RB4=0; //descida do Clock
DelayMs(espera);

```

```

RB4=1; //subida do Clock *****
DelayMs(espera);

```

```

switch (qual_4094) // strobe do 4094
{
case 0:
{
RA0=1; // subida do Strobe no pino RA0
DelayMs(espera); //*****
RA0 =0; //*****

```

```
        break;
    }
case 1:
    {
        RA1=1; // subida do Strobe no pino RA1
        DelayMs(espera); //*****
        RA1 =0; //*****
        break;
    }
case 2:
    {
        RA2=1; // subida do Strobe no pino RA2
        DelayMs(espera); //*****
        RA2 =0; //*****
        break;
    }
case 3:
    {
        RA3=1; // subida do Strobe no pino RA3
        DelayMs(espera); //*****
        RA3 =0; //*****
        break;
    }
case 4:
    {
        RA5=1; // subida do Strobe no pino RA5
        DelayMs(espera); //*****
        RA5 =0; //*****
        break;
    }
case 5:
    {
        RC0=1; // subida do Strobe no pino RC0
        DelayMs(espera); //*****
        RC0 =0; //*****
        break;
    }
case 6:
    {
        RC1=1; // subida do Strobe no pino RC1
        DelayMs(espera); //*****
        RC1 =0; //*****
        break;
    }
case 7:
    {
        RC2=1; // subida do Strobe no pino RC2
```

```

    DelayMs(espera); //*****
    RC2 =0; //*****
    break;
}
case 8:
{
    RB2=1; //subida do Strobe seleção da exp. no pino RB2
    DelayMs(espera); //*****
    RB2 =0; //*****
    break;
}
}
DelayMs(espera);
RB4=1; //subida do Clock
DelayMs(espera);
}

```

```
void escrever_pot_spi(void)
```

```
{
int qual_pot;
int dado;
```

```

qual_pot= ler_ftdi();
putst ("Potenciometro: ");
putdec(qual_pot);
putch(10);
putch(13);

```

```

dado=ler_ftdi();
putst ("Dado para o Pot.: ");
putdec(dado);
putch(10);
putch(13);

```

```
write_spi(dado);
```

```
switch (qual_pot)
```

```

{
case 1:
{
    escreve_4094(0,0,0,0,0,0,0,1);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0);
    break;
}
case 2:

```

```
{
  escreve_4094(0,0,0,0,0,0,0,1,0);
  DelayMs(50);
  escreve_4094(0,0,0,0,0,0,0,0,0);
  break;
}
case 3:
  {
    escreve_4094(0,0,0,0,0,0,1,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
case 4:
  {
    escreve_4094(0,0,0,0,0,1,0,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
case 5:
  {
    escreve_4094(0,0,0,0,1,0,0,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
case 6:
  {
    escreve_4094(0,0,0,1,0,0,0,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
case 7:
  {
    escreve_4094(0,0,1,0,0,0,0,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
case 8:
  {
    escreve_4094(0,1,0,0,0,0,0,0,0);
    DelayMs(50);
    escreve_4094(0,0,0,0,0,0,0,0,0);
    break;
  }
```

```
    }  
  }  
}
```

```
escrever_rele()
```

```
{  
int dado;  
int n_rele;  
int comando;
```

```
dado= ler_ftdi();
```

```
n_rele = 0b00011111 & dado;  
putst ("Rele: ");  
putdec(n_rele);  
putch(10);  
putch(13);
```

```
comando =0b00100000 & dado;  
if (comando == 32)  
{  
  comando = 1;  
}
```

```
putst ("Atuacao: ");  
putdec(comando);  
putch(10);  
putch(13);
```

```
if ((n_rele >=1) && (n_rele <= 8)) // reles de 1 a 8  
{  
  putst ("Rele de 1 a 8");  
  putch(10);  
  putch(13);  
  switch (n_rele)  
  {  
    case 1:  
    {  
      r1=comando;  
      break;  
    }  
    case 2:  
    {  
      r2= comando;  
      break;  
    }  
  }
```

```
    case 3:
    {
        r3= comando;
        break;
    }
    case 4:
    {
        r4= comando;
        break;
    }
    case 5:
    {
        r5= comando;
        break;
    }
    case 6:
    {
        r6= comando;
        break;
    }
    case 7:
    {
        r7= comando;
        break;
    }
    case 8:
    {
        r8= comando;
        break;
    }
}

escreve_4094(4,r8,r7,r6,r5,r4,r3,r2,r1);
}
```

```
if ((n_rele >=9) && (n_rele <= 16))
{
    putst ("Rele de 9 a 16");
    putch(10);
    putch(13);
    switch (n_rele)
    {
        case 9:
        {
            r9= comando;
            break;
        }
    }
}
```

```
case 10:
    {
        r10= comando;
        break;
    }
case 11:
    {
        r11= comando;
        break;
    }
case 12:
    {
        r12= comando;
        break;
    }
case 13:
    {
        r13= comando;
        break;
    }
case 14:
    {
        r14= comando;
        break;
    }
case 15:
    {
        r15= comando;
        break;
    }
case 16:
    {
        r16= comando;
        break;
    }
}
// escrita no 4094 n. 5 que controla relês de 9 a 16
escreve_4094(5,r16,r15,r14,r13,r12,r11,r10,r9);
}

if ((n_rele >=17) && (n_rele <= 23)) // reles de 17 a 23
{
    putst ("Rele de 17 a 23");
    putch(10);
    putch(13);
    switch (n_rele)
    {
```

```
case 17:
    {
        r17= comando;
        break;
    }
case 18:
    {
        r18= comando;
        break;
    }
case 19:
    {
        r19= comando;
        break;
    }
case 20:
    {
        r20= comando;
        break;
    }
case 21:
    {
        r21= comando;
        break;
    }
case 22:
    {
        r22= comando;
        break;
    }
case 23:
    {
        r23= comando;
        break;
    }
}
// escrita no 4094 n. 6 que controla relês de 17 a 23 e ponto
// de medição m1
escreve_4094(6,m1,r23,r22,r21,r20,r19,r18,r17);
}
```

```
void escreve_banco_res()
{
    ler_ftdi();
}
```

```
if (RD7==0 && RD6==0)
{
    putst ("Banco 1 ");
    putch(10);
    putch(13);

    b1d0 = RD0;
    b1d1 = RD1;
    b1d2 = RD2;
    b1ch0 =RD3;
    b1ch1 =RD4;
    b1ch2 =RD5;
    escreve_4094(1,b2d1,b2d0,b1ch2,b1ch1,b1ch0,b1d2,b1d1,b1d0);
}

if (RD7==0 && RD6==1)
{
    putst ("Banco 2 ");
    putch(10);
    putch(13);

    b2d0 = RD0;
    b2d1 = RD1;
    b2d2 = RD2;
    b2ch0 =RD3;
    b2ch1 =RD4;
    b2ch2 =RD5;
    escreve_4094(1,b2d1,b2d0,b1ch2,b1ch1,b1ch0,b1d2,b1d1,b1d0);
    escreve_4094(2,b3ch0,b3d2,b3d1,b3d0,b2ch2,b2ch1,b2ch0,b2d2);
}

if (RD7==1 && RD6==0)
{
    putst ("Banco 3 ");
    putch(10);
    putch(13);

    b3d0 = RD0;
    b3d1 = RD1;
    b3d2 = RD2;
    b3ch0 =RD3;
    b3ch1 =RD4;
    b3ch2 =RD5;
    escreve_4094(2,b3ch0,b3d2,b3d1,b3d0,b2ch2,b2ch1,b2ch0,b2d2);
    escreve_4094(3,b4ch2,b4ch1,b4ch0,b4d2,b4d1,b4d0,b3ch2,b3ch1);
}
```

```
if (RD7==1 && RD6==1) //se banco 4
{
    putst ("Banco 4 ");
    putch(10);
    putch(13);
    b4d0 = RD0;
    b4d1 = RD1;
    b4d2 = RD2;
    b4ch0 =RD3;
    b4ch1 =RD4;
    b4ch2 =RD5;
    escreve_4094(2,b3ch0,b3d2,b3d1,b3d0,b2ch2,b2ch1,b2ch0,b2d2);
    escreve_4094(3,b4ch2,b4ch1,b4ch0,b4d2,b4d1,b4d0,b3ch2,b3ch1);
}
}
```

```
escreve_pontos_med()
{
    ler_ftdi();
    m1 =RD0;
    m2 =RD1;
    m3 =RD2;
    m4 =RD3;
    m5 =RD4;
    m6 =RD5;
    m7 =RD6;
    m8 =RD7;
    putst ("Byte 1: ");
    putdec(PORTD);
    putch(10);
    putch(13);

    ler_ftdi();
    m9 =RD0;
    putst ("Byte 2: ");
    putdec(PORTD);
    putch(10);
    putch(13);

    escreve_4094(6,m1,r23,r22,r21,r20,r19,r18,r17);
    escreve_4094(7,m9,m8,m7,m6,m5,m4,m3,m2);
}
```

```
void selecionar_exp(void)
```

```
{
int qual_exp;

qual_exp = ler_ftdi();
putst ("experiencia: ");
putdec(qual_exp);
putch(10);
putch(13);

switch (qual_exp)
{
case 1:
{
escreve_4094(8,0,0,0,0,0,0,1);
break;
}
case 2:
{
escreve_4094(8,0,0,0,0,0,1,0);
break;
}
case 3:
{
escreve_4094(8,0,0,0,0,1,0,0);
break;
}
case 4:
{
escreve_4094(8,0,0,0,1,0,0,0);
break;
}
case 5:
{
escreve_4094(8,0,0,1,0,0,0,0);
break;
}
case 6:
{
escreve_4094(8,0,1,0,0,0,0,0);
break;
}
case 7:
{
escreve_4094(8,0,1,0,0,0,0,0);
break;
}
case 8:
```

```
        {
        escreve_4094(8,1,0,0,0,0,0,0);
        break;
        }
    }
}

//=====
```

```
#include <pic.h>
#include "delay.h"
#include "mserial.h"
```

```
// variáveis para o relé
```

```
bit r1;
bit r2;
bit r3;
bit r4;
bit r5;
bit r6;
bit r7;
bit r8;
bit r9;
bit r10;
bit r11;
bit r12;
bit r13;
bit r14;
bit r15;
bit r16;
bit r17;
bit r18;
bit r19;
bit r20;
bit r21;
bit r22;
bit r23;
```

```
//variáveis para banco de resistores 1
```

```
bit b1d0;
bit b1d1;
bit b1d2;
bit b1ch0;
bit b1ch1;
bit b1ch2;
```

```
//variáveis para banco de resistores 2
bit b2d0;
bit b2d1;
bit b2d2;
bit b2ch0;
bit b2ch1;
bit b2ch2;

//variáveis para banco de resistores 3
bit b3d0;
bit b3d1;
bit b3d2;
bit b3ch0;
bit b3ch1;
bit b3ch2;

//variáveis para banco de resistores 4
bit b4d0;
bit b4d1;
bit b4d2;
bit b4ch0;
bit b4ch1;
bit b4ch2;

//variáveis para os pontos de medição
bit m1;
bit m2;
bit m3;
bit m4;
bit m5;
bit m6;
bit m7;
bit m8;
bit m9;

#include "mplaca.h"

void main()
{
  unsigned int cabec1;
  unsigned int cabec2;
  unsigned int comando;

  // inicia comunicacao rs232
  init_uart();

  putst("Placa de Controle - Laboratorio Remoto");
  putch(10);
  putch(13);
```

```
putst ("Inicio inicializacao da Placa");
putch(10);
putch(13);

//inicia limpando os 4094
escreve_4094(8,0,0,0,0,1,0,0,0); //seleciona exp 4
escreve_4094(0,0,0,0,0,0,0,0,0);
escreve_4094(1,0,0,0,0,0,0,0,0);
escreve_4094(2,0,0,0,0,0,0,0,0);
escreve_4094(3,0,0,0,0,0,0,0,0);
escreve_4094(4,0,0,0,0,0,0,0,0);
escreve_4094(5,0,0,0,0,0,0,0,0);
escreve_4094(6,0,0,0,0,0,0,0,0); // colocar inializacao chaves
escreve_4094(7,0,0,0,0,0,0,0,0);

putst ("Fim da inicializacao da Placa");
putch(10);
putch(13);

//inicia comunicação SPI
init_spi();

while (1)
{
cabec1 = ler_ftdi(); // cabecalho
cabec2 = ler_ftdi(); // cabecalho
comando = ler_ftdi(); // comando

if ((cabec1 == 's') && (cabec2 == 'm'))
{
switch (comando)
{
case 's': //comando para potenciometro SPI
{
putst ("Comando para Pot SPI");
putch(10);
putch(13);
escrever_pot_spi();
break;
}
case 'b': //comando para banco de resistores
{
putst ("Comando para Banco de Resistores");
putch(10);
putch(13);
escreve_banco_res();
break;
}
case 'r': //comando para reles
```

```
    {
    putst ("Comando para Reles");
    putch(10);
    putch(13);
    escrever_rele();
    break;
    }
case 'm': //comando para pontos de medição
    {
    putst ("Comando para Ponto de Medicao");
    putch(10);
    putch(13);
    escreve_pontos_med();
    break;
    }
case 'e': //comando para selecao da experiencia
    {
    putst ("Comando para selecao da experiencia");
    putch(10);
    putch(13);
    selecionar_exp();
    break;
    }
    }
}
}
```

---

## APÊNDICE E

### O SOFTWARE DO HOST (PC)

```
Private Declare Function FT_Open Lib "FTD2XX.DLL" (ByVal intDeviceNumber As Integer, ByVal lnghandle As Long) As Long
Private Declare Function FT_Close Lib "FTD2XX.DLL" (ByVal lnghandle As Long) As Long
Private Declare Function FT_Read Lib "FTD2XX.DLL" (ByVal lnghandle As Long, ByVal lpszBuffer As String, ByVal lngBufferSize As Long, ByRef lngBytesReturned As Long) As Long
Private Declare Function FT_Write Lib "FTD2XX.DLL" (ByVal lnghandle As Long, ByVal lpszBuffer As String, ByVal lngBufferSize As Long, ByRef lngBytesWritten As Long) As Long
Private Declare Function FT_SetTimeouts Lib "FTD2XX.DLL" (ByVal lnghandle As Long, ByVal lngReadTimeout As Long, ByVal lngWriteTimeout As Long) As Long
Private Declare Function FT_Purge Lib "FTD2XX.DLL" (ByVal lnghandle As Long, ByVal lngMask As Long) As Long
Private Declare Function FT_GetQueueStatus Lib "FTD2XX.DLL" (ByVal lnghandle As Long, ByRef lngRxBytes As Long) As Long
Private Declare Function FT_ResetDevice Lib "FTD2XX.DLL" (ByVal lnghandle As Long) As Long
```

```
Const FT_OK = 0
Const FT_INVALID_HANDLE = 1
Const FT_DEVICE_NOT_FOUND = 2
Const FT_DEVICE_NOT_OPENED = 3
Const FT_IO_ERROR = 4
Const FT_INSUFFICIENT_RESOURCES = 5
```

```
Dim lnghandle As Long
Dim lngBytesWritten As Long
Dim Connected As Integer
Dim strReadBuffer As String * 256
Dim lngBytesRead As Long
Dim lngTotalBytestRead As Long
Dim lngBytesToRead As Long
```

```
Dim chave1 As Integer
Dim chave2 As Integer
Dim chave3 As Integer
Dim chave4 As Integer
Dim chave5 As Integer
Dim chave6 As Integer
```

```
Dim chave7 As Integer
Dim chave8 As Integer
Dim chave9 As Integer
```

```
Dim ind_ch1(5) As Integer
Dim ind_ch2(5) As Integer
Dim ind_ch3(5) As Integer
```

```
Dim ponto_m As Integer
```

```
Dim v_ch_1 As Integer
Dim v_ch_2 As Integer
Dim v_ch_3 As Integer
Dim v_ext_ch2 As Integer
```

```
Dim v_pch1 As Integer
Dim v_pch2 As Integer
Dim v_pch3 As Integer
```

```
Dim v_bit(5) As Integer
```

```
Private Sub Combo1_Click()
```

```
If FT_Write(Inghandle, "smm" & Chr(0) & Chr(0), 5, lngBytesWritten) <> FT_OK Then
    'Label1.Caption = "Falha ao Escrever"
Else
    'Label1.Caption = "Sucesso ao Escrever no FTDI"
End If
```

```
For f = 0 To 5
Label4(f).BackColor = &HC0C0C0
Next
```

```
Select Case Combo1.ListIndex + 1
    Case 1 'se for canal 1
        'limpa o indice do canal 1
        For f = 0 To 5
            ind_ch1(f) = 0
            Label4(f).Caption = " " & f & " "
        Next
        ind_ch1(ponto_m) = 1
        ind_ch2(ponto_m) = 0
        ind_ch3(ponto_m) = 0
        Select Case ponto_m
            Case 0
```

```

v_ch_1 = 0
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 1
v_ch_1 = 1
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 2
v_ch_1 = 2
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 3
v_ch_1 = 3
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 4
v_ch_1 = 4
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 5
v_ch_1 = 5
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 6
v_ch_1 = 6
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 7
v_ch_1 = 7
vt = v_ch_1 + v_ch_2 + v_ch_3
End Select
'v_ch_1 = 0
Case 2 ' se for canal 2

```

```

For f = 0 To 5
ind_ch2(f) = 0
Label4(f).Caption = " " & f & " "
Next
ind_ch1(ponto_m) = 0
ind_ch2(ponto_m) = 1
ind_ch3(ponto_m) = 0
Select Case ponto_m
Case 0
v_ch_2 = 0
v_ext_ch2 = 0
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 1
v_ch_2 = 64
v_ext_ch2 = 0
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 2
v_ch_2 = 128
v_ext_ch2 = 0
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 3
v_ch_2 = 192
v_ext_ch2 = 0
vt = v_ch_1 + v_ch_2 + v_ch_3

```

```
Case 4
v_ch_2 = 0
v_ext_ch2 = 1
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 5
v_ch_2 = 64
v_ext_ch2 = 1
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 6
v_ch_2 = 128
v_ext_ch2 = 1
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 7
v_ch_2 = 192
v_ext_ch2 = 1
vt = v_ch_1 + v_ch_2 + v_ch_3
End Select
'v_ch_2 = 0
Case 3

For f = 0 To 5
ind_ch3(f) = 0
Label4(f).Caption = " " & f & " "
Next
ind_ch1(ponto_m) = 0
ind_ch2(ponto_m) = 0
ind_ch3(ponto_m) = 1
Select Case ponto_m
Case 0
v_ch_3 = 0 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 1
v_ch_3 = 1 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 2
v_ch_3 = 2 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 3
v_ch_3 = 3 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 4
v_ch_3 = 4 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 5
v_ch_3 = 5 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 6
v_ch_3 = 6 * 8
vt = v_ch_1 + v_ch_2 + v_ch_3
Case 7
v_ch_3 = 7 * 8
```

```
        vt = v_ch_1 + v_ch_2 + v_ch_3
    End Select

End Select

For f = 0 To 5
    If ind_ch1(f) = 1 Then
        Label4(f).BackColor = &H80FF80 'verde
        Label4(f).Caption = " Ch 1 "
        'Form2.Text1.Text = f
        'CWDataSocket2.Data.Value = f & "pma"
    End If

    If ind_ch2(f) = 1 Then
        Label4(f).BackColor = &HC0C0FF 'vermelho
        Label4(f).Caption = " Ch 2 "
        'Form2.Text2.Text = f
        'CWDataSocket2.Data.Value = f & "pmb"
    End If

    If ind_ch3(f) = 1 Then
        Label4(f).BackColor = &HFFFF80 'azul
        Label4(f).Caption = " Ref. "
        'Form2.Text3.Text = f
        'CWDataSocket2.Data.Value = f & "pmc"

    End If
Next

If FT_Write(Inghandle, "smm" & Chr(vt) & Chr(v_ext_ch2), 5, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

'Text1.Text = vt
'Text2.Text = v_ext_ch2

Combo1.Visible = False

End Sub

Private Sub Combo2_Click()
    p_med3 = Val(Combo2.Text)

    Select Case p_med3
        Case 1
            v_ch_1 = 3
```

```
Text1.Text = "Ponto 2"

Case 2
  v_ch_2 = 3
  Text2.Text = "Ponto 2"
Case 3
  v_ch_3 = 3
  Text3.Text = "ponto 2"
End Select
```

```
End Sub
```

```
Private Sub Combo3_Click()
  p_med4 = Val(Combo3.Text)
```

```
Select Case p_med4
  Case 1
    v_ch_1 = 4
    Text1.Text = "Ponto 3"

  Case 2
    v_ch_2 = 4
    Text2.Text = "Ponto 3"
  Case 3
    v_ch_3 = 4
    Text3.Text = "Ponto 3"
End Select
```

```
End Sub
```

```
Private Sub Combo4_Click()
  p_med5 = Val(Combo4.Text)
```

```
Select Case p_med5
  Case 1
    v_ch_1 = 5
    Text1.Text = "Ponto 5"

  Case 2
    v_ch_2 = 5
    Text2.Text = "Ponto 5"
  Case 3
    v_ch_3 = 5
    Text3.Text = "Ponto 5"
End Select
```

```
Combo4.Visible = False
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
Command1.Enabled = False
```

```
Dim erro As Integer
```

```
erro = 0
```

```
If FT_Open(0, lnghandle) <> FT_OK Then
```

```
Label7.ForeColor = &HC0&
```

```
Label7.Caption = "Falha ao conectar a Placa de Controle"
```

```
erro = 1
```

```
Else
```

```
Label7.ForeColor = &H8000&
```

```
Label7.Caption = "Sucesso ao Conectar a Placa de Controle"
```

```
End If
```

```
DoEvents
```

```
If FT_SetTimeouts(lnghandle, 20, 0) <> FT_OK Then
```

```
Label7.ForeColor = &HC0&
```

```
Label7.Caption = "Falha ao conectar a Placa de Controle"
```

```
End If
```

```
If FT_Write(lnghandle, "smm" & Chr(0) & Chr(0), 5, lngBytesWritten) <> FT_OK Then
```

```
Label1.Caption = "Falha ao Escrever"
```

```
Else
```

```
Label1.Caption = "Sucesso ao Escrever no FTDI"
```

```
End If
```

```
If FT_Write(lnghandle, "smr" & Chr(1), 4, lngBytesWritten) <> FT_OK Then
```

```
Label7.ForeColor = &HC0&
```

```
Label7.Caption = "Falha ao Escrever"
```

```
erro = 1
```

```
Else
```

```
Image10.Picture = chave_h_des.Picture
```

```
Label7.ForeColor = &H8000&
```

```
Label7.Caption = "chave 1 na posição inicial"
```

```
End If
```

```
DoEvents
```

```
If FT_Write(lnghandle, "smr" & Chr(10), 4, lngBytesWritten) <> FT_OK Then
```

```
Label7.ForeColor = &HC0&
```

```
Label7.Caption = "Falha ao Escrever"
```

```
erro = 1
```

```
Else
```

```
Image12.Picture = chave_h_des.Picture
```

```
Label7.ForeColor = &H8000&
```

```
Label7.Caption = "chave 2 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(3), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image11.Picture = ch3_p1.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 3 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(4), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image1.Picture = chave_v_des.Picture  
Image2.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 4 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(5), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image3.Picture = chave_v_des.Picture  
Image4.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 5 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(6), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image5.Picture = chave_v_des.Picture  
Image6.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&
```

```
Label7.Caption = "chave 6 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(7), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image7.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 7 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(8), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image8.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 8 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "smr" & Chr(9), 4, lngBytesWritten) <> FT_OK Then  
Label7.ForeColor = &HC0&  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Image9.Picture = chave_v_des.Picture  
Label7.ForeColor = &H8000&  
Label7.Caption = "chave 9 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "sms" & Chr(3) & Chr(64), 5, lngBytesWritten) <> FT_OK Then  
Label7.Caption = "Falha ao Escrever"  
erro = 1  
Else  
Label7.Caption = "potenciometro 3 na posição inicial"  
End If  
DoEvents
```

```
If FT_Write(Inghandle, "sms" & Chr(6) & Chr(64), 5, lngBytesWritten) <> FT_OK Then
```

```
Label7.Caption = "Falha ao Escrever"
erro = 1
Else
Label7.Caption = "potenciometro 6 na posição inicial"
End If
DoEvents

If FT_Write(Inghandle, "smm" & Chr(0) & Chr(0), 5, lngBytesWritten) <> FT_OK Then
Label7.Caption = "Falha ao Escrever"
Else
Label7.Caption = "Pontos de Medição inicializados"
End If

If erro = 0 Then
Label7.Caption = "Placa de Controle Iniciada com Sucesso"

chave1 = 0
chave2 = 0
chave3 = 1
chave4 = 0
chave5 = 0
chave6 = 0
chave7 = 0
chave8 = 0
chave9 = 0

v_ch_1 = 0
v_ch_2 = 0
v_ch_3 = 0
v_ext_ch2 = 0
ponto_m = 0

v_pch1 = 6
v_pch2 = 7
v_pch3 = 8

For f = 0 To 5
Label4(f).BackColor = &HC0C0C0
Label4(f).Caption = " " & f & " "
Next

Else
Label7.ForeColor = &HC0&
Label7.Caption = "Falha na inicialização da Placa de Controle"
End If
Command2.Enabled = True
End Sub
```

```
Private Sub Command2_Click()
If FT_Close(Inghandle) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao desconectar da Placa de Controle"
Else
    Label7.ForeColor = &H800000
    Label7.Caption = "Placa de Controle desconectada"
End If
Command1.Enabled = True
Command2.Enabled = False
End Sub

Private Sub Command3_Click()
If FT_Close(Inghandle) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao desconectar da Placa de Controle"
Else
    Label7.ForeColor = &H800000
    Label7.Caption = "Placa de Controle desconectada"
End If
DoEvents
End
End Sub

Private Sub CWDataSocket1_OnStatusUpdated(ByVal Status As Long, ByVal Error As Long, ByVal
Message As String)
Text1.Text = Message
End Sub

Private Sub Command4_Click()
If FT_Write(Inghandle, "smr" & Chr(9), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub CWSlide1_PointerValueCommitted(ByVal Pointer As Long, Value As Variant)

If FT_Write(Inghandle, "sms" & Chr(6) & Chr(Value), 5, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

End Sub
```

```
Private Sub CWSlide2_PointerValueCommitted(ByVal Pointer As Long, Value As Variant)
```

```
If FT_Write(Inghandle, "sms" & Chr(3) & Chr(Value), 5, lngBytesWritten) <> FT_OK Then  
    Label7.ForeColor = &HC0&  
    Label7.Caption = "Falha ao Escrever"  
Else  
    Label7.Caption = "Sucesso ao Escrever no FTDI"  
End If
```

```
End Sub
```

```
Private Sub enviar_Click()  
Dim a(2) As String  
CWDataSocket1.Data.Value = Text2.Text  
End Sub
```

```
Private Sub Form_Load()
```

```
chave1 = 0  
chave2 = 0  
chave3 = 1  
chave4 = 0  
chave5 = 0  
chave6 = 0  
chave7 = 0  
chave8 = 0  
chave9 = 0
```

```
v_ch_1 = 0  
v_ch_2 = 0  
v_ch_3 = 0  
v_ext_ch2 = 0  
ponto_m = 0  
v_pch1 = 6  
v_pch2 = 7  
v_pch3 = 8  
For f = 0 To 5  
Label4(f).Caption = " " & f & " "  
Next
```

```
Combo1.AddItem "Canal 1"  
Combo1.AddItem "Canal 2"  
Combo1.AddItem "Referência "
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
If FT_Close(Inghandle) <> FT_OK Then  
    Label7.ForeColor = &HC0&  
    Label7.Caption = "Falha ao desconectar da Placa de Controle"
```

```
Else
  Label7.ForeColor = &H800000
  Label7.Caption = "Placa de Controle desconectada"
End If
End Sub
Private Sub Image1_Click()
5 ' chave 4
If chave4 = 0 Then
  Image1.Picture = chave_v_lig.Picture
  Image2.Picture = chave_v_lig.Picture
  atua_relé = 4 + 32
  chave4 = 1
Else
  Image1.Picture = chave_v_des.Picture
  Image2.Picture = chave_v_des.Picture
  atua_relé = 4
  chave4 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
  Label7.ForeColor = &HC0&
  Label7.Caption = "Falha ao Escrever"
Else
  Label7.ForeColor = &H8000&
  Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

End Sub

Private Sub Image10_Click()
' chave 1
If chave1 = 0 Then
  Image10.Picture = chave_h_lig.Picture
  atua_relé = 1 + 32
  chave1 = 1
Else
  Image10.Picture = chave_h_des.Picture
  atua_relé = 1
  chave1 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
  Label7.ForeColor = &HC0&
  Label7.Caption = "Falha ao Escrever"
Else
  Label7.ForeColor = &H8000&
  Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

End Sub

Private Sub Image11_Click()
```

```
' chave 3
If chave3 = 0 Then
    Image11.Picture = ch3_p1.Picture
    atua_relé = 3
    chave3 = 1
Else
    Image11.Picture = ch3_p2.Picture
    atua_relé = 3 + 32
    chave3 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image12_Click()

' troca pela saída 9
If chave2 = 0 Then
    Image12.Picture = chave_h_lig.Picture
    atua_relé = 10 + 32
    chave2 = 1
Else
    Image12.Picture = chave_h_des.Picture
    atua_relé = 10
    chave2 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

End Sub

Private Sub Image2_Click()
' chave 4
If chave4 = 0 Then
    Image1.Picture = chave_v_lig.Picture
    Image2.Picture = chave_v_lig.Picture
    atua_relé = 4 + 32
    chave4 = 4
Else
    Image1.Picture = chave_v_des.Picture
    Image2.Picture = chave_v_des.Picture
    atua_relé = 4
```

```
    chave4 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image3_Click()
' chave 5
If chave5 = 0 Then
    Image3.Picture = chave_v_lig.Picture
    Image4.Picture = chave_v_lig.Picture
    atua_relé = 5 + 32
    chave5 = 1
Else
    Image3.Picture = chave_v_des.Picture
    Image4.Picture = chave_v_des.Picture
    atua_relé = 5
    chave5 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If

End Sub

Private Sub Image4_Click()
' chave 5
If chave5 = 0 Then
    Image3.Picture = chave_v_lig.Picture
    Image4.Picture = chave_v_lig.Picture
    atua_relé = 5 + 32
    chave5 = 1
Else
    Image3.Picture = chave_v_des.Picture
    Image4.Picture = chave_v_des.Picture
    atua_relé = 5
    chave5 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
```

```
Label7.ForeColor = &H8000&
Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image5_Click()
' chave 6
If chave6 = 0 Then
Image5.Picture = chave_v_lig.Picture
Image6.Picture = chave_v_lig.Picture
atua_relé = 6 + 32
chave6 = 1
Else
Image5.Picture = chave_v_des.Picture
Image6.Picture = chave_v_des.Picture
atua_relé = 6
chave6 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
Label7.ForeColor = &HC0&
Label7.Caption = "Falha ao Escrever"
Else
Label7.ForeColor = &H8000&
Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image6_Click()
' chave 6
If chave6 = 0 Then
Image5.Picture = chave_v_lig.Picture
Image6.Picture = chave_v_lig.Picture
atua_relé = 6 + 32
chave6 = 6
Else
Image5.Picture = chave_v_des.Picture
Image6.Picture = chave_v_des.Picture
atua_relé = 6
chave6 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
Label7.ForeColor = &HC0&
Label7.Caption = "Falha ao Escrever"
Else
Label7.ForeColor = &H8000&
Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image7_Click()
' chave 7
```

```
If chave7 = 0 Then
    Image7.Picture = chave_v_lig.Picture
    atua_relé = 7 + 32
    chave7 = 1
Else
    Image7.Picture = chave_v_des.Picture
    atua_relé = 7
    chave7 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image8_Click()
' chave 8
If chave8 = 0 Then
    Image8.Picture = chave_v_lig.Picture
    atua_relé = 8 + 32
    chave8 = 1
Else
    Image8.Picture = chave_v_des.Picture
    atua_relé = 8
    chave8 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
    Label7.Caption = "Falha ao Escrever"
Else
    Label7.ForeColor = &H8000&
    Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Image9_Click()
' chave 9
If chave9 = 0 Then
    Image9.Picture = chave_v_lig.Picture
    atua_relé = 9 + 32
    chave9 = 1
Else
    Image9.Picture = chave_v_des.Picture
    atua_relé = 9
    chave9 = 0
End If
If FT_Write(Inghandle, "smr" & Chr(atua_relé), 4, lngBytesWritten) <> FT_OK Then
    Label7.ForeColor = &HC0&
```

```
Label7.Caption = "Falha ao Escrever"
Else
Label7.ForeColor = &H8000&
Label7.Caption = "Sucesso ao Escrever no FTDI"
End If
End Sub

Private Sub Label1_Click()
p_med4 = p_med4 + 1

Select Case p_med4
Case 1
Text1.Text = "Canal 1"
Case 2
Text1.Text = "Canal 2"
Case 3
Text1.Text = "Ground"
End Select
If p_med4 = 3 Then
p_med4 = 0
End If
End Sub

Private Sub Label1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
Combo3.Visible = True
End Sub

Private Sub Label2_Click()
p_med3 = p_med3 + 1

Select Case p_med3
Case 1
Text1.Text = "Ponto 2"
Case 2
Text2.Text = "Ponto 2"
Case 3
Text3.Text = "Ponto 2"
End Select
If p_med3 = 3 Then
p_med3 = 0
End If
End Sub

Private Sub Label2_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
Combo2.Visible = True
End Sub

Private Sub Label3_Click()
```

```
'p_med2 = p_med2 + 1
```

```
Select Case p_med2
```

```
Case 1
```

```
Text1.Text = "Ponto 1"
```

```
Case 2
```

```
Text2.Text = "Ponto 1"
```

```
Case 3
```

```
Text3.Text = "Ponto 1"
```

```
End Select
```

```
If p_med2 = 3 Then
```

```
p_med2 = 0
```

```
End If
```

```
End Sub
```

```
Private Sub Label3_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
Combo1.Visible = True
```

```
End Sub
```

```
Private Sub Label5_Click()
```

```
Combo4.Visible = True
```

```
End Sub
```

```
Private Sub Label6_Click()
```

```
p_med5 = p_med5 + 1
```

```
Select Case p_med5
```

```
Case 1
```

```
Text1.Text = "Canal 1"
```

```
Case 2
```

```
Text1.Text = "Canal 2"
```

```
Case 3
```

```
Text1.Text = "Ground"
```

```
End Select
```

```
If p_med5 = 3 Then
```

```
p_med5 = 0
```

```
End If
```

```
End Sub
```

```
Private Sub Label4_Click(Index As Integer)
```

```
ponto_m = Index
```

```
Combo1.Visible = True
```

```
Combo1.Text = "Sel. de Canal"
```

```
Combo1.Top = Label4(Index).Top
```

```
Combo1.Left = Label4(Index).Left + 250
```

```
End Sub
```