

**UNIVERSIDADE FEDERAL DO RIO DE JANEIRO**  
**ESCOLA DE ENGENHARIA**  
**DEPARTAMENTO DE ELETRÔNICA**

**TESTE E CORREÇÃO DE SISTEMA DE AQUISIÇÃO DE DADOS**  
**EM DUTOS DE PETRÓLEO**

**Autor:**

---

**Fábio de Azevedo Petra Bittencourt**

**Orientador:**

---

**Mário Vaz da Silva Filho, D.Sc.**

**Examinador:**

---

**Antônio Carneiro de Mesquita Filho, Dr. D'État**

**Examinador:**

---

**Edson do Prado Granja, D.Sc.**

**Examinador:**

---

**Fábio da Silva Dutra, M.Sc.**

**DEL**

**Abril de 2003**

Aos meus pais Marcos e Sonia

Ao meu irmão Nelson

À minha noiva Tatiane

## AGRADECIMENTOS

Gostaria de agradecer às seguintes pessoas que, direta ou indiretamente, me ajudaram a fazer esse trabalho:

Ao amigo Fábio Dutra, pela paciência, tempo e dedicação cedidos; e pelas valiosas informações fornecidas, sem as quais não seria de maneira alguma possível realizar o trabalho.

Ao Professor Mário Vaz, pela perspicaz orientação e por todo empenho e paciência com que me acompanhou no trabalho.

Ao Professor Fico, que infelizmente não pôde participar da banca examinadora, por seus esclarecimentos essenciais.

Ao Professores Mesquita e Granja, pela participação na banca examinadora.

Ao pessoal do Laboratório de Metalografia, pela gentileza em ceder o uso de seus equipamentos de microscopia.

Ao Lafex/CBPF, por ter gentilmente cedido o analisador lógico para as medidas.

A todo o pessoal do LPC, pelas dicas, ajuda e descontração.

Aos meus amigos da faculdade, pelo apoio e motivação que me deram para que eu concluísse o curso de Engenharia.

A todos os meus amigos e quem mais eu possa eventualmente ter aqui esquecido.

E devo agradecer profundamente às seguintes pessoas:

Aos meus pais, pelo carinho e orientação que me proporcionaram ao longo da vida, erguendo os pilares básicos para minha formação, tanto profissional, quanto como ser humano.

Ao Nelson, pelo companheirismo que só um verdadeiro irmão pode dar.

À minha futura esposa Tatiane, por todo amor, carinho e compreensão que ninguém mais poderia me dar, para continuarmos sempre unidos em nossa jornada.

A Deus, por ser a inspiração de Tudo.

“É somente nas misteriosas equações do amor  
que qualquer lógica ou razão pode ser encontrada.”

Trecho do discurso de John F. Nash no filme  
“Uma Mente Brilhante”, de Ron Howard (2001)

## RESUMO

Este trabalho documenta os testes para caracterização e validação do DAQP02, um circuito integrado para aquisição de dados desenvolvido pelo Laboratório de Projeto de Circuitos Integrados (LPC) da UFRJ, para uso em um *Pig*, que é um equipamento para detecção de falhas em oleodutos.

É também descrito neste trabalho o processo de diagnóstico e correção das falhas encontradas no DAQP02 e o circuito eletrônico onde ele é usado, o PIGTP. Por fim, uma nova proposta de implementação para o equipamento, baseada em microcontrolador comercial, é apresentada.

## **PALAVRAS-CHAVE**

1. Teste e testabilidade de circuitos integrados
2. Sistema de aquisição de dados
3. Instrumentação eletrônica
4. Inspeção de oleodutos

# ÍNDICE

<b>Capítulo 1. Introdução .....</b>	<b>1</b>
1.1. Objetivo .....	1
1.2. Organização do trabalho .....	2
<b>Capítulo 2. Pigs - Tecnologia e Funcionamento .....</b>	<b>4</b>
2.1. Motivação .....	4
2.2. Pigs Instrumentados .....	4
2.3. Estrutura de um Pig Instrumentado .....	5
<b>Capítulo 3. DAQP02 .....</b>	<b>8</b>
3.1. Especificações técnicas do DAQP02 .....	8
3.2. Modos de operação do DAQP02 .....	9
3.2.1. Modo de Inicialização .....	10
3.2.2. Modo de Aquisição .....	10
3.2.3. Modo de Leitura .....	12
3.2.4. Modo de Teste .....	13
3.3. Arquitetura do DAQP02 .....	13
3.3.1. Condicionador de Sinais .....	14
3.3.2. Circuito Amostrador .....	15
3.3.3. Conversor A/D .....	15
3.3.4. Controlador .....	16
3.4. <i>Floorplan</i> e <i>layout</i> do DAQP02 .....	18
3.5. Pinagem do DAQP02 .....	19
<b>Capítulo 4. PIGTP .....</b>	<b>22</b>
4.1. Especificações técnicas do PIGTP .....	22
4.2. Descrição do <i>hardware</i> do PIGTP .....	23

4.2.1. Componentes da placa superior .....	24
4.2.2. Componentes da placa inferior .....	24
4.3. Descrição do <i>software</i> do PIGTP .....	25
4.4. Procedimentos para operação .....	26
4.4.1. Inicialização .....	26
4.4.2. Aquisição .....	26
4.4.3. Leitura .....	26
<b>Capítulo 5. Testes e Caracterização do Sistema .....</b>	<b>29</b>
5.1. Consumo de corrente .....	29
5.2. Conversão analógico-digital .....	30
5.2.1. Teste de conversão de dados .....	30
5.2.2. Teste de linearidade .....	32
5.3. Circuitos digitais .....	35
5.3.1. Testes do Controlador .....	35
5.3.1.1. Modo de Inicialização .....	36
5.3.1.2. Modo de Aquisição .....	38
5.3.1.3. Modo de Leitura .....	41
5.3.2. Comunicação com o <i>software</i> .....	43
<b>Capítulo 6. Diagnóstico e Correção de Falhas .....</b>	<b>45</b>
6.1. Microscopia .....	46
6.2. Verificação de esquemático e <i>layout</i> .....	47
6.3. Testes em bancada .....	49
6.4. Soluções adotadas .....	50
6.4.1. Correção de trilhas nas placas .....	50
6.4.2. Multiplexador analógico .....	50
6.4.3. <i>Buffer</i> .....	52
6.4.4. Escrita na memória .....	54



<b>Capítulo 7. Conclusão e Trabalhos Futuros .....</b>	<b>55</b>
<b>Bibliografia .....</b>	<b>59</b>
<b>Anexo I – Esquemáticos e <i>Layouts</i> do PIGTP .....</b>	<b>61</b>
<b>Anexo II – Manuais de Componentes .....</b>	<b>65</b>

# Capítulo 1

## Introdução

---

Em projetos de engenharia, há muitas etapas entre o desenvolvimento de um protótipo inicial e a apresentação do produto final. No caso de dispositivos ou equipamentos eletrônicos, uma etapa essencial é o teste funcional após a fabricação ou montagem. Tal verificação é feita através de diversas baterias de testes com o intuito de verificar cada uma das especificações.

Se através dos testes for verificado que o equipamento encontra-se fora das especificações, deve-se tentar descobrir o motivo. Caso sejam encontradas falhas no protótipo, pode-se optar por corrigir tais falhas ou partir para o desenvolvimento de uma nova versão do produto.

### 1.1. Objetivo

---

Este trabalho visa documentar um procedimento de testes adotado para caracterização e validação do DAQP02, um circuito integrado para aquisição de dados desenvolvido pelo Laboratório de Projeto de Circuitos Integrados (LPC) da UFRJ. O DAQP02 foi projetado para a Petrobrás, para uso num *Pig* em desenvolvimento, que faria amostragem de temperatura e

pressão ao longo de um oleoduto. Esta aplicação exige um circuito que apresente volume, peso e consumo de potência reduzidos, para que possa ser introduzido em dutos e operar por várias horas. O circuito deve ser confiável e robusto, além de apresentar boa linearidade, precisão e estabilidade. O teste deste CI também é interessante para o aprendizado de técnicas de testes e medidas.

## **1.2. Organização do trabalho**

---

**Capítulo 1. Introdução** – Este capítulo faz uma breve descrição do trabalho, apresentando a motivação que levou ao seu desenvolvimento e seus objetivos;

**Capítulo 2. Pigs – Tecnologia e Funcionamento** – Este capítulo dá uma breve explicação sobre *Pigs* e suas aplicações, mostrando também a estrutura básica de um *Pig* Instrumentado e diversas formas para sua implementação;

**Capítulo 3. DAQP02** – Este capítulo descreve brevemente o funcionamento e a arquitetura do circuito integrado DAQP02;

**Capítulo 4. PIGTP** – Aqui temos a descrição da estrutura e do funcionamento do sistema eletrônico de aquisição de dados PIGTP, além dos procedimentos a serem seguidos para sua operação;

**Capítulo 5. Testes e Caracterização do Sistema** – Este capítulo descreve os procedimentos utilizados para os testes de cada parte do sistema de aquisição de dados, assim como os respectivos resultados observados;

**Capítulo 6. Diagnóstico e Correção de Falhas** – Este capítulo descreve os métodos usados para se diagnosticar os problemas encontrados no sistema de aquisição de dados, bem como as soluções encontradas para corrigí-los;

**Capítulo 7. Conclusão e Trabalhos Futuros** – Aqui são mostrados os objetivos e resultados alcançados. Também é apresentada uma alternativa para implementação do sistema de aquisição de dados, usando um microcontrolador;

## Capítulo 2

### *Pigs* – Tecnologia e Funcionamento

---

#### 2.1 Motivação

---

O transporte de óleo através de dutos é muito utilizado pela indústria petrolífera. Os dutos estão sujeitos a falhas que podem comprometer a operação da linha, resultando em grandes prejuízos, em caso de vazamento do produto.

Dessa forma, há uma grande preocupação da indústria petrolífera com o monitoramento das condições dessas estruturas, para permitir uma manutenção preventiva eficiente. Para essa tarefa são usados equipamentos especiais, conhecidos por *Pigs*.

A Petrobrás desenvolve diversos tipos deste equipamento, dominando a tecnologia e prestando serviços importantes no setor petrolífero, da manutenção de dutos à proteção ambiental.

#### 2.2. *Pigs* Instrumentados

---

Um *Pig* é um equipamento que se desloca junto com o óleo transportado, permitindo a limpeza do duto. Entretanto, somente a limpeza não é suficiente para uma manutenção eficaz

do duto. É necessário obter informações para detecção de falhas prejudiciais ao processo, tais como vazamentos e nível de corrosão das paredes do duto.

A solução encontrada consiste em adicionar à estrutura do *Pig* um sistema de aquisição de dados que faça a coleta de informações durante a limpeza do duto. Este equipamento é chamado de *Pig Instrumentado* ou *Smart Pig*. Num *Pig Instrumentado*, sensores e circuitos eletrônicos são inseridos à sua estrutura e, normalmente, um receptáculo impermeável e resistente a vibração e alta pressão é usado para comportar a eletrônica necessária. Como fatores de mérito no projeto de um *Pig Instrumentado*, podemos mencionar: baixo consumo de energia, permitindo aumento da vida da bateria e maior tempo de operação; alta capacidade de armazenamento, para maior autonomia do equipamento; e miniaturização do sistema, que é importante para dutos estreitos (diâmetro < 100 mm).

Existem diversas técnicas empregadas em *Pigs Instrumentados*, sendo que cada uma delas possibilita a detecção de um certo tipo de falha. Entre as principais técnicas utilizadas, temos: fluxo magnético, ultra-som, variação de pressão, variação de temperatura, uso de giroscópios e uso de câmera de vídeo. Entre as principais falhas que podemos encontrar nos oleodutos, temos: corrosão da estrutura por oxidação; falha no isolamento térmico, em caso de óleos aquecidos; anomalias na geometria do duto, como formação de dentes e ovalização; depósito de substâncias nas paredes internas; e vazamentos, que constituem a situação mais crítica.

### **2.3. Estrutura de um *Pig Instrumentado***

---

Para se ter uma melhor idéia do seu funcionamento, vamos ilustrar aqui um exemplo de estrutura para um *Pig Instrumentado*. Antes do desenvolvimento do equipamento, deve-se decidir qual técnica de detecção será utilizada, já que isto influencia diretamente na maneira com que os dados serão coletados. Vamos levar em consideração um sistema de aquisição de dados baseado na análise da variação de temperatura, pressão, aceleração e/ou vibração.

Nesse caso, o *Pig* proposto deve realizar as seguintes funções: coletar informações provenientes da leitura de sensores; converter os sinais analógicos coletados em sinais digitais; e transferir o conteúdo da memória para um computador externo, permitindo uma análise dos dados *off-line*. Para a realização das tarefas mencionadas, a estrutura básica de um *Pig Instrumentado* é composta por três blocos, ilustrados na Figura 2.1:

- **Estrutura Mecânica:** apresenta características que permitem a limpeza do duto. Além de odômetros acoplados, sua estrutura serve como suporte para o sistema eletrônico de aquisição de dados.
- **Sistema de Aquisição de Dados:** fica embutido no corpo do *Pig*, onde sensores permanecem em contato com o óleo e são responsáveis pela leitura das informações necessárias. Esse é um equipamento portátil e sua alimentação é fornecida por baterias.
- **Software:** programa que implementa a comunicação entre o equipamento e um computador, permitindo a análise dos dados coletados.

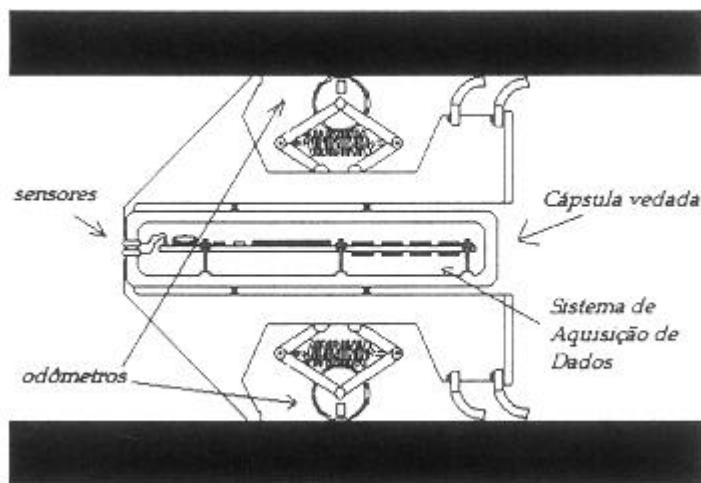


Figura 2.1 Estrutura de um *Pig* Instrumentado

Existem no mercado diversos dispositivos que podem ser utilizados para implementar o sistema de aquisição de dados proposto. Entre estes dispositivos, temos os microcontroladores ( $\hat{C}$ ), os processadores digitais de sinais (DSP) e os dispositivos de lógica programável (CPLD/FPGA).

Uma outra alternativa é o desenvolvimento de um processador dedicado para realizar as funções de aquisição de dados necessárias ao *Pig* Instrumentado. Como exemplo de implementação dessa alternativa temos o circuito integrado DAQP02 e o sistema eletrônico de aquisição de dados PIGTP. O PIGTP é um sistema composto de *hardware* e *software* que utiliza o DAQP02 como processador central. Esse sistema de aquisição de dados foi desenvolvido pelo Laboratório de Projeto de Circuitos Integrados (LPC) da UFRJ em parceria com o Centro de Pesquisas da Petrobrás (CENPES). Na época em que começou a ser

desenvolvido, um sistema dedicado apresentava muitas vantagens em relação aos dispositivos existentes no mercado, permitindo atender facilmente aos fatores mencionados: baixo consumo, miniaturização do sistema e alta capacidade de memória. Maiores detalhes sobre esta implementação serão descritos nos capítulos a seguir.



## Capítulo 3

### DAQP02

---

Este capítulo descreve o processador DAQP02, um circuito integrado dedicado para aquisição de dados em dispositivo *Pig* para manutenção de dutos de petróleo. O circuito integrado foi projetado pelo Laboratório de Projeto de Circuitos Integrados (LPC) da UFRJ, empregando técnicas de integração de circuitos analógicos e digitais em uma mesma pastilha, com auxílio do *software* Cadence e implementação de códigos em VHDL. Foram fabricadas 10 unidades deste circuito integrado através da empresa CMP, e mais quatro unidades através do programa PMU/FAPESP. Esta é a segunda versão do circuito, feita para ser usada no *Pig* de pressão e temperatura desenvolvido pelo CENPES, corrigindo alguns problemas de sua versão anterior, o DAQP01 [3].

#### 3.1. Especificações técnicas do DAQP02

---

O DAQP02 apresenta as seguintes especificações:

- Capacidade de monitoramento de dois sinais analógicos, podendo ser pressão e temperatura;
- Capacidade de armazenamento dos dados em até 4 Mbytes de memória;
- Necessita somente de circuitos pré-condicionadores e memória para implementação de um *Pig Instrumentado*;
- Tecnologia de integração: AMS CMOS 0,8  $\mu$ m CYE;
- Encapsulamento do tipo JLCC com 84 pinos;
- Alimentação de 5V, frequência de operação de 10 Hz a 200 kHz e excursão do sinal de entrada de 0 a 2,522 V;
- Barramento de dados de 8 bits;
- Barramento de endereços de 22 bits;

### 3.2. Modos de operação do DAQP02

---

São quatro os modos de operação do DAQP02: Inicialização, Aquisição, Leitura e Teste. A Figura 3.1 apresenta a seqüência dos três primeiros:

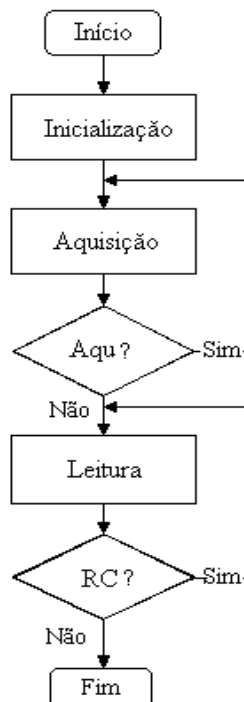


Figura 3.1 – Fluxograma dos Modos de Operação do DAQP02.

O sinal Aqu permite a seleção entre os modos de Leitura e Aquisição. O sinal RC permite reiniciar o processo de Leitura.

### 3.2.1. Modo de Inicialização

O sinal Reset faz o processador entrar no modo de Inicialização, ilustrado na Figura 3.2. O conteúdo da memória é preenchido com o valor (FF)h, valor este nunca gerado pelo conversor A/D em funcionamento normal. Portanto, a leitura deste dado indica que o dado anterior foi o último a ser armazenado na memória, o que pode ser usado para recuperação de dados da memória em caso de perda de alimentação durante a aquisição de dados. Conforme será descrito na Seção 3.2.4, o modo de Inicialização pode ser desabilitado para realização de testes.

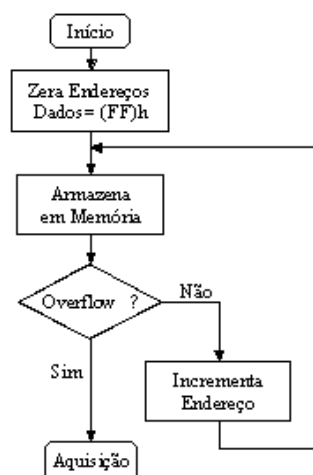


Figura 3.2 – Fluxograma do Modo de Inicialização do DAQP02.

### 3.2.2. Modo de Aquisição

Neste modo de operação, o DAQP02 inicia o processo de leitura e armazenamento dos sinais dos sensores, conforme a Figura 3.3. O sinal Aqu possibilita a seleção entre os modos de Aquisição e Leitura.

Quando em modo de Aquisição, o DAQP02 fica à espera de pulsos do odômetro para fazer a aquisição de dados. Os sinais dos sensores são multiplexados para um único conversor A/D. Para cada pulso do odômetro, uma amostra de sinal de cada sensor é convertida em uma

palavra de 8 bits armazenada em memória. Se não houver pulso no odômetro, não haverá aquisição de dados.

O fim do modo de Aquisição ocorrerá quando toda a memória estiver preenchida, ou o modo de operação for modificado através do Aqu.

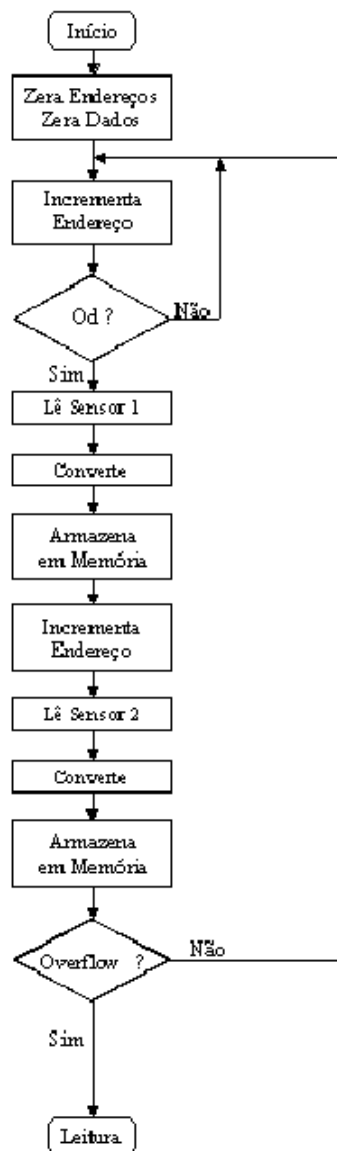


Figura 3.3 – Fluxograma do Modo de Aquisição do DAQP02.

### 3.2.3. Modo de Leitura

Os dados armazenados em memória podem ser lidos através da porta paralela de um computador IBM-PC ou compatível. Para este modo de operação foi implementado um protocolo de comunicação de 4 bits, para leitura segundo o padrão EPP (*Enhanced Parallel Port*), sendo para isto utilizados os sinais REQ, ACK, RC e EOF, conforme a Figura 3.4.

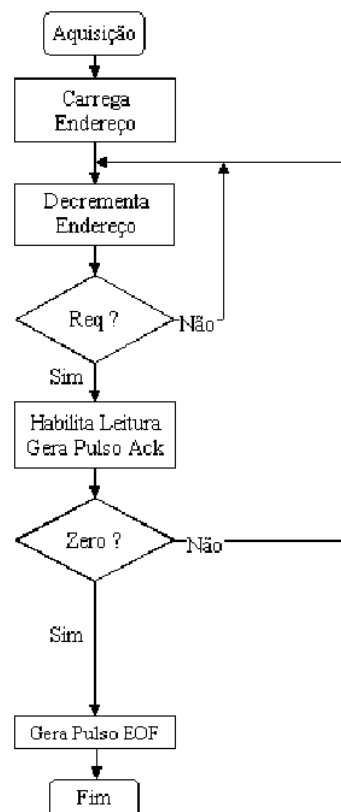


Figura 3.4 – Fluxograma do Modo de Leitura do DAQP02.

O pulso REQ faz gerar um novo endereço, habilitar a leitura da memória e enviar um pulso de ACK indicando que a leitura foi realizada com sucesso. Quando todo o conteúdo da memória for lido, o DAQP02 coloca o sinal EOF em nível lógico alto, permanecendo assim até início de outro modo de operação. Através do sinal RC, pode-se reiniciar todo o processo de leitura da memória.

### 3.2.4. Modo de Teste

Este modo permite a realização dos testes do processador DAQP02, desabilitando o controlador, para entrada de sinais externos de controle, o que pode ser feito através de um dispositivo de lógica programável. Assim, é possível verificar o funcionamento dos circuitos analógicos, se houver algum problema na lógica digital do CI.

### 3.3. Arquitetura do DAQP02

Na Figura 3.5 temos o diagrama de blocos do DAQP02, que foi desenvolvido em seis blocos:

- **Condicionador de Sinais** - para tratamento dos sinais provenientes dos sensores;
- **Conversor A/D (8 bits)** - para conversão analógico-digital dos dados;
- **Multiplexador (Mux)** - para entrada dos sinais de controle externamente;
- **Controlador** - para controle da conversão, escrita e leitura dos dados em memória;
- **Power On-Reset** - responsável pela inicialização automática do processador;
- **Latch** - permite a leitura dos dados em memória pelo computador.

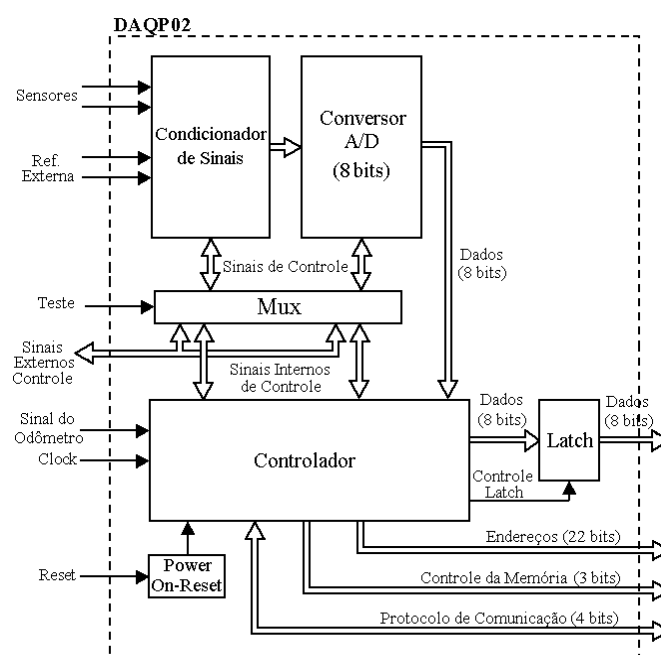


Figura 3.5 – Diagrama de Blocos do DAQP02.

A seguir temos uma descrição mais detalhada dos blocos principais: Condicionador de Sinais, Conversor A/D e Controlador.

### 3.3.1. Condicionador de Sinais

O Condicionador de Sinais é o circuito responsável pelo tratamento dos sinais dos sensores, ajustando os níveis de tensão para a conversão A/D. Os circuitos representados pelos da Figura 3.6 são descritos a seguir:

- **Mux 1 e Mux 2** – multiplexadores analógicos
- **Buffer** – isoladores
- **LevelShift (Deslocador de Nível)** – constituído por amplificadores operacionais e resistores. Em [2] mostra-se que a relação entre a saída e a entrada é dada pela equação:  $V_{out} = 1,261 V + V_{in}$
- **GenRef (Gerador de Referência)** – gera as seguintes tensões de referência máxima e mínima para conversão digital dos dados:  $V_{REFN} = 1,261 V$  e  $V_{REFP} = 3,783 V$ .

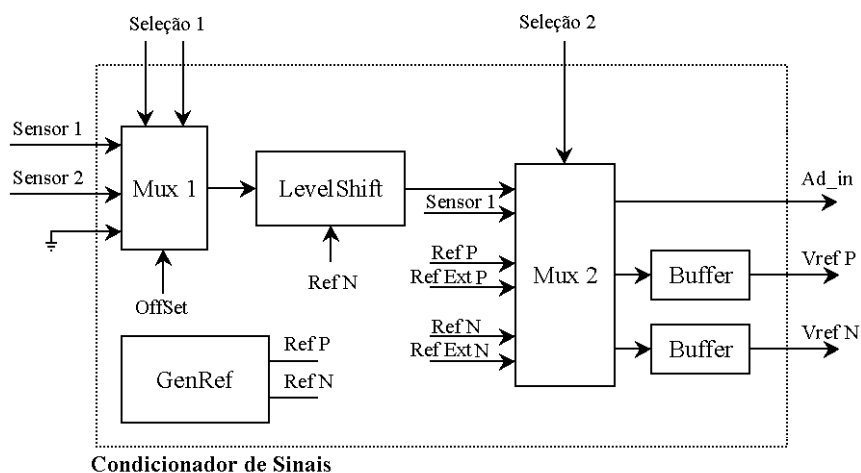


Figura 3.6 – Diagrama de Blocos do Condicionador de Sinais do DAQP02.

### 3.3.2 Circuito Amostrador

A saída do Condicionador de Sinais está ligada ao Circuito Amostrador (*Sample and Hold*), que retém o sinal durante a conversão analógico-digital. O amplificador operacional configurado como *buffer* possui um *fan-out* baixo e não é capaz de fornecer corrente suficiente ao conversor A/D. Assim, existem capacitores para manter a estabilidade do sinal na entrada do conversor A/D. Esses capacitores, de poli-silício e com dimensão de 140,0 x 140,1  $\mu\text{m}$ , são maiores que os amplificadores operacionais, de forma a fornecer a corrente necessária à entrada do conversor A/D.

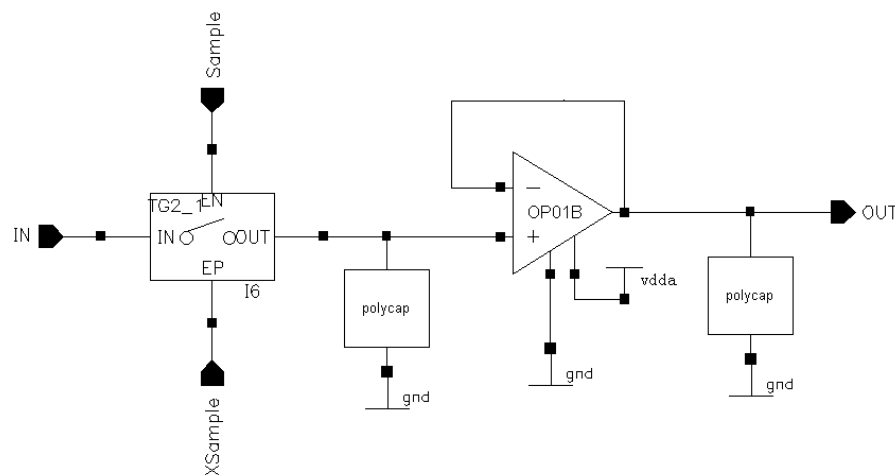


Figura 3.7 – Circuito Amostrador do DAQP02.

### 3.3.3. Conversor A/D

O Conversor A/D utiliza a técnica de aproximações sucessivas [2], que requer pelo menos  $N$  períodos de *clock* para converter  $N$  bits, e que a amostra permaneça durante todo este tempo, exigindo um circuito amostrador que mantenha a amostra do sinal. Sua arquitetura utiliza um divisor resistivo e uma matriz de capacitores para redistribuição de cargas, como pode ser visto no diagrama de blocos da Figura 3.8.

A conversão do dado se completa em 10 ciclos do *clock*. O primeiro ciclo inicia a operação, os 8 ciclos seguintes são utilizados para cálculo do resultado, e o último ciclo garante que o dado na saída seja estável. A faixa de conversão é definida por (VREFP –



VREFN). O conversor possui dois modos de operação, *Handshake* e *Direto* [2], que não serão detalhados aqui.

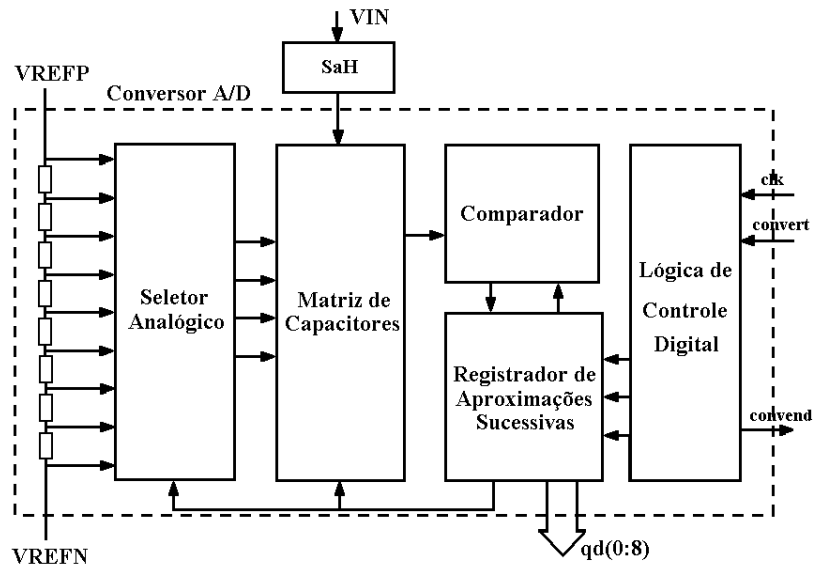


Figura 3.8 – Diagrama de Blocos do Conversor A/D do DAQP02.

### 3.3.4. Controlador

O Controlador é responsável pelas seguintes operações: seleção dos sensores, início da conversão dos dados, controle do processo de escrita dos dados em memória, controle da comunicação com o PC e do processo de leitura dos dados via *software*.

Dois contadores de 22 bits são responsáveis pelo endereçamento da memória, sendo um deles utilizado no modo de Aquisição, e o outro no modo de Leitura. Com o último endereço gerado no modo de Aquisição está sempre armazenado, um comparador de 22 bits determina o fim de operação do modo de Leitura, através da comparação dos dois contadores. O sinal AQU seleciona qual contador fornecerá o valor de endereço a ser usado.

A Figura 3.9 traz um diagrama de blocos do Controlador. Seus elementos são os seguintes:

- **Sinc** – bloco responsável pelo sincronismo do sinal do odômetro com o *clock* do circuito e divisão da frequência por 128;
- **Controle Mestre** – responsável pela geração do pulso de controle de todo o sistema;

- **Controle A/D** – responsável pelo início da conversão dos dados, e seleção dos sensores. A cada pulso do sinal do Controle Mestre, dá-se início a duas conversões em seqüência;
- **Controle Memória** – este bloco controla a leitura, a escrita em memória e o protocolo de comunicação;
- **Contador 1** – responsável pela geração dos endereços quando em modo de Aquisição;
- **Contador 2** – responsável pela geração dos endereços quando em modo de Leitura;
- **Mux** – multiplexador responsável pela seleção dos dois barramentos de endereços disponíveis;
- **Comparador** – responsável pela indicação de fim do modo de Leitura, comparando o conteúdo dos dois contadores;
- **(FF)h => (FE)h** – não permite que o valor (FF)h seja utilizado pelos dados, convertendo-os para (FE)h.

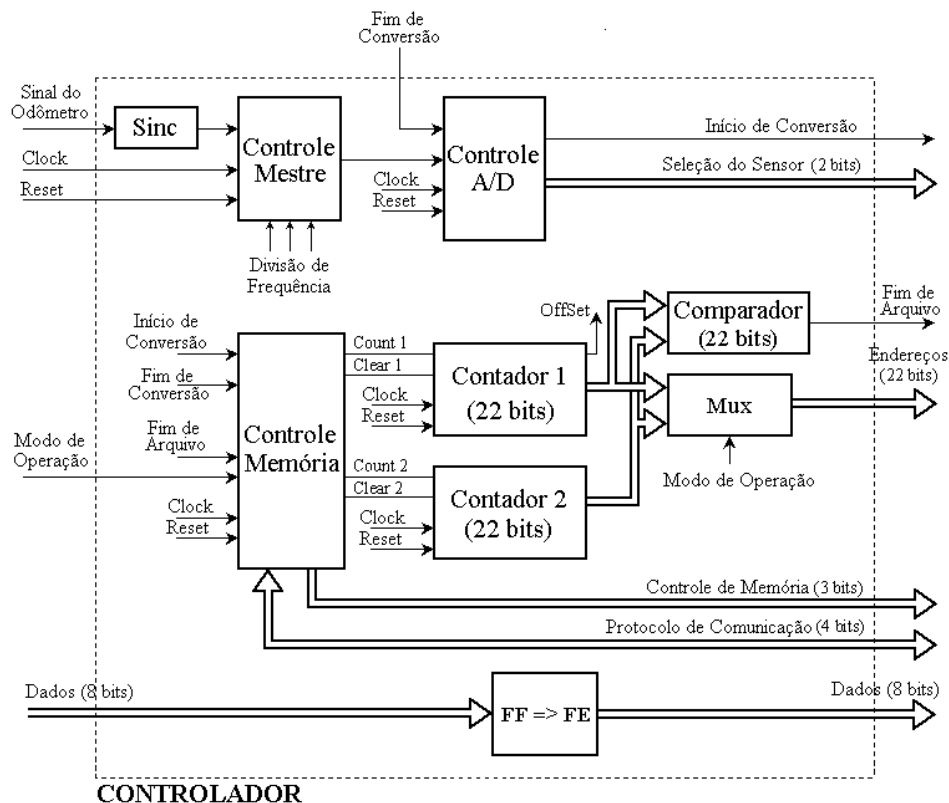


Figura 3.9 - Diagrama de Blocos do Controlador.

### 3.4. Floorplan e layout do DAQP02

Na Figura 3.10, pode-se ver o *floorplan* (“planta baixa”) do DAQP02 com o posicionamento de cada bloco do circuito. Na Figura 3.11, tem-se o *layout* final. As dimensões do CI são 10,6 mm<sup>2</sup> (3,362 x 3,156 mm), implementado em tecnologia AMS CMOS 0,8μ CYE, com um total de 84 pinos.

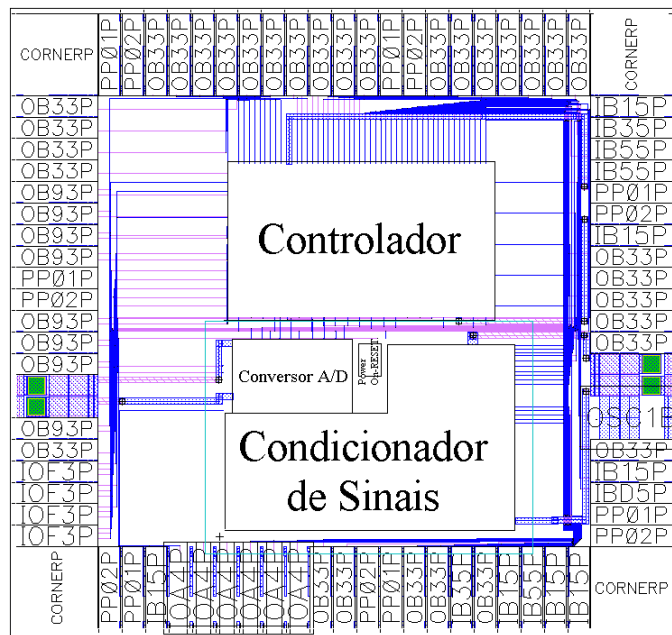


Figura 3.10 – Planta baixa do DAQP02.

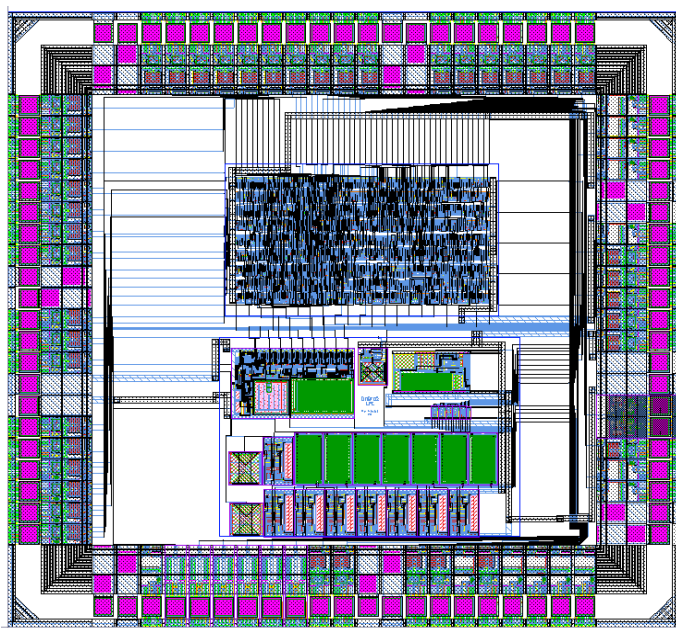


Figura 3.11 – Layout do DAQP02.

### 3.5. Pinagem do DAQP02

---

Esta seção traz uma descrição da função dos 84 pinos do DAQP02. A Tabela 3.1 dá o número e o nome atribuído a cada pino.

- **ACK** – utilizado para comunicação com o computador no modo de Leitura (*Acknowledgment*). Indica que um endereço foi gerado e que a memória já pode ser lida;
- **ADD<21:0>** – barramento de endereços da memória;
- **ADEOC** – indica fim da conversão de um dado (*AD end of conversion*);
- **ADIN** – saída analógica que permite acesso ao sinal na entrada do conversor A/D após o Circuito Amostrador;
- **ADSTART** – indica início de conversão de um dado. Quando em modo de Aquisição, é um sinal de saída, permitindo monitorar o sinal gerado pelo controlador. Quando em modo de Teste, é um sinal de entrada, permitindo controlar o conversor A/D externamente;
- **AQU** – permite mudar o modo de operação entre Aquisição e Leitura. Quando em nível lógico alto “1”, está em modo de Aquisição e, quando em nível lógico baixo “0”, está em modo de Leitura;
- **CLK** – é a base de tempo do sistema. O *pad* utilizado consiste de um oscilador para utilização com um cristal. Pode-se também aplicar um sinal externamente, através de um gerador de função, ao pino 38, deixando o pino 39 flutuando;
- **CLK\_INT** – possibilita entrar com sinal do *clock*. Esta entrada está depois do divisor. Sendo assim, o sinal aplicado não sofrerá divisão em frequência;
- **Ctrl\_m** – é o sinal do Controle Mestre, antes de passar pelo bloco de sincronismo;
- **Ctrl\_m1** – é o sinal do Controle Mestre, após o bloco de sincronismo;
- **Cupboot** – permite monitorar a inicialização do sistema, indicando se o processador está em Modo de Inicialização;
- **D<7:0>** – barramento de dados (8 bits);
- **DISTANCE** – entrada do sinal proveniente do odômetro;
- **DIV<2:0>** – este barramento de 3 bits possibilita a divisão da frequência do pulso do odômetro ou do *clock*;

- **EOF** – indica que toda memória foi “varrida”, quando em modo de Leitura;
- **GND, GNDA** – pinos de alimentação do circuito. Correspondem ao terra digital e analógico, respectivamente;
- **Help** – possibilita entrada com sinais de controle externo, fazendo com que o DAQP02 opere em modo de Teste;
- **LS3** – permite acesso à saída analógica do circuito deslocador de nível;
- **Normal** – indica se o DAQP02 está lendo os sensores ou se as entradas estão aterradas. É o complemento do *Offset*;
- **Offset** – permite armazenar o valor médio do ruído de fundo existente no processo, aterrando as entradas dos sensores;
- **Overflow** – indica que o contador de 22 bits, responsável pela geração dos endereços, quando em modo de Aquisição, estourou sua capacidade;
- **P** – entrada analógica de um dos sensores;
- **PEXT** – também utilizado pelo modo de Teste, habilita a entrada externa dos sinais de referência VREFNEXT e VREFPEXT;
- **RC** – permite reiniciar o processo de leitura da memória, quando em modo de Leitura (*Reset Counter*);
- **REQ** – o *Request* é utilizado para comunicação com o computador, quando em modo de Leitura. O processador deve gerar um endereço quando houver um pedido de *Request*;
- **RESET** – sinal de inicialização do DAQP02;
- **SP,ST** – seleção dos sensores. Quando em modo de Aquisição, são sinais de saída onde é possível monitorar o funcionamento do controlador do conversor. Quando em modo de Teste, permite a seleção dos sensores externamente;
- **SVAR** – seleciona a variável de controle de coleta dos dados: tempo (*clock*) ou distância (odômetro). Quando em nível lógico alto, seleciona o tempo, caso contrário, a distância;
- **T** – entrada analógica do segundo sensor;
- **VDD, VDDA** – terminais positivos de alimentação do circuito. Correspondem ao Vcc digital e analógico, respectivamente;
- **VREFNEXT** – permite entrar externamente com a tensão de referência mínima;
- **VREFPEXT** – permite entrar com a tensão de referência máxima;

- **ZBOOTENA** – permite desabilitar o modo de Inicialização. Para isto, deve estar em nível lógico alto;
- **ZCE** – *Chip Enable* barrado, habilita a memória;
- **ZOE** – *Output Enable* barrado, habilita a leitura em memória;
- **ZWE** - *Write Enable* barrado, habilita a escrita em memória.

Tabela 3.1 – Descrição dos Pinos do DAQP02.

#	Nome	Tipo	#	Nome	Tipo	#	Nome	Tipo	#	Nome	Tipo
1	D3	<i>Tri-State</i>	22	Ctrl_m1	Saída	43	ZOE	Saída	64	ADD9	Saída
2	D2	<i>Tri-State</i>	23	VDD	Fonte	44	EOF	Saída	65	ADD10	Saída
3	D1	<i>Tri-State</i>	24	GND	Fonte	45	ZWE	Saída	66	ADD11	Saída
4	GND A	Fonte	25	Ctrl_m	Saída	46	ACK	Saída	67	ADD12	Saída
5	VDD A	Fonte	26	Overflow	Saída	47	HELP	Entrada	68	ADD13	Saída
6	D0	<i>Tri-State</i>	27	ZBOOTENA	Entrada	48	VDD	Fonte	69	ADD14	Saída
7	Normal	Saída	28	ADEOC	Saída	49	GND	Fonte	70	ADD15	Saída
8	Offset	Bidirecional	29	DIV0	Entrada	50	RC	Entrada	71	ADD16	Saída
9	SP	Bidirecional	30	CLK_INT	Entrada	51	REQ	Entrada	72	ADD17	Saída
10	ST	Bidirecional	31	DIV2	Entrada	52	AQU	Entrada	73	VDD	Fonte
11	ADSTART	Bidirecional	32	DIV1	Entrada	53	SVAR	Entrada	74	GND	Fonte
12	VDD	Fonte	33	VDD	Fonte	54	ADD1	Saída	75	ADD18	Saída
13	GND	Fonte	34	GND	Fonte	55	ADD2	Saída	76	ADD19	Saída
14	RESET	Entrada	35	DISTANCE	Entrada	56	ADD3	Saída	77	ADD20	Saída
15	ADIN	Saída	36	PEXT	Entrada	57	ADD4	Saída	78	ADD21	Saída
16	T	Entrada	37	ADD0	Saída	58	ADD5	Saída	79	D7	<i>Tri-State</i>
17	VREFNEXT	Entrada	38	CLK	Entrada	59	ADD6	Saída	80	D6	<i>Tri-State</i>
18	VREFPEXT	Entrada	39	CLK	Entrada	60	ADD7	Saída	81	D5	<i>Tri-State</i>
19	LS3	Saída	40	GND A	Fonte	61	VDD	Fonte	82	D4	<i>Tri-State</i>
20	P	Entrada	41	VDD A	Fonte	62	GND	Fonte	83	GND	Fonte
21	Cupboot	Saída	42	ZCE	Saída	63	ADD8	Saída	84	VDD	Fonte

Os pinos analógicos estão marcados.

# Capítulo 4

## PIGTP

---

Este capítulo descreve o sistema eletrônico de aquisição de dados PIGTP, projetado pelo Laboratório de Projeto de Circuitos Integrados (LPC) da UFRJ, baseado no processador DAQP02, para uso no *Pig* de Temperatura e Pressão desenvolvido pelo CENPES [3]. O PIGTP é um sistema composto de *hardware* e *software*, com duas placas de circuito impresso e um programa para leitura de dados por computador.

### 4.1. Especificações técnicas do PIGTP

---

O PIGTP apresenta as seguintes especificações:

- Capacidade de aquisição dos dados de temperatura e pressão em memória NVRAM de 2 Mbytes;
- Medição e armazenamento de dados do nível de ruído presente no processo de aquisição;
- A aquisição dos dados ocorre em função do sinal pulsado do odômetro;

- Protocolo de comunicação para leitura do conteúdo da memória através da porta paralela de um computador tipo IBM PC;
- Quatro modos de operação (Inicialização, Aquisição, Leitura, e Teste);
- Alimentação de 7,2 Vdc a 60 mA máximo;
- Excursão de sinal dos sensores de 1 a 2 V;
- Taxa de aquisição máxima de 4 kHz;
- Frequência de operação de 79,36 kHz.
- Barramento de dados de 8 bits;
- Barramento de endereços de 22 bits;

#### 4.2. Descrição do *hardware* do PIGTP

Na Figura 4.1 temos o diagrama de blocos do PIGTP.

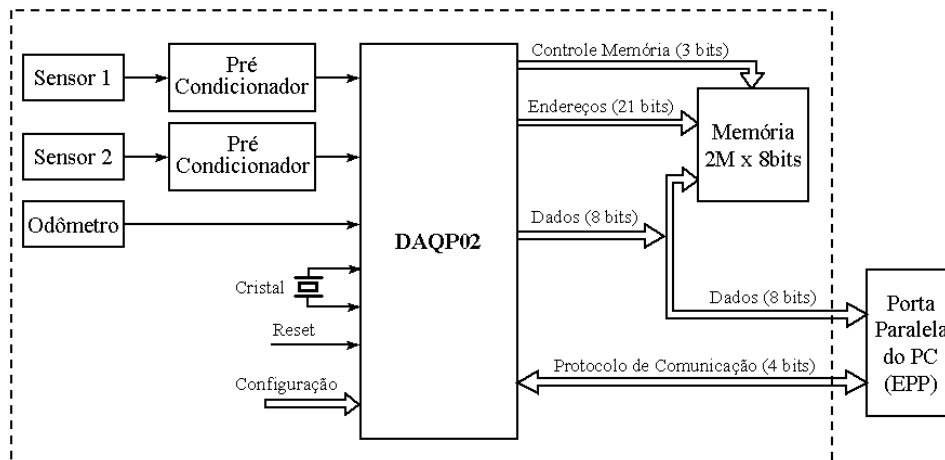


Figura 4.1 – Diagrama de Blocos do PIGTP.

O circuito eletrônico do PIGTP é composto de duas placas de circuito impresso sobrepostas. A placa superior traz o processador DAQP02 e as entradas dos sensores e do odômetro. A placa inferior traz a memória, a entrada de alimentação, circuitos condicionadores e de controle, e a conexão com a porta paralela.

A seguir temos uma descrição da função de cada componente. A numeração dos componentes, o posicionamento na placa e a função de cada pino estão indicados nas figuras e esquemáticos no Anexo I.



#### 4.2.1. Componentes da placa superior

**U1 – DAQP02** – Processador principal.

**J1 – Conector de encaixe** – Faz a comunicação do barramento de endereços com a placa inferior. Os conectores para encaixe das duas placas estão posicionados de modo a evitar erros de encaixe.

**J2 – Conector do odômetro** – O sinal proveniente do odômetro deve ser um pulso digital, que pode ser padrão CMOS ou TTL. O DAQP02 deve receber um pulso quadrado de 5 V e duração de pelo menos duas vezes o pulso de clock do processador.

**J3 – Conector do sensor S1** – Entrada do sinal do sensor de pressão S1. Os sensores de pressão são do fabricante Lucas Nova Sensor, da série NPI-15. É necessário um prévio condicionamento do sinal antes de entrar no processador. No protótipo anterior, isto era feito por um circuito de condicionamento montado na placa do PIGTP, com uso de um amplificador de instrumentação Burr-Brown INA-118. Porém, este circuito foi posteriormente acoplado ao sensor. Assim, o sinal agora é aplicado diretamente ao pino do conector.

**J4 – Conector do sensor S2** – Entrada do sinal do sensor de pressão S2

**J5 – Conector de encaixe** – Faz a comunicação do barramento de dados, da alimentação e dos sinais de controle com a placa inferior.

**C1 a C6 – Capacitores de *by-pass*** – Filtram ruídos espúrios da alimentação.

#### 4.2.2. Componentes da placa inferior

**U1 – Regulador de tensão** – O regulador REG11175 da Burr-Brown fornece 5 V para alimentação dos componentes do circuito.

**U3 – Memória** – Memória Dallas DS1270A NVRAM (RAM não volátil) para armazenamento dos dados coletados. Apesar do DAQP02 ter capacidade de endereçar 4 Mbytes, utiliza-se uma memória de 2 Mbytes.

**U4 – CI 74LS00 (portas NAND)** – Tem duas funções. A primeira é interromper o processo de aquisição em 2 Mbytes, já que o DAQP02 endereça 4 Mbytes. O circuito com as portas U4A e U4B evita que os dados sejam sobrescritos. O último bit de endereço é responsável pelo controle do pulso do odômetro, desabilitando-o assim que a capacidade ultrapassar os 2 Mbytes. A segunda função é o controle de seleção e leitura da memória (ZCE

e ZOE), que era feito anteriormente pelo DAQP02. O ZCE é mantido em GND, e o ZOE fica sob controle do sinal AQU. Esta mudança melhorou a comunicação com a porta paralela.

**J1 – Conector de interface com porta paralela** – Interface entre o equipamento e a porta paralela do computador.

**J2 – Conector de encaixe** – Faz a comunicação do barramento de endereços com a placa superior.

**J3 – Conector de encaixe** – Faz a comunicação do barramento de dados, da alimentação e dos sinais de controle com a placa superior.

**J4 – Conector de alimentação** – Entrada da alimentação de 7 V. O circuito pode ser alimentado com tensão entre 7 e 15 V.

**JP1 e JP2 – Jumpers** – Para testes com a alimentação de 5 V.

**R1 a R3 – Resistores de pull-up** – R2 e R3 estão conectados ao cabo de comunicação para os pinos REQ e RC, já que estes sinais são provenientes de transistores em coletor aberto da porta paralela do computador.

**C5 e R4 – Circuito de reset** – Circuito RC para inicialização do DAQP02.

**C1 a C4 – Capacitores de by-pass** – Filtram ruídos espúrios da alimentação.

### 4.3. Descrição do *software* do PIGTP

---

O TPIG é o *software* que permite comunicação entre o PIGTP e o computador através da porta paralela. O *software* foi desenvolvido em linguagem C++ e consiste no arquivo executável *tpig.exe*. Basta copiá-lo para o computador e rodar o arquivo. O programa é compatível com a plataforma Windows 95/98/ME. O funcionamento é garantido em máquinas com a seguinte configuração mínima: Pentium 120 MHz ou superior; 5 Mbytes de espaço em disco rígido; e 16 Mbytes de memória RAM. O PC deve suportar o modo PS2 para transferência de dados via porta paralela, caso contrário o *software* não funcionará.

## **4.4. Procedimentos para operação**

---

Uma descrição dos procedimentos para operação do PIGTP é fornecida a seguir.

### **4.4.1. Inicialização**

O sistema de inicialização do processador DAQP02 faz o preenchimento de toda memória com o valor FFh, e o equipamento ainda não está apto a operar. O circuito *power-on-reset* integrado ao DAQP02, é responsável pela inicialização automática e o equipamento pode ser reiniciado a qualquer momento.

### **4.4.2. Aquisição**

Após a inicialização, com o odômetro conectado ao circuito, o equipamento iniciará a coleta de dados de ambos os sensores, para cada pulso do odômetro. O equipamento deve permanecer ligado após a aquisição de dados, para leitura da memória, caso contrário o processador perderá o conteúdo de seus registradores, e não será possível ler a memória através do *software* desenvolvido.

### **4.4.3. Leitura**

Para leitura *off-line* dos dados em memória, o cabo da porta paralela deve estar conectado para comunicação com o *software*, configurando o *hardware* para operar em modo de Leitura. O programa deve ser executado antes da conexão do *hardware* ao computador, para garantir que os sinais de comunicação estão zerados.

Ao iniciar, uma janela permite a escolha do diretório para armazenar os dados, como indica a Figura 4.9. As informações de pressão serão armazenadas no arquivo *press.txt*, e as de temperatura, no arquivo *temp.txt*.

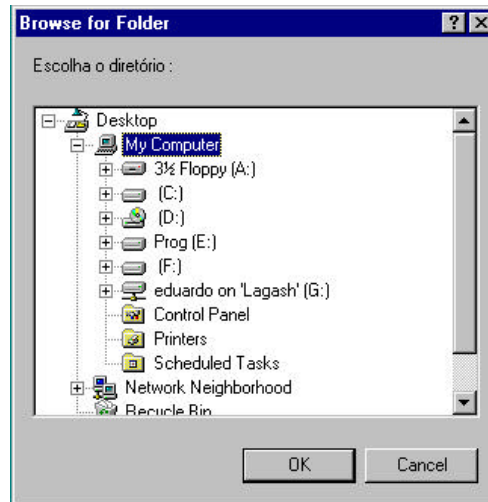


Figura 4.9 - Janela de Seleção de Diretório.

Na janela principal, indicada pela Figura 4.10, é preciso setar os valores mínimos e máximos de temperatura e pressão que serão utilizados como referência para plotagem do gráfico e armazenamento em arquivo. Para iniciar a comunicação, basta pressionar a tecla COLETAR.

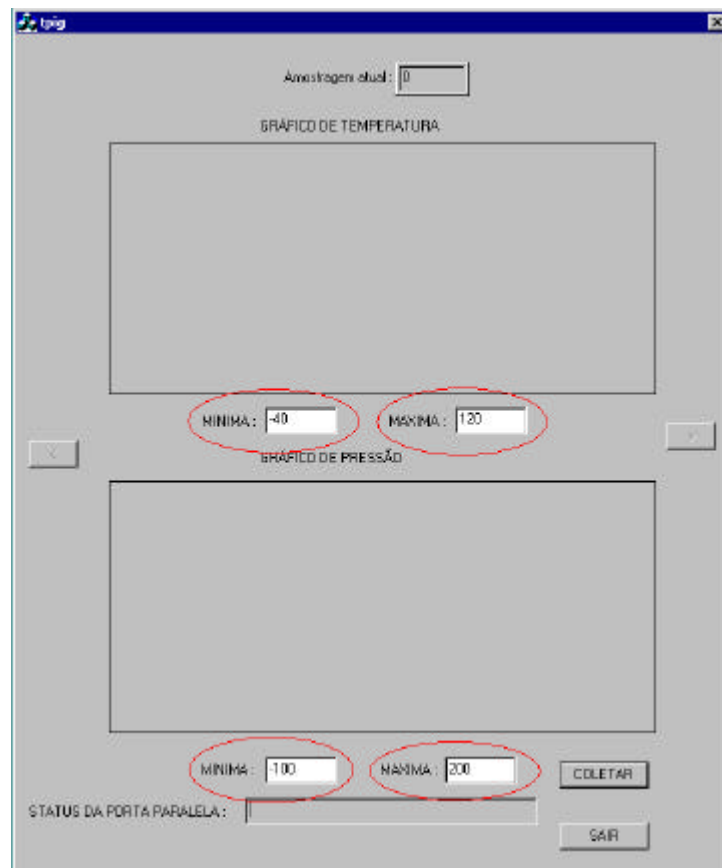


Figura 4.10 - Janela Principal do TPIG.

Os dados armazenados na memória do PIGTP serão lidos através da porta paralela. Esta operação pode levar alguns minutos, dependendo da quantidade de informação coletada. O campo STATUS DA PORTA diz se ela conseguiu ser configurada para aquisição de dados.

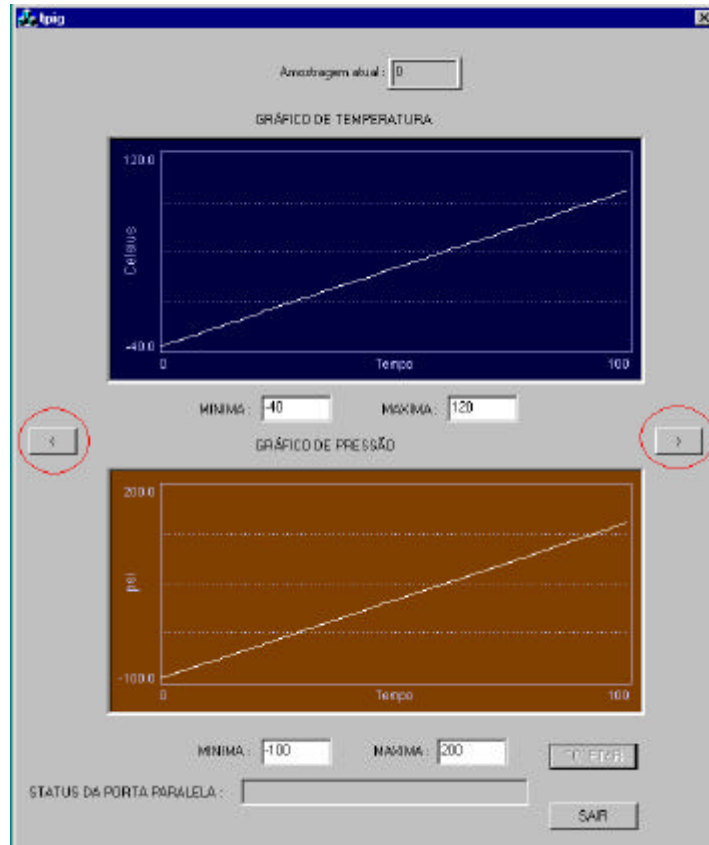


Figura 4.11 – Janela de Plotagem dos Resultados.

Ao término da leitura, um sinal sonoro é emitido e os dados são plotados na tela, como indica a Figura 4.11. O valor de 100 amostras para cada sensor é exibido e o botão ">" coloca o gráfico em modo de *Slide Show*, onde todos os dados são exibidos. Dependendo da quantidade de informação armazenada, esta operação pode demorar alguns minutos. O botão "<" permite retornar ao início do gráfico.

## Capítulo 5

# Testes e Caracterização do Sistema

---

Este capítulo descreve os procedimentos utilizados para testar e verificar a funcionalidade do sistema de aquisição de dados PIGTP e do circuito integrado DAQP02. Foi constatado que os resultados de algumas medições não correspondiam ao esperado, tornando necessária uma fase posterior de testes mais detalhados para diagnosticar as causas dessas discrepâncias. Foram realizados vários tipos de medição, cada um para verificar o funcionamento de uma parte do sistema.

Para a realização dos testes em bancada, foi usada a instrumentação disponível no LPC: uma fonte de tensão variável com controle de corrente, um gerador de sinais, um osciloscópio digital, um analisador lógico de 16 canais, um multímetro digital e alguns componentes eletrônicos auxiliares.

### 5.1 Consumo de corrente

---

A primeira medida que deve ser feita num teste do protótipo de um *chip* é verificar seu consumo de corrente. Para isso, simplesmente o circuito foi alimentado e verificou-se a

corrente total consumida, indicada pela fonte da bancada. A monitoração de corrente também foi feita para todos os modos de operação do circuito. Constatou-se então que o DAQP02 não consome mais que 500 uA, em qualquer modo de operação.

Além disso, constatou-se que a operação do circuito completo, com memória, não consome mais que 25 mA, qualquer que seja o modo de operação.

## **5.2 Conversão analógico-digital**

---

Num circuito de aquisição e digitalização de sinais, é extremamente importante que a conversão A/D seja feita corretamente, atendendo aos requisitos de precisão e linearidade. Visando verificar o funcionamento do conversor A/D, foi feito um teste de conversão de dados. Depois foi feito um teste visando verificar a linearidade.

### **5.2.1. Teste de conversão de dados**

Nesse teste, foram inseridos sinais analógicos nos conectores de entrada dos sensores S1 e S2. Com o analisador lógico, a saída digital foi monitorada nos pinos do barramento de dados D<7:0>. Assim, verificou-se um problema na conversão dos dados: a saída digital não correspondia aos valores aplicados nas entradas dos sensores. Na verdade, o resultado da conversão era sempre (00)h.

Para detectar o problema, foi necessário monitorar os sinais na saída do Deslocador de Nível e na saída do Circuito Amostrador. Por se tratar de um protótipo, há pinos que permitem o monitoramento de pontos chave no circuito, e este recurso permitiu a localização inicial do problema.

Um sinal senoidal com frequência de 100Hz e amplitude de aproximadamente 2 V foi aplicado na entrada de ambos os sensores, e o sinal na saída do Deslocador de Nível foi monitorado. A forma de onda resultante pode ser vista na Figura 5.1.

No canal 1, temos o sinal na saída do Deslocador de Nível e, no canal 2, o pulso que dá início à conversão. Para cada conversão, há uma leitura do sensor S1 e uma leitura do sensor S2. Pode-se ver que na ocorrência do pulso existe dado válido, pois nesse momento há

a passagem do sinal aplicado nas entradas dos sensores. Quando o pulso está ausente, a saída do deslocador de nível permanece em 1,3 V, que é a tensão de referência negativa medida.

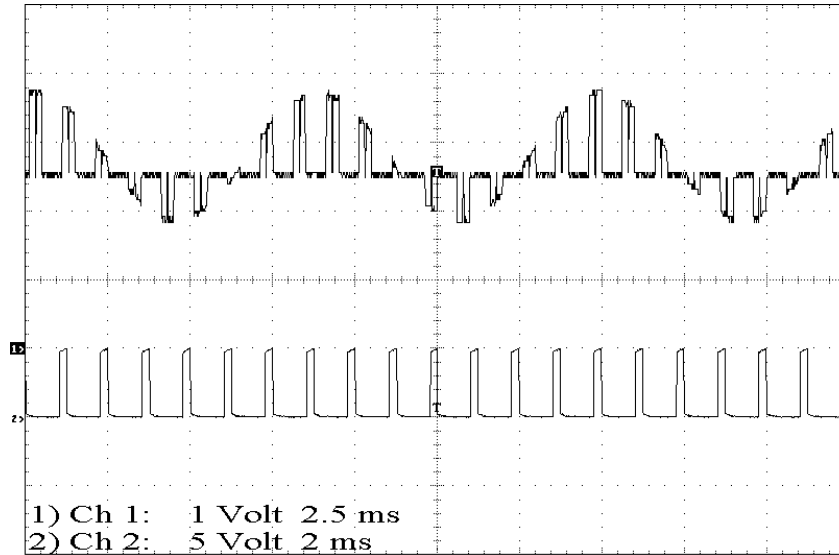


Figura 5.1 – Resultados do Teste de Conversão de Dados (1)

A tensão de referência encontra-se um pouco acima do valor teórico esperado, que é 1,26 V. Apesar disso, não se pode concluir que o Deslocador de Nível é o responsável pela ausência de sinal na entrada do A/D, pois está funcionando corretamente durante o período de aquisição do sinal.

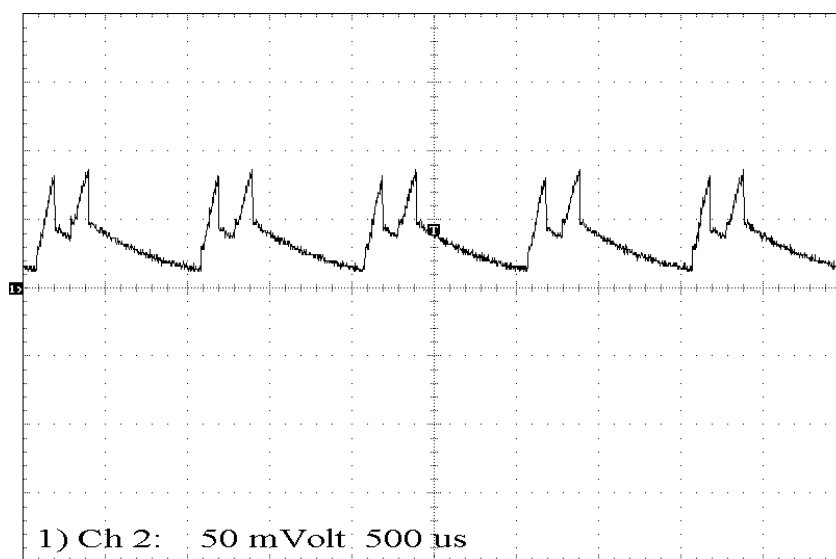


Figura 5.2 – Resultados do Teste da Conversão de Dados (2)



Após passar pelo Deslocador de Nível, o sinal entra no Circuito Amostrador, seguindo para o Conversor A/D. Ao ser monitorada, a saída do Circuito Amostrador apresentou o resultado visto na Figura 5.2. O sinal, que deveria ser senoidal, está completamente distorcido e é praticamente nulo, atingindo menos de 100 mV. Esta é uma situação que ficou para ser avaliada com mais cuidado posteriormente, e será vista no Capítulo 6.

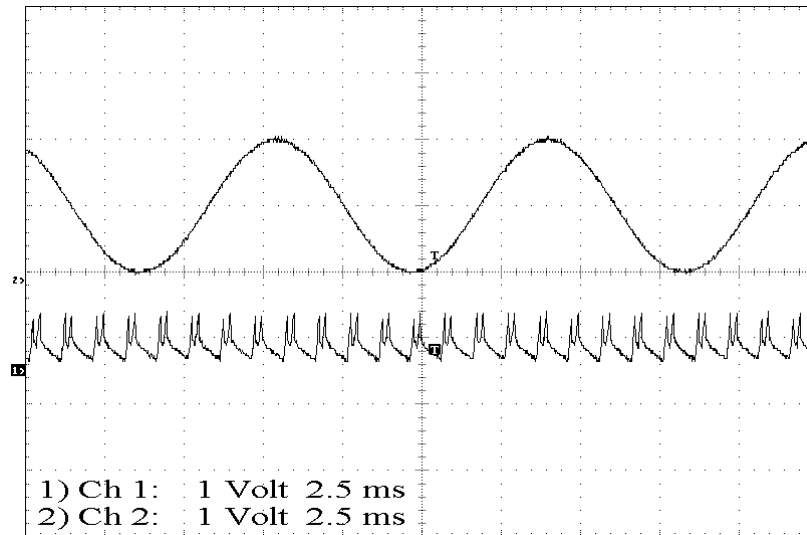


Figura 5.3 – Resultados do Teste de Conversão de Dados (3)

Na Figura 5.3 observa-se o sinal de entrada e o resultado na entrada do Conversor A/D. Com essas medidas não foi possível avaliar a funcionalidade do conversor A/D, pois o sinal foi atenuado pelo circuito analógico de entrada, antes de alcançá-lo. Então, para verificar o funcionamento do Conversor A/D, isolando este problema, foi feito outro procedimento.

Aplicou-se um sinal diretamente na entrada do Conversor A/D, e aí constatou-se seu correto funcionamento. Entretanto, a faixa de conversão medida foi de 3 V, com sinais de referência máximo e mínimo iguais a 4,30 e 1,31 V, respectivamente. O circuito fora projetado para sinais de referência de 3,78 e 1,26 V, o que forneceria uma faixa de 2,52 V, indicando assim um possível problema no Gerador de Referência. Essa é mais uma questão a ser avaliada, e será vista no Capítulo 6.

### 5.2.2. Teste de linearidade

Uma avaliação simples da linearidade é obter um gráfico de  $V_{in} \times V_{out}$ , sendo que  $V_{out}$ , na verdade, não é um valor de tensão, e sim um valor digital. Para obter-se esse gráfico, vários valores foram aplicados diretamente na entrada do Conversor A/D, procurando-se abranger toda a faixa dinâmica da conversão, e monitorou-se a saída com o analisador lógico.

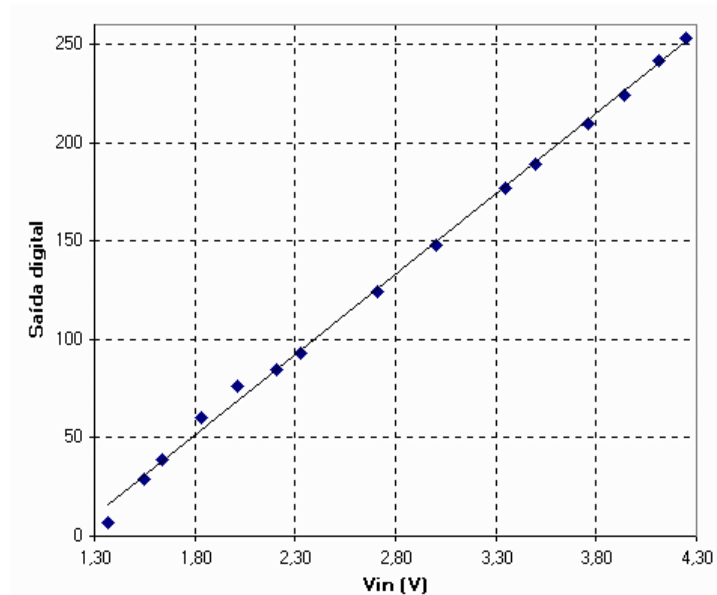


Figura 5.4 – Reta Aproximada Obtida com os Valores Medidos

Os valores medidos foram plotados num gráfico  $V_{in} \times V_{out}$ , como mostra a Figura 5.4. Fazendo-se uma aproximação pelo método dos mínimos quadrados, foi obtida uma reta que se mostrou condizente com os valores esperados para a conversão A/D. Pode-se ver que os pontos não se encontram muito afastados da reta, mostrando que a conversão está “bem linear”.

Uma forma quantitativa e mais minuciosa de se avaliar a linearidade de um conversor A/D é obter *INL* e *DNL* [5]. Estas duas grandezas são medidas em *LSB* (*Least Significant Bit*), a tensão equivalente ao bit menos significativo do A/D. Sendo  $n$  o número de bits de resolução do A/D, temos que:

$$LSB = (V_{ref_{MAX}} - V_{ref_{MIN}}) / 2^n$$

É fácil entender esse conceito ao observarmos a curva da função de transferência do A/D, na Figura 5.5, onde um LSB é a altura (ou largura) de um “degrau”.

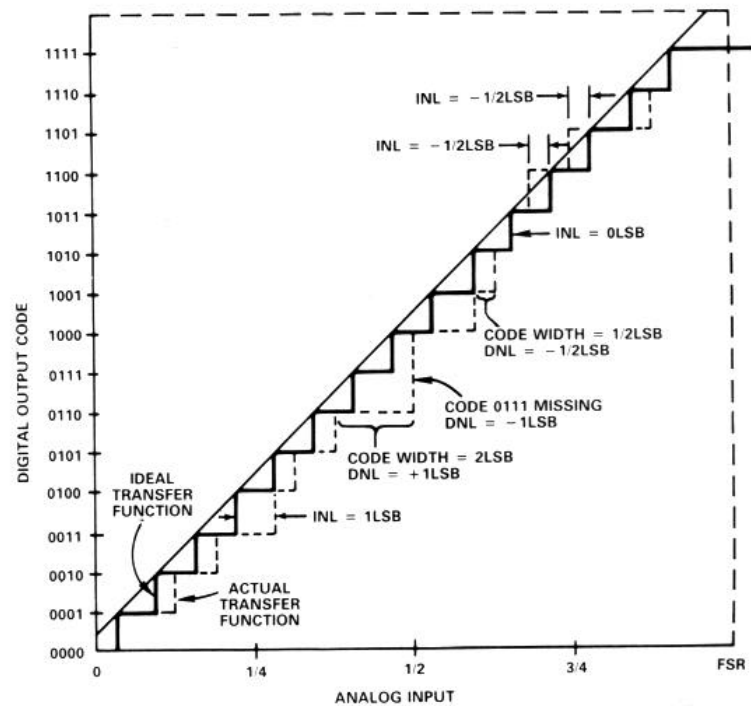


Figura 5.5 – Exemplos de INL e DNL para um conversor A/D de 4 bits [5]

A DNL (*Não Linearidade Diferencial*) é definida como a largura do código decrescida de um LSB, ou seja:

$$DNL = \Delta V_{in} - 1 \text{ LSB}$$

Se tivermos  $DNL < -1 \text{ LSB}$ , haverá um valor de saída faltante, ou seja, não há nenhuma entrada que faça o valor digital aparecer na saída. Devido aos espaçamentos dados entre uma medida e outra serem maiores que 1 LSB, foram obtidos valores médios para DNL, que foram plotados num gráfico, indicado na Figura 5.6. O resultado pode ser considerado satisfatório, pois não há nenhum valor de DNL com módulo maior que 1 LSB.

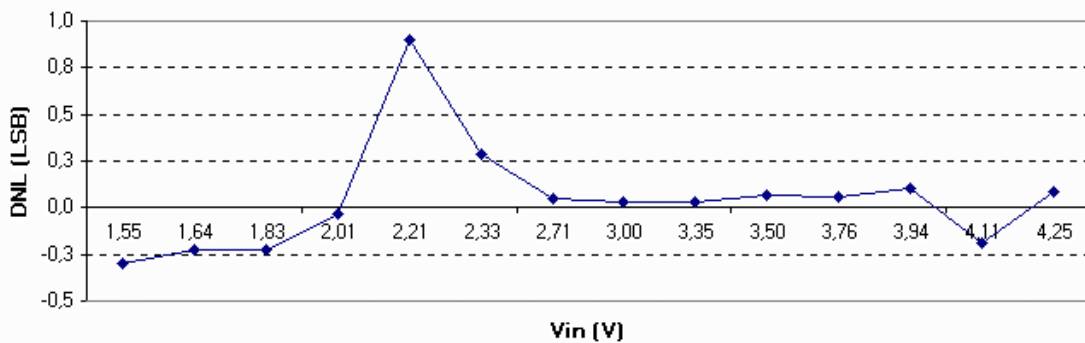


Figura 5.6 – Valores Obtidos de DNL

A INL (*Não Linearidade Integral*) é definida como a largura entre o valor de saída teórico (a “escada”) e o valor de saída experimental. Os valores obtidos de INL foram plotados, conforme a Figura 5.7, obtendo-se um resultado distante do ideal, principalmente para entradas entre 1,5 e 2,5 V.

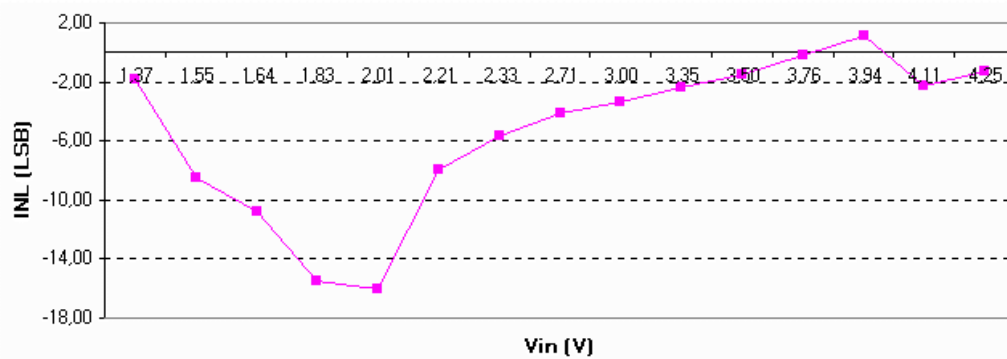


Figura 5.7 – Valores Obtidos de INL

O próximo passo aqui seria refazer essas medidas, com obtenção de mais pontos, que permitiriam a construção de gráficos mais detalhados, para confirmar esses resultados. Assim, os incrementos de tensão a serem feitos entre cada medida deveriam ser da ordem de alguns milivolts. Isso tornaria necessário o uso de uma instrumentação mais precisa, como um multímetro com mais dígitos de resolução. Não foi possível realizar essas medidas, pois não existiam tais equipamentos disponíveis.

### 5.3. Circuitos digitais

Os testes relacionados à parte digital do PIGTP foram feitos em duas etapas: testes do Controlador e a comunicação com o software.

#### 5.3.1. Testes do Controlador

Estes testes foram feitos configurando-se os sinais de controle e, com auxílio do analisador lógico, os sinais gerados pelo DAQP02 foram monitorados.

### 5.3.1.1. Modo de Inicialização

Durante o modo de Inicialização o processador escreve em toda a memória o valor (FF)h. Para testar se isto é feito corretamente, foram atribuídos valores a todos os pinos de configuração conforme a Tabela 5.1.

O pino ZBOOTENA ao receber sinal de nível lógico baixo, habilita o modo de Inicialização. O pino AQU em nível lógico alto leva o processador a iniciar um processo de aquisição de dados. Como os testes foram realizados tendo como base o pulso do odômetro, configurou-se o pino SVAR em nível lógico baixo. O DIV<2:0> em zero determina que não há divisão da frequência do pulso do odômetro. ADSTART, ST, SP e OFFSET são pinos de saída neste modo de operação. Os pinos DISTANCE e CLK\_INT ficam inicialmente em zero, pois o sinal de clock é fornecido pelo cristal. Os demais pinos não são necessários neste modo de operação e podem permanecer em nível lógico baixo.

Tabela 5.1 – Configuração para o Modo de Inicialização.

Pino	Descrição	Nível Lógico
ZBOOTENA	Habilita o modo de Inicialização – lógica invertida	0
AQU	Seleção entre Aquisição e Leitura	1
SVAR	Seleção de variável	0
DIV<2:0>	Divisão de frequência	0
ADSTART	Início de conversão de dados	--
ST	Seleção de temperatura	--
SP	Seleção de pressão	--
OFFSET	Aterramento das entradas	--
REQ	Pedido de acesso à memória	0
RC	Reinicialização da Leitura	0
PEXT	Seleção entre sinais de referência interno e externo	0
VREFNEXT	Sinal de referência externo – mínimo	0
VREFPEXT	Sinal de referência externo – máximo	0
HELP	Sinal que permite teste do conversor A/D	0
CLK_INT	Clock externo	0
RESET	Reinicialização do sistema	Reset
DISTANCE	Pulso do odômetro	0

Após a aquisição dos sinais com o analisador lógico, pôde-se ver os resultados obtidos nos testes realizados. Observa-se na Figura 5.8 o início do modo de Inicialização. O sinal CUP\_BOOT está em nível lógico alto, indicando que o circuito está em modo de Inicialização. O barramento de dados D <7:0>, imediatamente após a inicialização do sistema, recebe o valor (FF)h. Os sinais de controle da memória ZCE e ZWE, em nível lógico baixo, já que funcionam com lógica invertida, e ZOE, em nível lógico alto, possibilitam a escrita em memória.

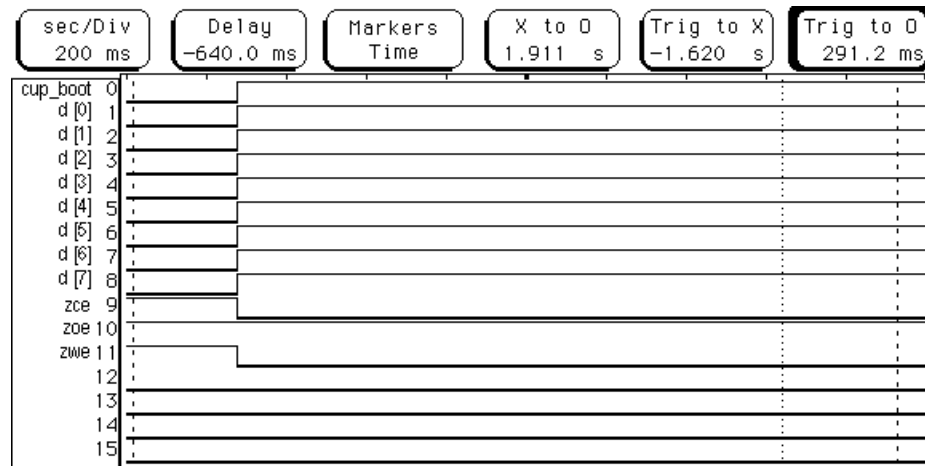


Figura 5.8 - Início do Modo de Inicialização.

A Figura 5.9 mostra o fim do modo de Inicialização, onde toda a memória foi preenchida com (FF)h. Somente o barramento de dados e os sinais de controle da memória podem ser vistos. Ao fim da operação, o sinal CUP\_BOOT e o barramento de dados retornam para nível lógico baixo. Os sinais de controle da memória retornam para nível lógico alto, desabilitando qualquer operação na mesma.

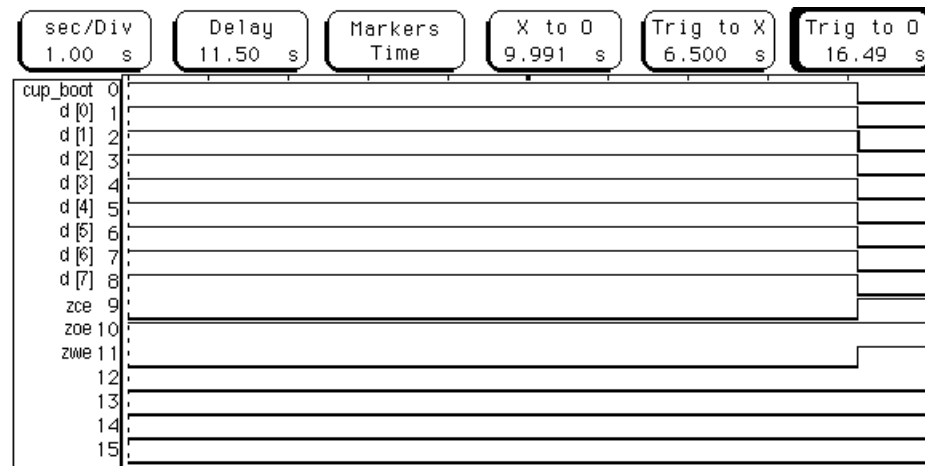


Figura 5.9 - Fim do Modo de Inicialização.

A geração dos endereços para escrita em memória pode ser vista na Figura 5.10. Somente 16 dos 22 pinos são exibidos já que não havia espaço suficiente para exibir todo o barramento em uma única janela.

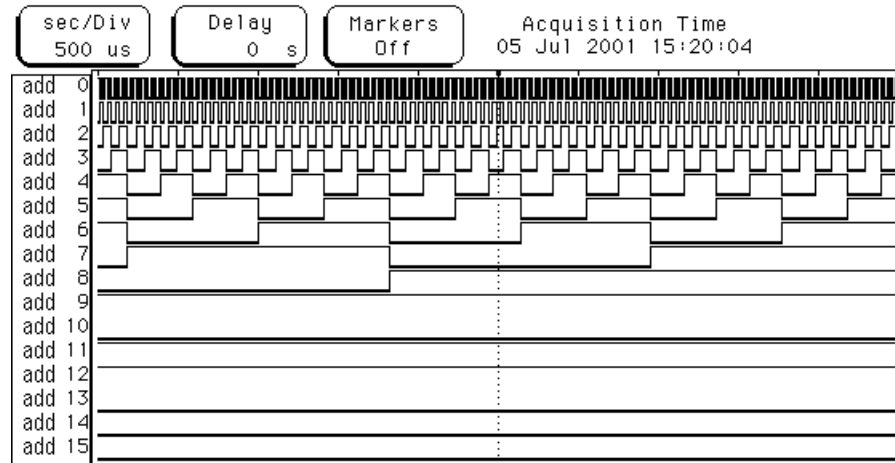


Figura 5.10 – Barramento de Endereços no Modo de Inicialização.

De acordo com o projeto, o período do *clock* seria de 12,6  $\mu\text{s}$ , e o período do bit ADD[0] (menos significativo) seria de 25,2  $\mu\text{s}$ . Na prática, constatou-se que o período do ADD[0] é de 25  $\mu\text{s}$ , com 10  $\mu\text{s}$  para nível lógico alto e 15  $\mu\text{s}$  para nível lógico baixo. Porém, esta discrepância não é considerada grave, pois não prejudica o funcionamento geral do sistema. O tempo total da inicialização com estes valores é de aproximadamente 52 segundos.

### 5.3.1.2. Modo de Aquisição

Durante o modo de Aquisição o processador deve ser capaz de iniciar a conversão dos dados e armazená-los em memória. A Tabela 5.2 traz a configuração necessária para isto.

Existem duas possibilidades de aquisição de dados: através do pulso do odômetro ou do próprio *clock* do circuito. Os testes foram feitos em função do pulso do odômetro. O DIV<2:0> recebe o valor 001, fazendo a frequência do pulso do odômetro ser dividida por dois. O sinal aplicado em DISTANCE simula o pulso do odômetro e consiste numa onda quadrada com frequência de 1kHz.

A frequência de 1 kHz é maior que a esperada, visto que o odômetro gera 10 pulsos por centímetro, segundo informação do CENPES. A velocidade do *Pig* não é constante, mas se considerarmos uma velocidade média de 3 m/s, teremos em torno de 300 Hz de frequência. Porém, a utilização de um valor maior facilita a visualização dos dados no analisador lógico. Conforme configuração do sinal DIV<2:0>, a frequência do odômetro é dividida por dois, levando a 500 Hz.

Tabela 5.2 – Configuração para o Modo de Aquisição.

Pino	Descrição	Nível Lógico
ZBOOTENA	Habilita o modo de Inicialização – lógica invertida	0
AQU	Seleção entre Aquisição e Leitura	1
SVAR	Seleção de variável	0
DIV<2:0>	Divisão de frequência	1
ADSTART	Início de conversão de dados	--
ST	Seleção de temperatura	--
SP	Seleção de pressão	--
OFFSET	Aterramento das entradas	--
REQ	Pedido de acesso à memória	0
RC	Reinicialização da Leitura	0
PEXT	Seleção entre sinais de referência interno e externo	0
VREFNEXT	Sinal de referência externo – mínimo	0
VREFPEXT	Sinal de referência externo – máximo	0
HELP	Sinal que permite teste do conversor A/D	0
CLK_INT	Clock externo	0
RESET	Reinicialização do sistema	Reset
DISTANCE	Pulso do odômetro	1 kHz

Após o término do modo de Inicialização, o DAQP02 entra automaticamente no modo de Aquisição e começa a conversão dos dados. As entradas dos sensores foram aterradas, já que não havia interesse em testar a conversão de dados nesse momento. Para cada pulso do odômetro, duas aquisições são realizadas: uma para SP e outra para ST.

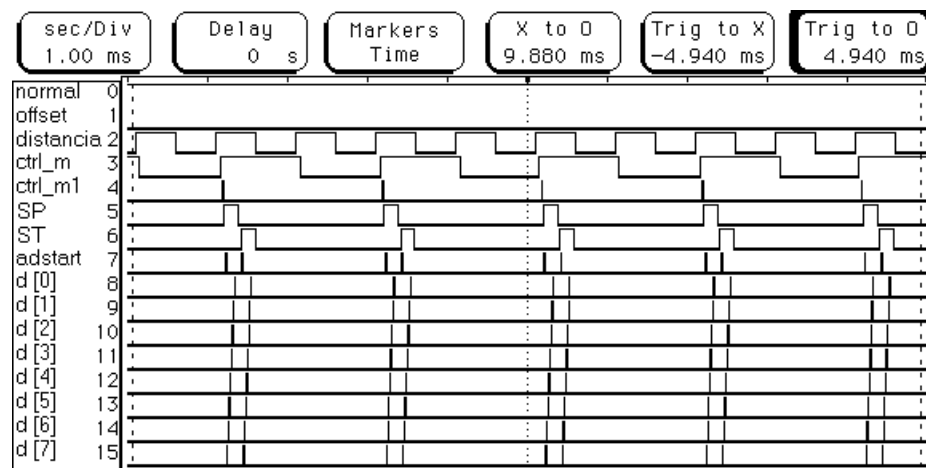


Figura 5.11 – Aquisição com entradas aterradas.

Como indicado na Figura 5.11, após cada pulso do odômetro, um pulso em CTRL\_M1 é gerado e a entrada do primeiro sensor (S1) é selecionada pelo pulso SP. Em seguida, o sinal ADSTART dá início à conversão dos dados de S1, como se vê nas linhas d[0] a d[7]. Logo após ocorre a mesma sequência para o pulso ST, que seleciona o segundo sensor (S2).



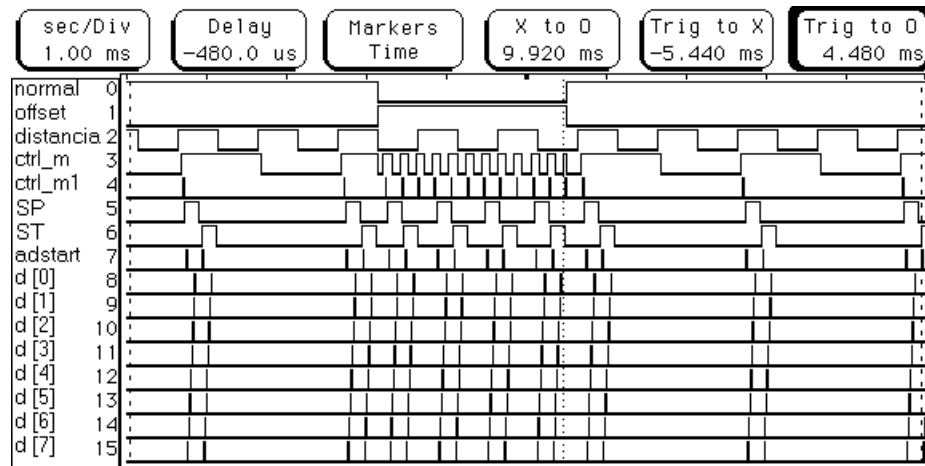


Figura 5.12 – Aquisição de ruído de fundo.

Na Figura 5.12, tem-se uma demonstração da aquisição do ruído de fundo. O sinal NORMAL vai para nível lógico baixo, enquanto o sinal OFFSET vai para nível lógico alto. Observando-se os pulsos CTRL\_M, CTRL\_M1 e ADSTART, vê-se que a aquisição passa a ocorrer em função do *clock*, e oito aquisições são realizadas. Embora os sinais SP e ST sejam gerados, eles não influenciam neste momento. NORMAL e OFFSET retornam ao estado normal de operação logo em seguida.

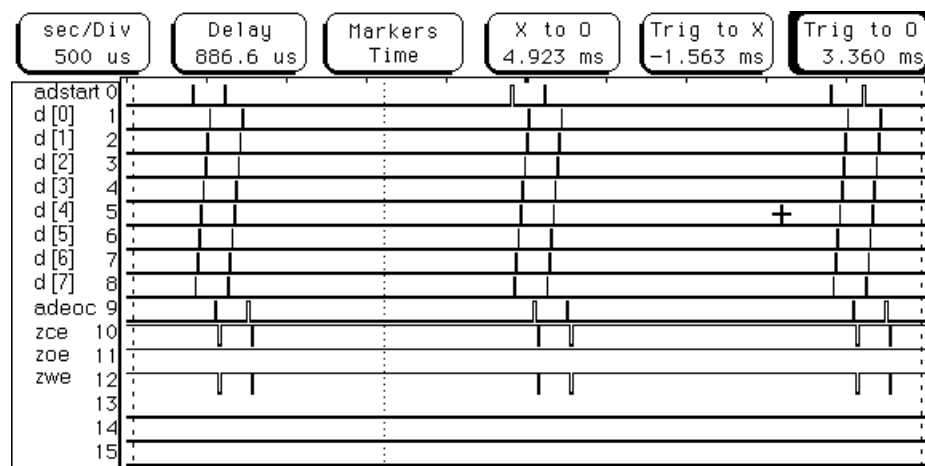


Figura 5.13 – Conversão e escrita na memória.

Nas Figuras 5.13 e 5.14, pode-se ver o sinal de início de conversão ADSTART e o de fim de conversão ADEOC sendo gerados. Ao fim de cada conversão, os sinais de controle de escrita na memória, ZWE e ZCE, recebem valores para escrita do dado em memória. A largura destes sinais é de um período de *clock* (25  $\mu$ s), suficiente para escrita na memória.

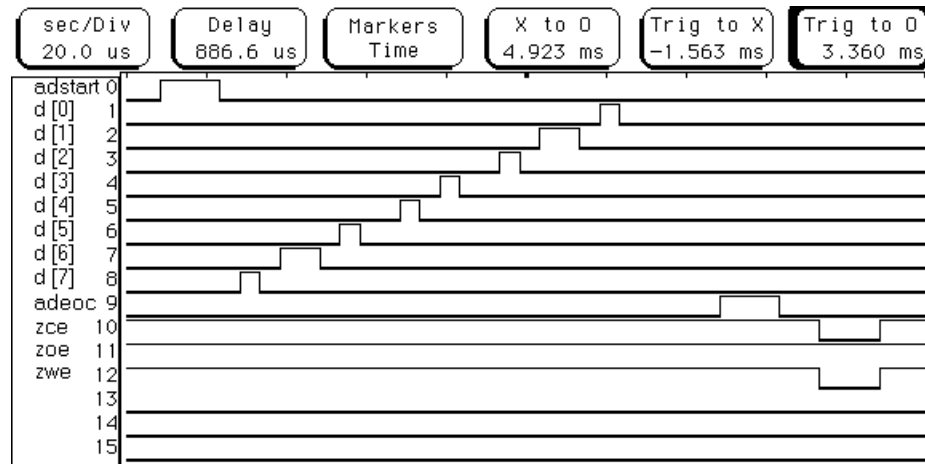


Figura 5.14 – Conversão e escrita na memória (ampliação).

Na Figura 5.15, são mostrados os oito primeiros bits do barramento de endereço sendo gerados e o controle da memória.

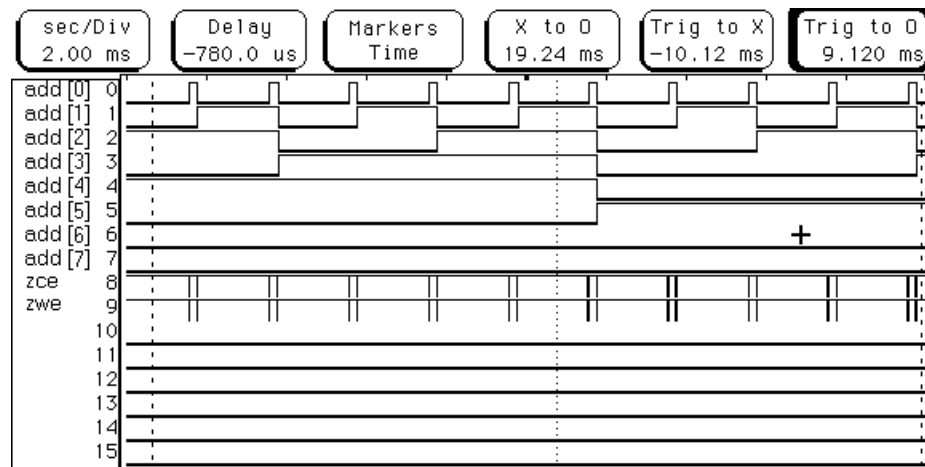


Figura 5.15 – Barramento de Endereços no Modo de Aquisição.

### 5.3.1.3. Modo de Leitura

Durante o modo de Leitura o processador deve ser capaz de se comunicar com a porta paralela de um PC e gerar os endereços necessários para leitura dos dados em memória. O sinal AQU agora recebe valor zero, conforme a Tabela 5.3, indicando que o processador irá operar em modo de leitura.

Tabela 5.3 – Configuração para o Modo de Leitura.

Pino	Descrição	Nível Lógico
ZBOOTENA	Habilita o modo de Inicialização – lógica invertida	0
AQU	Seleção entre Aquisição e Leitura	0
SVAR	Seleção de variável	0
DIV<2:0>	Divisão de frequência	0
ADSTART	Início de conversão de dados	--
ST	Seleção de temperatura	--
SP	Seleção de pressão	--
OFFSET	Aterramento das entradas	--
REQ	Pedido de acesso à memória	2 kHz
RC	Reinicialização da Leitura	0
PEXT	Seleção entre sinais de referência interno e externo	0
VREFNEXT	Sinal de referência externo – mínimo	0
VREFPEXT	Sinal de referência externo – máximo	0
HELP	Sinal que permite teste do conversor A/D	0
CLK_INT	Clock externo	0
RESET	Reinicialização do sistema	Reset
DISTANCE	Pulso do odômetro	0

Uma onda quadrada com frequência de 2 kHz foi aplicada no sinal REQ do protocolo de comunicação, responsável pelo pedido de leitura da memória. Para cada pulso do REQ, um novo endereço e um pulso em ACK são gerados. O resultado pode ser visto na Figura 5.16. Na Figura 5.17, tem-se o fim de operação do Modo de Leitura. Ao ser aplicado um pulso em RC, verifica-se o reinício de todo o processo, conforme esperado.

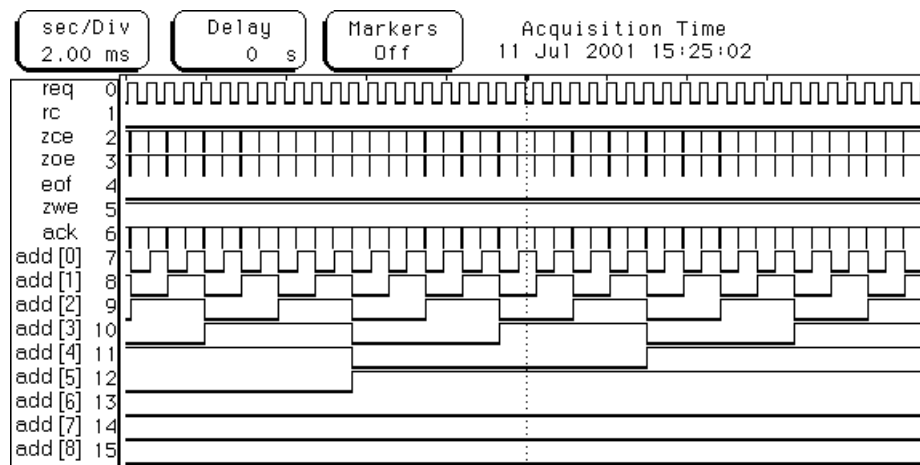


Figura 5.16 – Modo de Leitura.

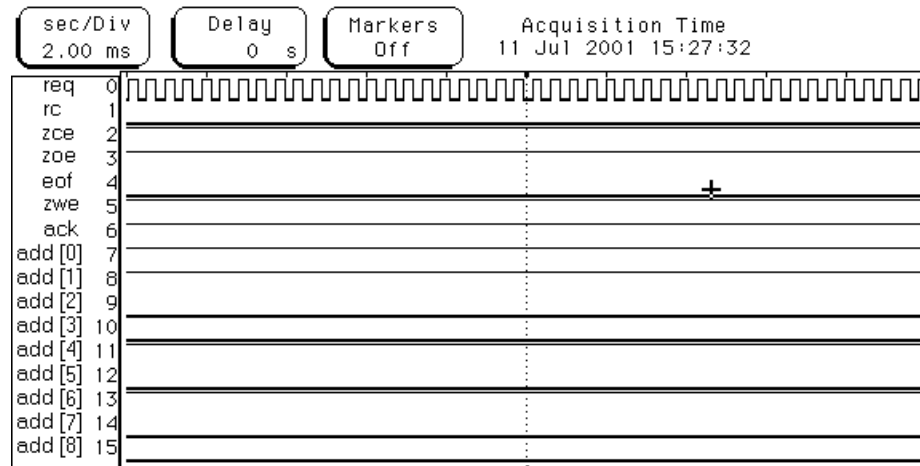


Figura 5.17 – Fim do Modo de Leitura.

### 5.3.2. Comunicação com o software

Para testes da leitura e escrita em memória foi feito o seguinte procedimento: primeiro iniciou-se o circuito e foram gerados alguns endereços; em seguida, leu-se o conteúdo da memória, monitorando-se o barramento de dados com o analisador lógico, conforme indicado pelas Figuras 5.18 e 5.19.

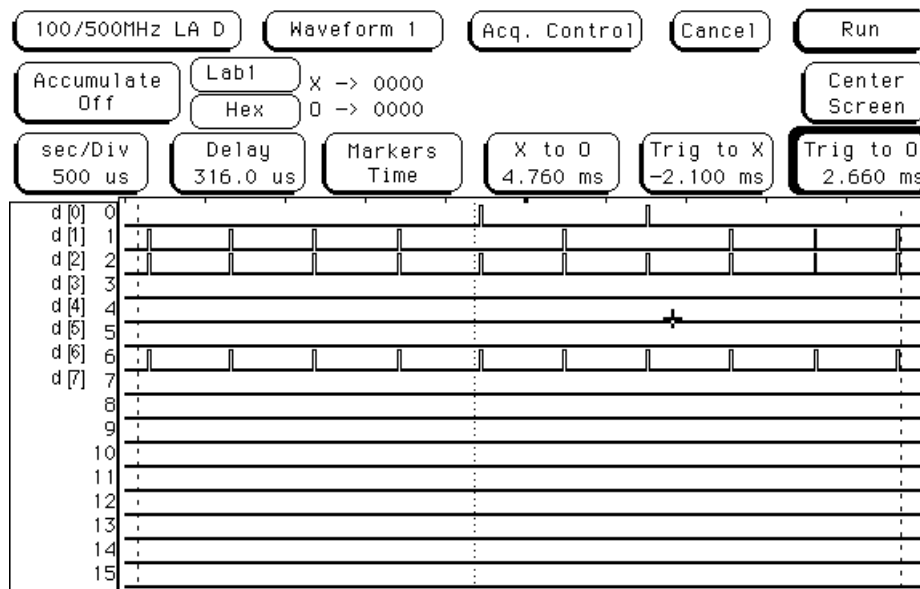


Figura 5.18 - Leitura em memória

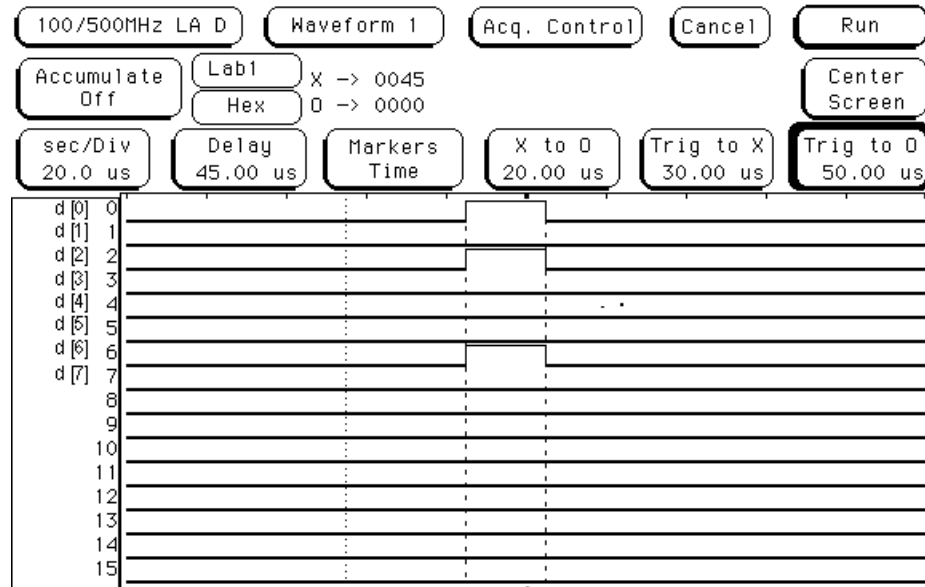


Figura 5.19 - Leitura em memória (ampliação)

Os testes com o programa desenvolvido para leitura e armazenamento em arquivo dos dados armazenados em memória foram realizados com sucesso. Os testes nesta etapa atestaram a funcionalidade da parte do circuito responsável pela leitura dos dados por computador.

## Capítulo 6

# Diagnóstico e Correção de Falhas

---

Quando, através dos testes, é verificado que um equipamento encontra-se fora das especificações, deve-se tentar descobrir o motivo. Quando são encontradas falhas no protótipo, pode-se optar por corrigir tais falhas ou partir para o desenvolvimento de uma nova versão.

A partir dos testes realizados com o sistema de aquisição de dados PIGTP e o CI DAQP02, foram observados alguns problemas no seu funcionamento. Os resultados mais críticos foram indicados pelos testes de conversão de dados, na Seção 5.2.1, levando a crer que a parte analógica do circuito possuía algum defeito grave.

A seguir são descritos os métodos usados para se diagnosticar tais problemas, bem como as soluções encontradas para corrigí-los. Mas antes, deve-se registrar que, com informações obtidas junto aos projetistas do CI, chegou-se à suspeita de haver uma falha nas conexões internas do circuito.

Aqui, torna-se importante um breve esclarecimento sobre as fases de projeto de um circuito integrado. Dentre as várias fases existentes, temos o projeto do esquemático, onde é feita a simulação elétrica dos componentes, sem se considerar como será sua implementação física dentro do circuito. Uma outra fase é o projeto do *layout*, onde os componentes são

dimensionados fisicamente e posicionados no *chip*. É extremamente importante que estes projetos sejam funcionais e, além disso, coerentes entre si, para que o circuito real funcione como esperado.

A falha que suspeitou existir teria sido provocada por um possível *bug* na ferramenta de projeto, e ocorrido durante o projeto do *layout*. Esta falha seria um deslocamento em algumas máscaras, o que estaria interferindo na alimentação de algumas partes do circuito, e prejudicando assim o funcionamento da parte analógica do CI.

## 6.1. Microscopia

---

A análise de CI's através de microscopia pode ser útil para se achar falhas, notadamente quando há conexões erradas, normalmente provenientes de alguma falha no processo de fabricação, o que é raro ocorrer. No caso, como existe uma suspeita de falha de conexão, usou-se essa técnica na tentativa de se identificar a causa dos problemas observados.

Para isso, foi usado um microscópio óptico Leica DMR, pertencente ao Laboratório de Metalografia do Programa de Engenharia Metalúrgica e de Materiais (PEMM) da COPPE/UFRJ. Com auxílio de um computador acoplado ao microscópio e um software que permite a digitalização e aquisição das imagens observadas, foi possível registrar algumas imagens do layout do CI.

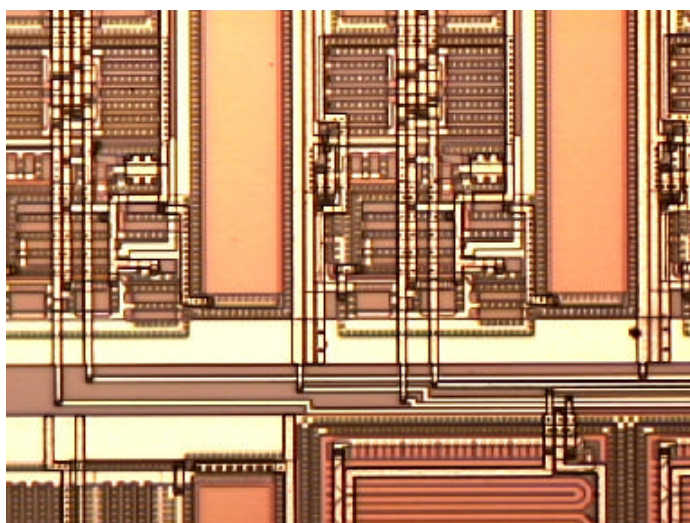


Figura 6.1 – Fotografia do CI DAQP02 (1)

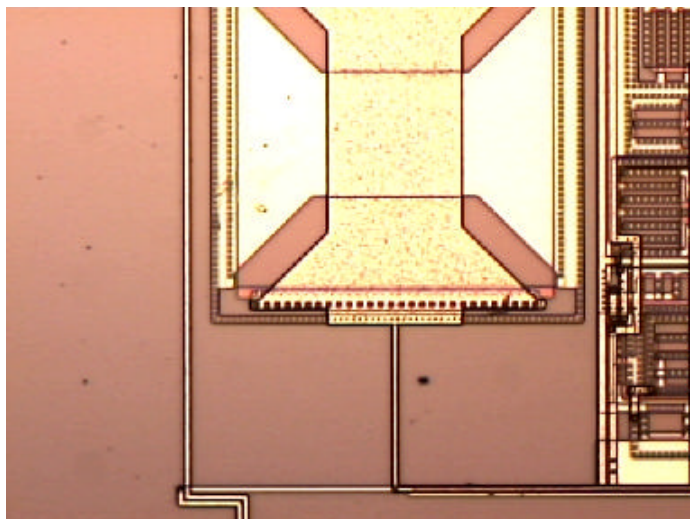


Figura 6.2 – Fotografia do CI DAQP02 (2)

Primeiramente, numa visão geral, com ampliação (*zoom*) de 25 vezes no microscópio, o circuito parecia condizente com o *layout* projetado, indicado na Figura 3.11. Mas havia possibilidade de haver falhas microscópicas no circuito e uma análise mais minuciosa foi feita.

Com o aumento da ampliação para 100 vezes, o circuito foi fotografado por partes. Nas Figuras 6.1 e 6.2, podemos ver as camadas de materiais diversos (metal, poli-silício, difusões, etc.), formando os componentes, contatos e trilhas. Várias fotografias como essas foram obtidas para a análise do circuito.

Mesmo assim, neste caso não se chegou a encontrar o defeito responsável pelos problemas observados. Porém, isso não invalida essa técnica, que pode vir a ser proveitosa em muitos casos, normalmente quando a falha é bem visível ao microscópio. Um bom exemplo é se houver degeneração de toda uma região do substrato do CI.

## 6.2. Verificação de esquemático e *layout*

---

Para descobrir se houve realmente algum erro durante a fase de projeto, foi feita uma revisão nos arquivos da ferramenta de *software* onde o circuito foi desenvolvido. A Figura 6.3 mostra um dos esquemáticos que foram analisados, onde se podem observar o Circuito Amostrador (bloco *sah*), os blocos internos do Condicionador de Sinais (*Mux3*, *Levelshift3*, *Genref3*, *Mux2x3*) e suas respectivas interligações. Porém, ao conferir esses esquemáticos, não se encontrou nenhuma anormalidade.



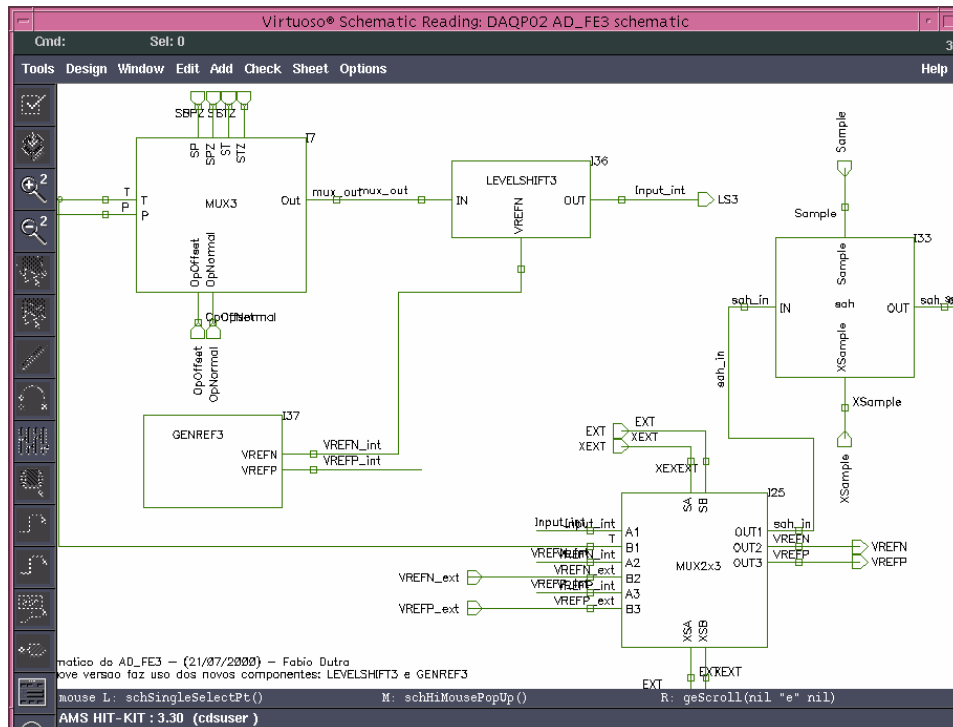


Figura 6.3 - Verificação do Esquemático

O próximo passo seria refazer algumas simulações desses esquemáticos, o que não foi possível, pois o *software* não estava disponível no momento.

Outro passo importante seria realizar a checagem do *layout* com o esquemático. A ferramenta que compara esquemático e *layout* chama-se *LVS (Layout Versus Schematic)*. De acordo com as informações fornecidas pelos projetistas, esta ferramenta foi utilizada durante o projeto e não houve problema. Porém, dessa vez o *layout* a ser usado para a comparação seria extraído do arquivo em formato *GDS2*, que é o formato final enviado para fabricação do CI. Como havia uma suspeita de que um *bug* na ferramenta de projeto tivesse alterado esse arquivo antes de seu envio para fabricação, poder-se-ia encontrar algum erro no projeto de *layout* por esse procedimento. Porém, isto também não foi possível pela indisponibilidade do *software*.

Essas duas últimas etapas seriam essenciais para uma identificação precisa das falhas que já se suspeitavam existir. Entretanto, a licença de uso para o pacote de *software* de posse do LPC/UFRJ estava vencida, permanecendo nessa situação até o fechamento deste trabalho. Isto impossibilitou a realização desses testes, pois, com a licença vencida, funcionalidades essenciais da ferramenta ficaram desabilitadas. Assim, fica aqui deixada em aberto uma possibilidade de trabalho a ser feito numa futura oportunidade, para complementação desta

verificação. Para isso será necessário acessar o arquivo *GDS2* que se encontra disponível no LPC.

### **6.3. Testes em bancada**

---

Os testes relatados no capítulo anterior revelaram alguns problemas no funcionamento do circuito. Esses problemas poderiam ser provenientes do circuito externo ao DAQP02. Muitas vezes encontramos componentes discretos defeituosos ou erros no projeto ou confecção da placa de circuito impresso. Também é muito comum se encontrar problemas de mau contato em encaixes, conectores, etc.

Uma maneira de verificar se há falhas num circuito discreto é conferir a continuidade das trilhas da placa. Isso pode ser feito com um multímetro com sinal sonoro de curto-circuito. Usando-se esta técnica, verificou-se que algumas trilhas não estavam corretas, fazendo com que a alimentação não chegasse aos CI 74LS00 (portas NAND).

Também foi verificado que alguns pinos do DAQP02 aonde deveria chegar a alimentação não estavam conectados. Tudo isso fazia com que o circuito não funcionasse corretamente.

Outro problema a ser verificado, relatado na Seção 5.2.1, era a atenuação do sinal pelos circuitos analógicos do DAQP02. Visando localizar o defeito, foi feito o seguinte procedimento: aterrou-se a entrada de um dos sensores, um sinal foi injetado na outra entrada, e o pino de entrada do Conversor A/D foi monitorado com o osciloscópio. Fez-se o mesmo procedimento, trocando-se a entrada que estava aterrada e a entrada com sinal. Com isso, verificou-se que uma das entradas era muito mais atenuada que a outra, indicando a existência de uma assimetria nos circuitos analógicos de entrada do CI.

## 6.4. Soluções adotadas

---

### 6.4.1. Correção de trilhas nas placas

Conforme observado em testes de bancada, na Seção 6.3, havia falhas em algumas trilhas das placas de circuito impresso. As seguintes correções foram feitas:

- Ligação de alguns pinos do DAQP02, que estavam indevidamente flutuando, ao Vdd;
- Aterramento do pino AQU, para corrigir configuração da frequência do odômetro na aquisição;
- Ligação dos pinos de alimentação do CI 74LS00, que estavam indevidamente flutuando.

### 6.4.2. Multiplexador analógico

Os sinais de entrada apresentavam grande distorção de valores na entrada do Conversor A/D, conforme descrito na Seção 5.2.1. Assim, decidiu-se não se utilizar a parte de entrada do CI, até que fossem esclarecidas as causas do problema. Então, os sinais dos sensores S1 e S2 deveriam ser colocados diretamente na entrada do conversor A/D (pino ADIN do DAQP02).

Assim, as entradas originais destinadas aos sensores foram aterradas, sendo inutilizadas, e os sinais passaram a ser inseridos na entrada do A/D. Apenas um dos pulsos de aquisição gerados pelo DAQP02, o pulso SP, passou a ser usado para aquisição dos sinais de S1 e S2. O pulso ST, que fazia a segunda aquisição na ocorrência de cada pulso de odômetro, não foi mais utilizado, pois era em sua aquisição que ocorriam os problemas de atenuação do sinal, conforme descrito na Seção 6.3.

Era necessário então alternar a aquisição dos dois sinais a cada ocorrência de SP. Isso seria possível com o uso de um *Multiplexador Analógico*. Este circuito conduz apenas um de vários sinais analógicos de entrada, de acordo com um sinal de controle. No nosso caso, um mux de dois canais seria suficiente para solucionar o problema, ou seja, o circuito deveria

apenas alternar a passagem de dois sinais. Para construção do mux, foi projetado o circuito ilustrado na Figura 6.4.

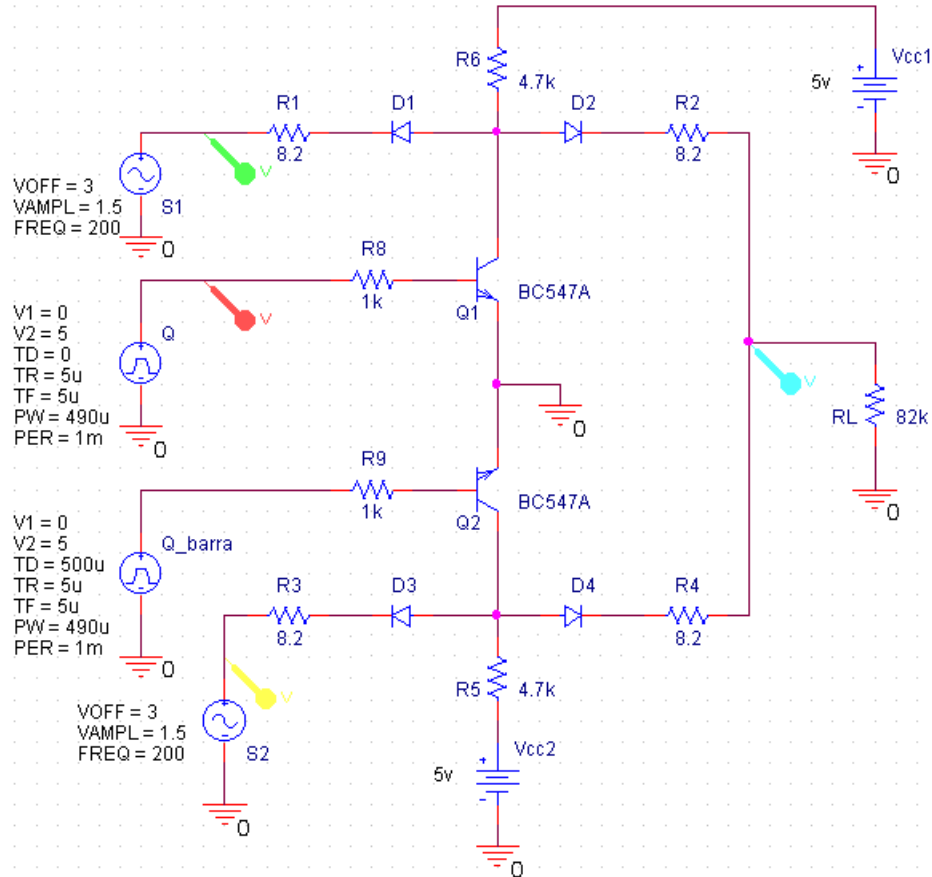


Figura 6.4 – Multiplexador Analógico

Os manuais dos componentes usados no circuito encontram-se no Anexo II. O seu funcionamento se baseia em alternar cada par de diodos entre os estados de condução e de corte, pelo chaveamento dos transistores comandado pelos sinais Q e Q', que podem ser as saídas de um flip-flop. Temos então dois estados, que se alternam de acordo com Q e Q':

- Q = '1' => D1 e D2 estão em condução, D3 e D4 estão em corte =>  $V_{out} = S1$
- Q = '0' => D3 e D4 estão em condução, D1 e D2 estão em corte =>  $V_{out} = S2$

Pode-se conferir seu funcionamento, ao simular-se S1 e S2 com sinais senoidais de 200 Hz e Q e Q' com duas ondas quadradas de 1 kHz. Na Figura 6.5 é apresentado o resultado desta simulação.

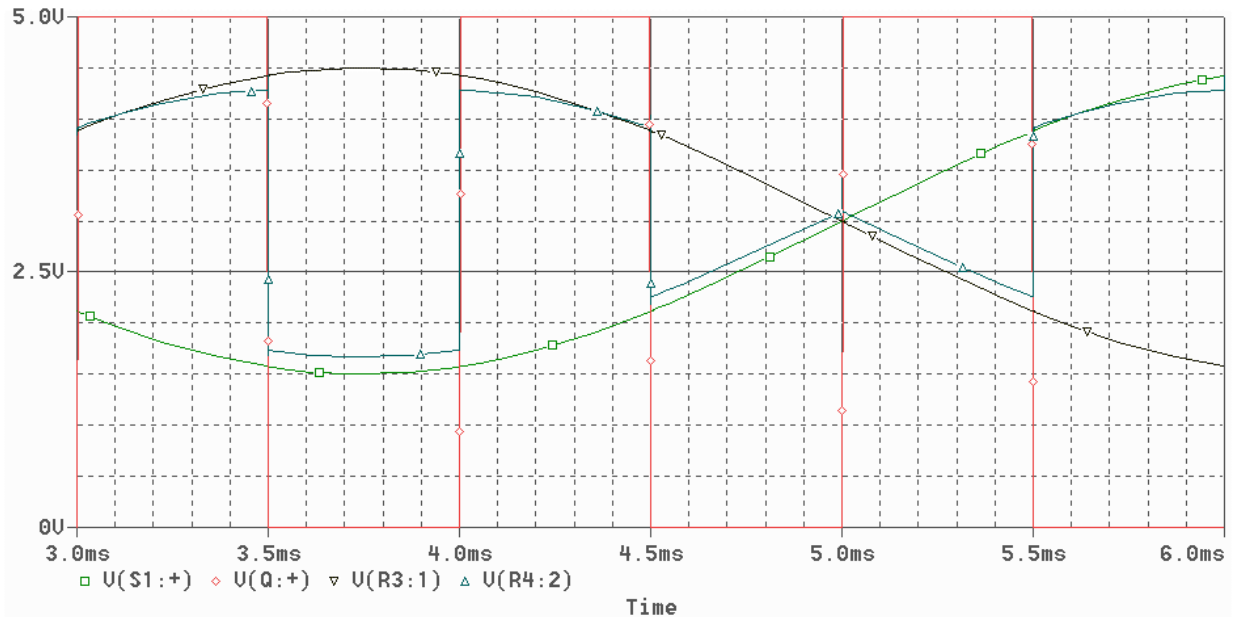


Figura 6.5 – Simulação do Multiplexador Analógico

O circuito foi então montado, usando-se um CI 7474 (flip-flop tipo D) com a entrada de *clock* acionada pelo pulso SP, para gerar os sinais Q e Q'. Usou-se uma placa padrão, de tamanho bem próximo ao das placas do PIGTP. Assim, ao ser acoplado ao sistema, o circuito adicional interferiria o mínimo possível no volume total.

### 6.4.3. Buffer

Ao conectar o Multiplexador ao circuito do PIGTP, o resultado observado não foi o mesmo obtido na simulação. Havia uma distorção do sinal, provavelmente provocada pela saída do Circuito Amostrador, que ainda se encontrava conectado à entrada do A/D e estaria injetando alguma corrente neste nó. Tornou-se necessário então aumentar a impedância vista pela saída do Mux, e injetar corrente suficiente no pino ADIN para estabilizar o sinal.

Assim, foi montado, com um amplificador operacional, um *buffer* de saída para o Multiplexador, no espaço ainda disponível na placa-padrão. O CI TLV2372 da Texas Instruments, cujas especificações encontram-se no Anexo II, foi escolhido para implementação do *buffer*. Sua escolha foi baseada nas seguintes especificações:

- **Alimentação:**  $2,7 < V_s < 15 \text{ V}$  – Facilita a conexão com o mux, já configurado com alimentação em nível TTL (5 V) por causa do CI 7474;

- **Corrente de polarização:**  $I_q = 0,66 \text{ mA}$  – É um baixo consumo adicional;
- **Slew Rate:**  $SR = 2,4 \text{ V}/\text{is}$  – O *slew rate* deve ser suficiente para responder à transição entre os sinais dos sensores, ao ocorrer o chaveamento. No pior caso, um sinal estaria em 0 V e o outro em 5 V. Com o *slew rate* dado, o tempo de resposta à uma transição seria  $t = 5 / 2,4 = 2,1 \text{ is}$ , valor satisfatório para esta aplicação;
- **Excursão do sinal: rail-to-rail input/output** – Permite uma excursão do sinal até os níveis de alimentação (0 e 5 V), tanto na entrada quanto na saída;
- **Corrente de saída:  $I_o \text{ (max)} = 100\text{mA}$**  – Fornece corrente suficiente para compensar a injeção de carga na entrada do conversor A/D.
- **Impedância de entrada:  $R_{in} = 1000 \text{ G}\Omega$**  – Tende a infinito, que seria o caso ideal, já que o objetivo é isolar a saída do mux e a entrada do conversor A/D;

No entanto, com uma impedância de entrada infinita, o amplificador operacional faz com que não circule corrente através dos diodos, atenuando o sinal de saída. Assim, foi colocado em paralelo com a entrada do *buffer* um resistor de 100 k $\Omega$  para diminuir a impedância de saída do Multiplexador e haver circulação de corrente. O circuito do Multiplexador com o *buffer* de saída está ilustrado na Figura 6.6.

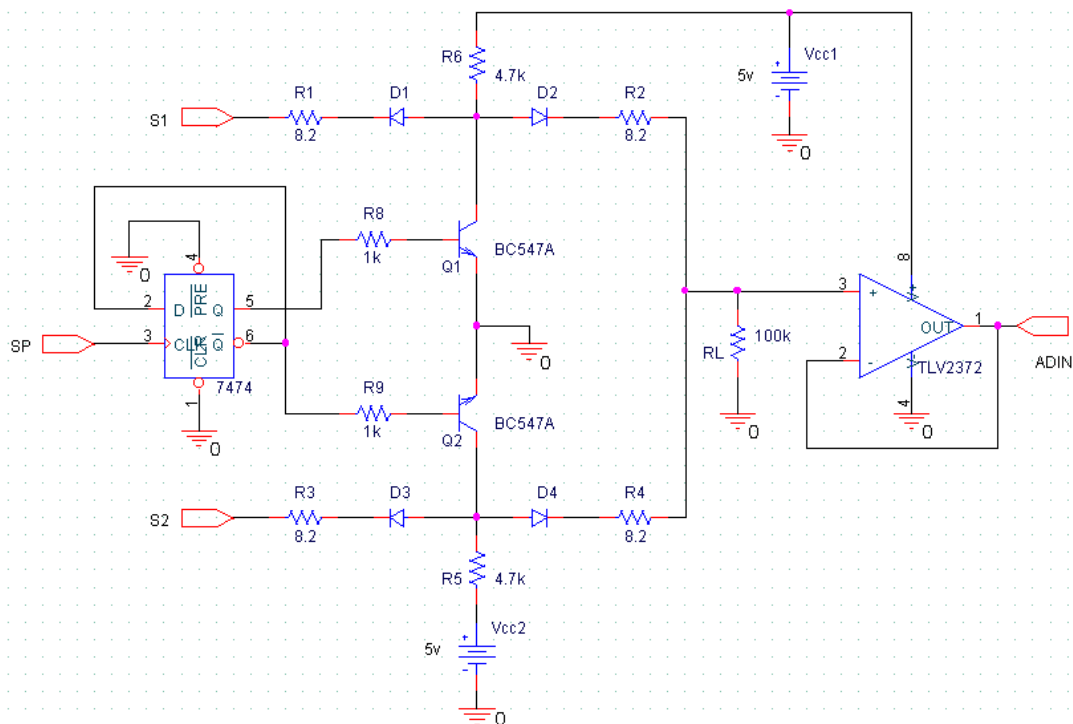


Figura 6.6 – Multiplexador Analógico com Buffer de Saída

#### 6.4.4. Escrita na memória

Apenas um dos pulsos de aquisição passou a ser usado. No entanto, os dois pulsos SP e ST continuam a ser gerados internamente pelo DAQP02 e, ao escrever na memória, o processador passou a alternar sucessivamente a escrita de um valor de leitura válido, na ocorrência o pulso SP, com a escrita de um valor não determinado, na ocorrência do pulso ST. Conseqüentemente, apenas metade da memória seria preenchida com valores válidos (1 Mbyte).

Para que todo o conteúdo da memória continuasse a ser aproveitado seria necessário que a escrita na memória só fosse realizada na ocorrência do pulso SP. Isso é possível com a colocação de uma lógica de interface entre o DAQP02 e a memória. Uma porta OR é suficiente para fazer esse papel, conforme mostrado na Figura 6.7.

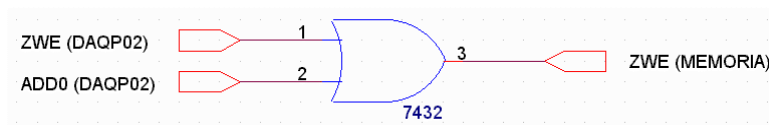


Figura 6.7 – Circuito de Escrita na Memória

Esse circuito faz com que a escrita só ocorra quando o endereço gerado pelo processador for par ( $ADD[0] = '0'$ ), correspondente à ocorrência do pulso SP. Porém, o circuito só deve agir durante o modo de aquisição. Ele deve ficar desabilitado no modo de inicialização, para que toda a memória seja preenchida com (FF)h. Seria necessária então a alteração do circuito já existente na interface entre a memória e o processador (ver componente U4 da seção 4.2.2).

O circuito de escrita na memória não chegou a ser implementado, pois seria necessária a adição de, pelo menos, mais um CI à lógica de interface memória/processador. À medida que cada circuito auxiliar para correção das falhas foi adicionado, houve um aumento considerável de volume e de consumo totais do sistema de aquisição de dados. Portanto, o sistema começou a divergir dos requisitos iniciais de baixo consumo e volume. Previu-se que a partir daí o sistema passaria a não mais atender os requisitos citados. Além disso, seria necessária uma mudança no software de leitura dos dados pelo computador, para corrigir a distância definida entre as aquisições, já que passaria a existir apenas um dado gravado na memória decorrente de cada pulso do odômetro.

## Capítulo 7

### Conclusão e Trabalhos Futuros

---

Neste capítulo é apresentado um exemplo de uma alternativa para implementação do sistema de aquisição de dados, baseado em um microcontrolador comercial. Finalmente, são relatados os objetivos e resultados alcançados no trabalho. Vejamos primeiro como poderia ser realizada uma outra forma de implementação do sistema.

Como foi descrito no Capítulo 2, existem no mercado diversos dispositivos que podem ser utilizados para implementar um sistema de aquisição de dados. Porém, quando o DAQP02 começou a ser desenvolvido, um sistema dedicado apresentava muitas vantagens em relação aos dispositivos existentes no mercado, e permitia atender facilmente às necessidades de baixo consumo, alta capacidade de memória, e miniaturização do sistema.

Porém, a evolução dos dispositivos comerciais nos últimos anos mudou essa situação. Uma mudança essencial para isso foi a melhora das tecnologias de fabricação de circuitos integrados, que permitiu reduzir seu consumo de potência, tamanho e custo. Com a diminuição da largura do canal dos transistores, hoje em 0.13  $\mu\text{m}$ , a integração ficou ainda maior, tornando o DAQP02, feito em tecnologia de 0.8  $\mu\text{m}$ , obsoleto.

Na verdade, essa evolução foi fortemente impulsionada pelo advento da telefonia celular nos últimos anos, o que incentivou o investimento da indústria de semicondutores no



desenvolvimento de CI's de baixo consumo e de novas arquiteturas, mais velozes e eficientes, para os processadores digitais em geral.

E finalmente, uma mudança decisiva foi a possibilidade cada vez maior de uso de componentes em tecnologia SMD (*Surface Mounted Devices*). Isso também permitiu uma considerável redução no tamanho dos circuitos eletrônicos.

Entre os dispositivos disponíveis no mercado, temos os microcontroladores (̂C), os processadores digitais de sinais (DSP) e os dispositivos de lógica programável (CPLD/FPGA). É sustentável dizer que, pelas características de preço, disponibilidade de ferramentas para desenvolvimento, e recursos disponíveis em sua arquitetura, os microcontroladores são uma boa opção para realização de um sistema de aquisição de dados para um *Pig* Instrumentado. O seu uso também permite a redução do tempo necessário entre prototipagem e produto final, devido à facilidade de correção e readaptação do projeto.

Não é nova a idéia de se realizar essa tarefa fazendo uso de microcontroladores. Existem referências de outras implementações funcionais realizadas [4], atendendo aos requisitos de volume para funcionar em dutos de pequeno diâmetro, além de baixo consumo e alta capacidade de memória.

Existem hoje microcontroladores de diversos fabricantes, como *Microchip*, *Texas Instruments*, *National*, *Intel*, *Philips* e outros. Num sistema de aquisição de dados poderíamos usar dispositivos da família PIC da *Microchip*, pois alguns modelos apresentam características bem úteis para o projeto.

Há modelos que possuem vários canais de conversores A/D, que podem ser multiplexados [7, 8]. Há também a possibilidade de conexão com uma memória para armazenamento dos dados. Alguns modelos possuem conversores A/D de 10 bits ou mais, oferecendo uma grande precisão, além de atenderem bem aos requisitos de consumo. As ferramentas de desenvolvimento de *software* para PIC são largamente usadas, permitindo programação em linguagens *C* ou *Assembly*.

Um diagrama de blocos para implementação do sistema com PIC é apresentado na Figura 7.1. O conversor A/D, o circuito de amostragem e o gerador de referência estão integrados no *chip*. Externamente são necessários a memória e os condicionadores. Ao fazer um projeto tirando-se o máximo proveito da tecnologia SMD, é possível desenvolver um produto que tenha volume tão reduzido quanto uma implementação com CI dedicado.

Uma possibilidade de aplicação desse sistema é na agricultura. Em plantações, sensores – de umidade ou temperatura, por exemplo – podem ser usados para fornecer dados para um

sistema de controle de irrigação. Os sinais desses sensores podem ser digitalizados e armazenados pelo sistema de aquisição de dados, para então serem transmitidos, processados e analisados.

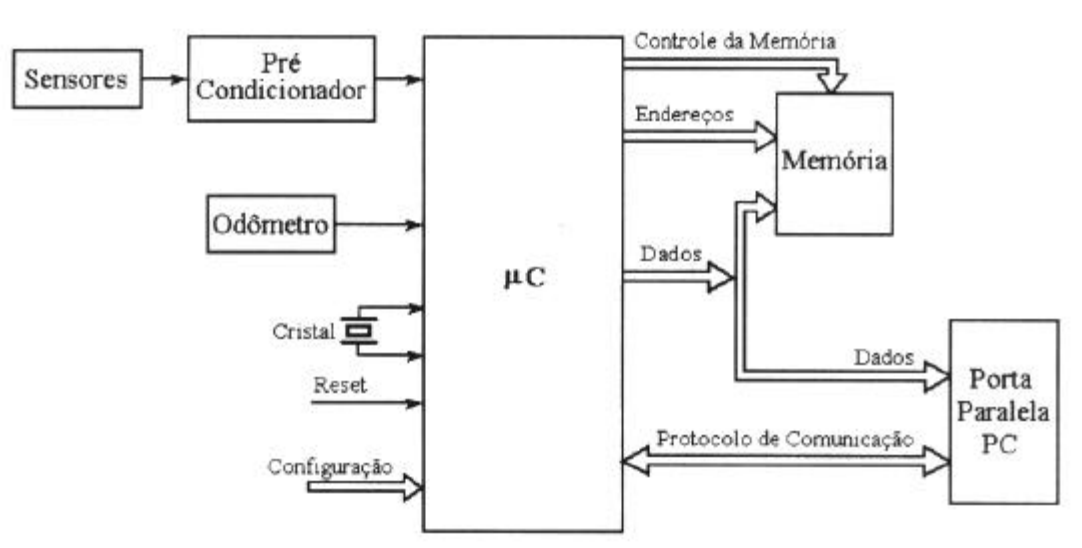


Figura 7.1 – Implementação do sistema de aquisição de dados com microcontrolador

Com essa proposta para continuidade de estudo na área, fecha-se aqui este trabalho e, a partir dele, algumas conclusões podem ser tiradas, sendo resumidas a seguir:

- Após vários testes realizados para caracterização e validação do circuito integrado DAQP02, foi observado que alguns resultados obtidos não estavam de acordo com os valores previstos. Verificou-se que isso era devido em parte a um mau funcionamento de certas partes do CI. Também foram encontradas falhas no circuito discreto do sistema, como trilhas descontinuadas nas placas de circuito impresso.
- A correção de algumas falhas foi viável. Algumas vezes, isso foi feito simplesmente por meio de soldagens, remendos ou mudanças a configuração do circuito. Outras falhas foram corrigidas com a anexação de pequenos circuitos auxiliares.
- Por outro lado, também foram verificadas anormalidades no funcionamento do circuito muito difíceis de serem completamente caracterizadas, não se chegando a descobrir exatamente sua causa. Estas se deviam a falhas internas do CI, mais precisamente em seu bloco analógico. Uma observação importante a ser feita é que, em circuitos VLSI,

os circuitos analógicos parecem ser mais passíveis de falhas de projeto que os circuitos digitais. Provavelmente por não possuírem tantos meios automatizados incluídos nas ferramentas de CAD quanto os digitais.

- No que se refere ao desenvolvimento de produto, podemos observar que localizar e corrigir defeitos num circuito integrado é um processo trabalhoso, demorado e caro. Isto porque, após a verificação de um mau funcionamento, deve-se corrigir o projeto original para que seja fabricada uma nova versão, e esta seja novamente testada. Este processo deve ser repetido quantas vezes for preciso até que se chegue à versão final do produto.
- Do ponto de vista da testabilidade, vemos que é de grande importância que haja meios de verificação dos pontos essenciais do circuito. Graças à existência de pinos que permitiam o acesso externo às principais entradas e saídas dos sub-circuitos implementados no CI, foi possível localizar algumas falhas.
- Numa nova proposta de implementação, vemos que, para futuros projetos, é uma boa idéia a construção de um equipamento baseado em microcontrolador comercial, por duas razões principais:
  - É viável se chegar a um produto que atenda muito bem às necessidades de um sistema de aquisição de dados em oleodutos de pequeno diâmetro;
  - O custo é bem mais baixo que um sistema baseado em processador dedicado.
- Assim, após todo o processo de diagnóstico e correção das falhas encontradas, conseguiu-se recuperar a funcionalidade de algumas partes do sistema. Apesar disso, a conclusão final é que é preferível não dar continuidade a este tipo de implementação, dando preferência a uma nova filosofia de projeto, mais barata e igualmente funcional.
- Finalmente, este trabalho se mostrou muito importante para o aprendizado de: técnicas de testes em circuitos integrados; métodos de teste e correção de falhas de circuitos eletrônicos discretos; e funcionamento de sistemas de aquisição de dados em oleodutos.

## Bibliografia

---

1. “Desenvolvimento de eletrônica de PIG para detecção de vazamento – Relatório Final”, LPC/UFRJ, 2001;
2. Dutra, Fábio da Silva – “Sistema de aquisição de dados em dutos de petróleo”, Dissertação (mestrado) - COPPE/UFRJ, Programa de Engenharia Elétrica, 2001;
3. Dutra, Fábio da Silva - “Unidade para aquisição de dados referentes a temperatura e pressão em dutos condutores de fluidos viscosos”, Projeto Final - Dep. de Eletrônica / Escola de Engenharia / UFRJ, 1999;
4. Oliveira, Alberto Alves de - "Logger para processamento de sinais em sistemas de monitoração de oleodutos", Dissertação (mestrado) - COPPE/UFRJ, Programa de Engenharia Elétrica, 2001;
5. Analog Devices Staff - “Analog-Digital Conversion Handbook”, Prentice-Hall, 1986;

6. Sedra, Adel S., Smith, Kenneth C. - “Microelectronic Circuits”, Oxford University Press, 1998;

7. Simons, Jim – “Matering The PIC16C7X A/D Converter”, Microchip Technology Inc., 1996;

Link: [www.microchip.com/download/appnote/articles/fact002.pdf](http://www.microchip.com/download/appnote/articles/fact002.pdf)

8. Baker, Bonnie – “Interfacing Pressure Sensors to Microchip’s Analog Peripherals”, Microchip Technology Inc., 2000.

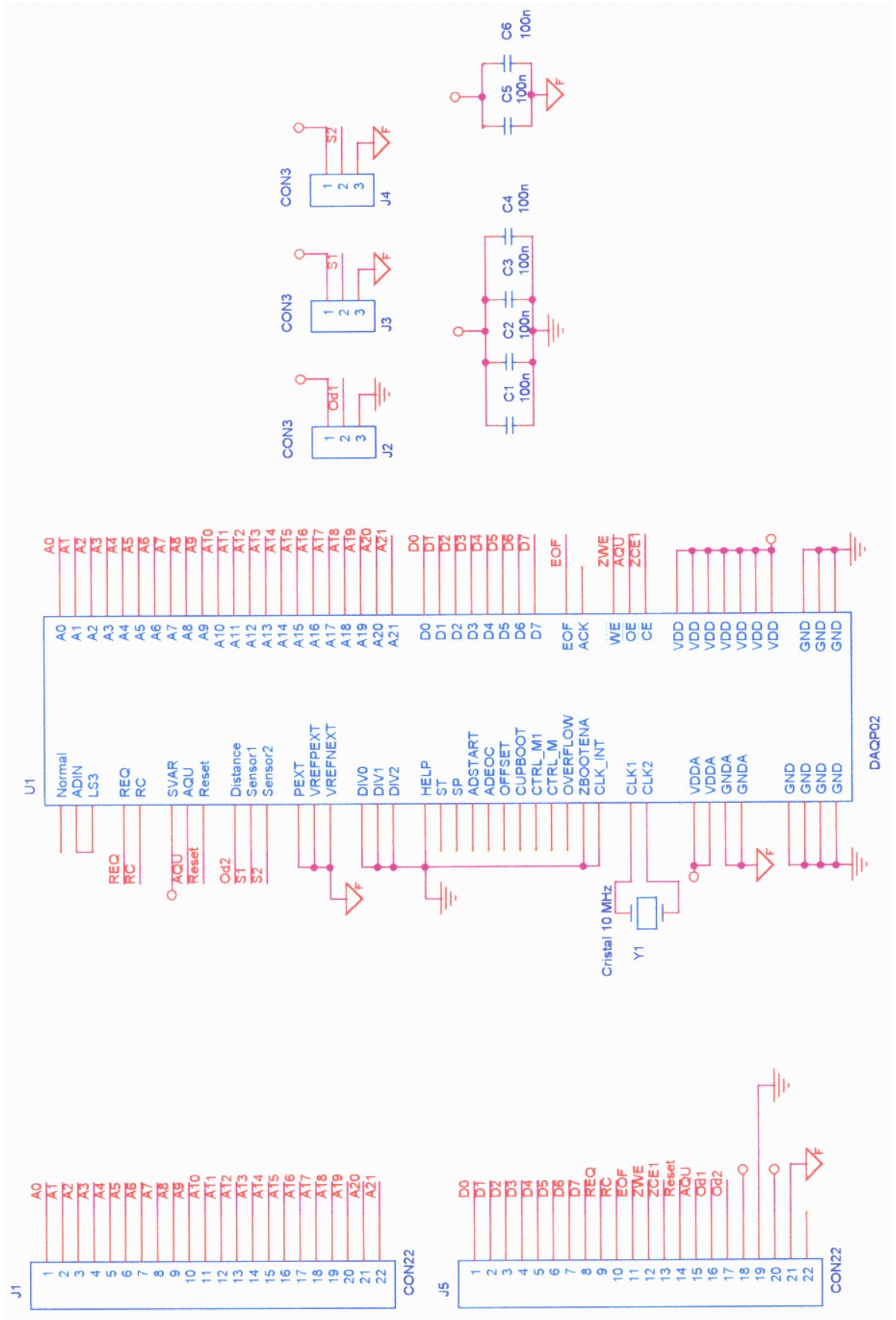
Link: [www.microchip.com/download/appnote/analog/opamps/00695a.pdf](http://www.microchip.com/download/appnote/analog/opamps/00695a.pdf)

## **Anexo I**

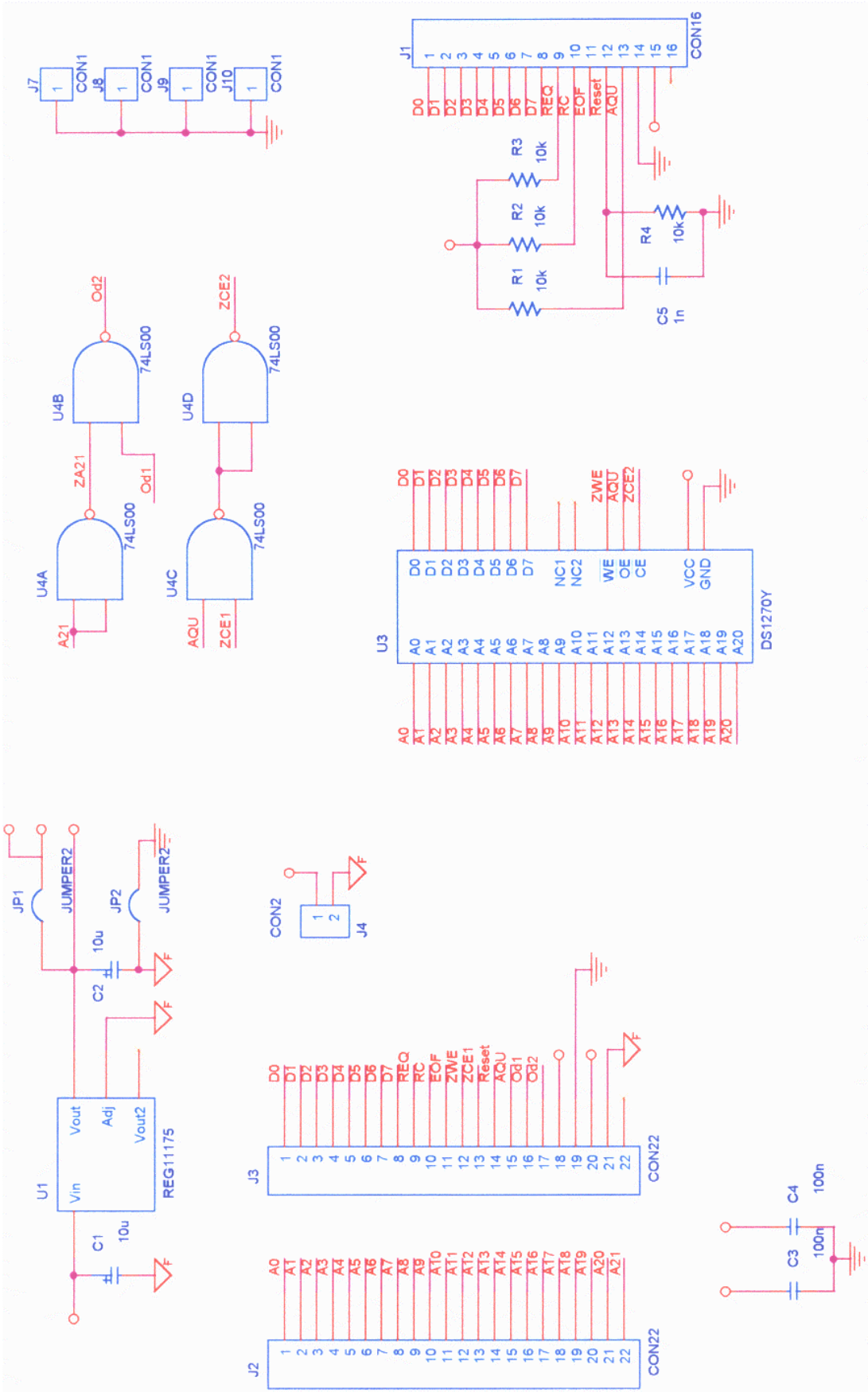
### **Esquemáticos e *Layouts* do PIGTP**

---

### Esquemático do PIGTP – Placa Superior

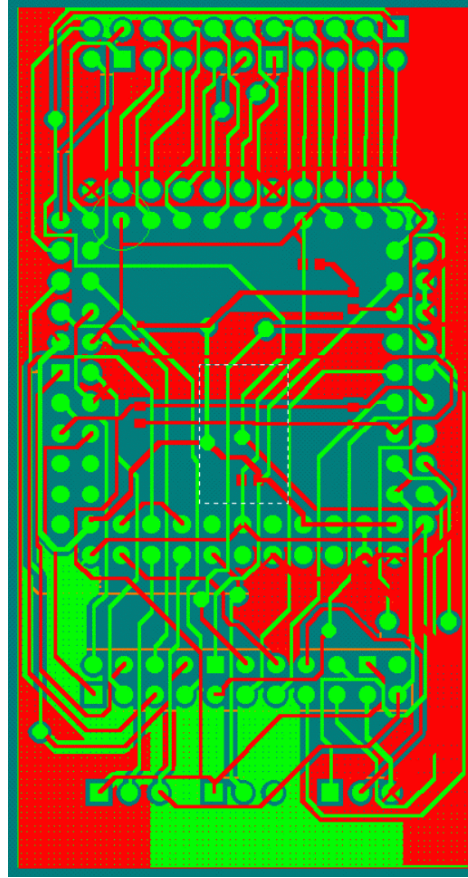


### Esquemático do PIGTP – Placa Inferior





Layout do PIGTP – Placa Superior



**Anexo II**

**Manuais dos Componentes**

---



















































































