

Centro Universitário Positivo – Unicenp  
Núcleo de Ciências Exatas e Tecnológicas  
Curso de Engenharia da Computação

**“DESENVOLVIMENTO DE UM SISTEMA ANALISADOR DE  
RESPIRADORES PULMONARES INTERLIGADO AO  
COMPUTADOR”**

Autor: Mauricio Perretto

Prof. Orientador: José Carlos da Cunha

Curitiba  
27/10/2002

## Sumário

<b>I - LISTA DE FIGURAS.....</b>	<b>I</b>
<b>II – LISTA DE ABREVIATURAS.....</b>	<b>II</b>
<b>III - LISTA DE TABELAS .....</b>	<b>III</b>
<b>1. INTRODUÇÃO .....</b>	<b>4</b>
<b>2. ESPECIFICAÇÃO .....</b>	<b>5</b>
2.1 DESCRIÇÃO DO PROJETO.....	5
2.2 ESTUDO TEÓRICO .....	7
2.2.1 HARDWARE .....	7
2.2.1.1 Sensores.....	7
2.2.1.2 Conversão A/D .....	9
2.2.1.2.1 Método da Tensão em Rampa.....	9
2.2.1.2.2 Método da Aproximação Sucessiva.....	11
2.2.1.2.3 Método <i>Flash</i> ou Conversão Paralela .....	13
2.2.1.3 Meios de Transmissão .....	13
2.3 ESPECIFICAÇÃO DE <i>HARDWARE</i> .....	15
2.3.1 SENSOR DE PRESSÃO .....	16
2.3.2 SENSOR DE FLUXO .....	17
2.3.3 CONVERSOR A/D .....	18
2.4 ESPECIFICAÇÃO DE <i>SOFTWARE</i> .....	19
2.5 ESPECIFICAÇÃO DE VALIDAÇÃO DO PROJETO .....	22
2.6 MÓDULO EXTRA .....	23
<b>3. PROJETO .....</b>	<b>24</b>
3.1 <i>HARDWARE</i> .....	24
3.1.1 Sensores.....	24
3.1.1.1 Sensor de Pressão .....	24
3.1.1.2 Sensor de Fluxo .....	25
3.1.2 Tratamento do Sinal.....	26
3.1.3 Conversão A/D .....	27
3.1.4 Transmissão .....	29
3.2 <i>SOFTWARE</i> .....	30
3.2.1 Itens de Tela .....	30
3.2.1.1 Frame 1 – Saída das informações .....	31
3.2.1.2 Frame 2 – Entrada de Dados.....	31
3.2.2 Menus .....	32
3.2.3 Arquivos de Log .....	33
3.2.4 Ciclo de execução .....	35
<b>4. IMPLEMENTAÇÃO .....</b>	<b>37</b>
4.1 CURVA DOS SENSORES E TRATAMENTO DO SINAL.....	37
4.1.1 Sensor de Pressão .....	37
4.1.2 Sensor de Fluxo .....	40
4.2 CONVERSÃO A/D E TRANSMISSÃO.....	43
4.3 SOFTWARE .....	44
<b>5. RESULTADOS.....</b>	<b>47</b>
5.1 TESTES DE VALIDAÇÃO.....	47
5.2 TESTE DE UTILIZAÇÃO .....	47

<b>6. CONCLUSÃO.....</b>	<b>51</b>
<b>7. REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>52</b>
<b>8. ANEXOS .....</b>	<b>53</b>

# I - LISTA DE FIGURAS

Figura 1 – Esquema em blocos do funcionamento do projeto .....	5
Figura 2 – Diagrama em blocos do sistema de <i>hardware</i> desenvolvido.....	6
Figura 3 – Diagrama em blocos da parte de <i>software</i> a ser desenvolvida .....	6
Figura 4 – Diagrama em blocos completo do sistema de análise .....	6
Figura 5 – Desenho de um extensômetro: (a) repouso; (b) tração; (c) compressão.....	8
Figura 6 – (a) raio do sensor passando entre as hélices (b) raio cortado pela hélice .....	8
Figura 7 – Configuração lógica para a técnica da Tensão em rampa.....	10
Figura 8 – Esquema lógico para o método da aproximação sucessiva .....	12
Figura 9 – Esquemático de um conversor A/D do tipo flash .....	13
Figura 10 – Modelo de pulmão artificial para testes.....	15
Figura 11 – Tela principal do <i>software</i> a ser desenvolvido .....	19
Figura 12 – Fluxograma de funcionamento básico do <i>software</i> .....	20
Figura 13 – DFD de nível 0 do sistema proposto .....	21
Figura 14 – Fluxograma do módulo extra a ser desenvolvido .....	23
Figura 15 – Conexão do Sensor de Pressão ao Amplificador de Instrumentação.....	25
Figura 16 – Etapa de Entrada do Sensor de Fluxo.....	26
Figura 17 – Amplificador não-inversor .....	27
Figura 18 – Esquema de ligação do circuito de conversão A/D .....	28
Figura 19 – Gerador de <i>clock</i> de 10 KHz para o ADC0808 .....	28
Figura 21 – Tela de definição de parâmetros para Análise.....	33
Figura 22 – Ciclo de Processamento do Software .....	35
Figura 23 – Fluxograma do Ciclo de Aquisição .....	36
Figura 24 – Curva de resposta do Sensor de Pressão.....	38
Figura 25 – Curva de resposta do circuito de tratamento do sinal de Pressão .....	39
Figura 26 – Curva de resposta do sensor de fluxo .....	41
Figura 27 – Curva de resposta do circuito conversor Frequência/Tensão .....	42
Figura 28 – Circuito de clock a cristal implementado .....	44
Figura 29 – Tela para seleção de log a ser visualizado.....	45
Figura 30 – Tela de um log anteriormente gravado.....	45
Figura 31 – Curvas obtidas pelo AVP .....	48
Figura 32 – Relatório final do AVP.....	48
Figura 32 – Relatório apresentado o limite de fluxo ultrapassado.....	49
Figura 33 – Log de erro salvo para futura consulta do técnico .....	50

## II – LISTA DE ABREVIATURAS

**A/D** – Analógico/Digital

**ABNT** – Associação Brasileira de Normas Técnicas

**ASP** – Ajustador do Sinal de Prova

**ASTM** – *American Society for Testing and Materials*/Sociedade Americana para Testes e Materiais

**CNTP** – Condições Normais de Temperatura e Pressão

**D/A** – Digital/Analógico

**LPM** – litros/minuto

**LSB** – Less Significant Bit – Bit Menos Significativo

**MSB** – *More Significant Bit* – Bit Mais Significativo

**PEEP** – *Positive End Expiratory Pressure* – Pressão Expiratória Positiva Final

**RS** – Registrador Sequencial

**SD** – Seqüenciador Digital

### III - LISTA DE TABELAS

Tabela 1 – Requisitos de <i>Hardware</i> .....	16
Tabela 2 – Características Principais do NPC1210 .....	17
Tabela 3 – Características Principais dos sensores da linha AWM700 .....	17
Tabela 4 – Características Principais do ADC0808.....	18
Tabela 5 – Atividades de teste do sistema proposto .....	22
Tabela 6 – Valores máximo e mínimo para as diversas propriedades .....	31
Tabela 7 – Campos de entrada de informações sobre o equipamento .....	31
Tabela 8 – Comandos do Menu Arquivo.....	32
Tabela 9 – Comandos do Menu Ajuda .....	33
Tabela 11 – Formato de gravação das informações no arquivo de log.....	34
Tabela 12 – Formato de gravação das informações no arquivo de erro.....	34
Tabela 13 – Relação pressão X saída do sensor .....	38
Tabela 14 – Valores do sinal do circuito .....	40
Tabela 15 – Valores de resposta do sensor de fluxo.....	41
Tabela 16 – Valores obtidos para o circuito conv. $F \rightarrow T$ .....	43
Tabela 17 – Parâmetros definidos.....	47
Tabela 18 – Valores amostrados pelo AVP .....	49

# 1. INTRODUÇÃO

A necessidade de auxílio ao processo respiratório, conhecido como Suporte Ventilatório, é comum a grande parte dos pacientes internados em Unidades de Terapia Intensiva, nas Salas de Emergência e nas Salas de Cirurgia [SEV - 00]. Ventiladores Pulmonares são dispositivos automáticos projetados para auxiliar ou substituir o ato da ventilação em pacientes que por algum motivo não podem efetuar o processo da respiração adequadamente, tendo deste modo uma insuficiência respiratória. Como os Ventiladores Pulmonares são classificados pelo Ministério da Saúde como equipamentos médico-hospitalares de alto risco, a realização de testes de desempenho precisos é indispensável para diagnosticar falhas ou funcionamento inadequado que podem resultar em acidentes durante sua utilização e, em algumas vezes, com lesões irreparáveis.

Os equipamentos médico-hospitalares para auxílio respiratório têm diversas normas de funcionamento. No Brasil a norma utilizada é a da ABNT (Associação Brasileira de Normas Técnicas) NBR-13763, que tem diversas definições e requisitos para os equipamentos ventiladores. Apesar disto, a norma brasileira não contém especificações para equipamentos de análise destes equipamentos. Uma norma internacional que contém especificações referentes as análises é a da ASTM (*American Society for Testing and Materials* – Sociedade Americana para Testes e Materiais), sendo a designação da norma F-1100. Nesta estão relatados os níveis normais de pressão, volume, fluxo e complacência utilizados em testes, parâmetros base deste projeto.

Este trabalho visa detalhar o processo de desenvolvimento de um dispositivo que seja capaz de monitorar estes equipamentos através da leitura dos parâmetros básicos da respiração, interfaceando com o computador, onde será possível, ao profissional responsável pela manutenção deste equipamento fazer uma análise mais precisa das condições de funcionamento do ventilador.

## 2. ESPECIFICAÇÃO

Nesta etapa descreveremos a especificação do projeto que consiste em especificar e delimitar a sua área de implementação, assim como a aplicação do mesmo.

### 2.1 DESCRIÇÃO DO PROJETO

Ventiladores pulmonares são equipamentos eletro-mecânicos utilizados para o auxílio do processo respiratório. Esses equipamentos podem trabalhar de duas formas distintas: na primeira, a ventilação assistida, o paciente deve efetuar pelo menos o esforço inicial da respiração para que o equipamento continue o ciclo da respiração, já no caso da ventilação intermitente o dispositivo trabalha sempre no mesmo ciclo [STO - 99]. Nos dois casos existem os limites de ciclagem, que são os pontos atingidos pelo respirador para que ele passe da fase inspiratória à expiratória. Esses limites podem ser definidos em função da pressão, volume e fluxo dos gases da respiração. Desses quatro parâmetros, serão monitorados a pressão e o fluxo do ar para efetuar a análise do equipamento. Foram definidos esses dois parâmetros, pois através do fluxo pode-se calcular o volume, tendo-se assim os três parâmetros fundamentais.

A interligação com o computador se faz necessária para que o sistema possa se tornar mais portátil, sendo facilmente transportado até os equipamentos que deverão ser testados. Desta forma, pode-se definir dois blocos distintos no projeto: *hardware*, externa ao computador e ligada ao equipamento a ser analisado, que irá fazer a aquisição e tratamento dos dados; e *software*, interno, que irá fazer a apresentação gráfica ao usuário. Essa ligação entre interface externa e interna é mostrada no diagrama da figura 1.

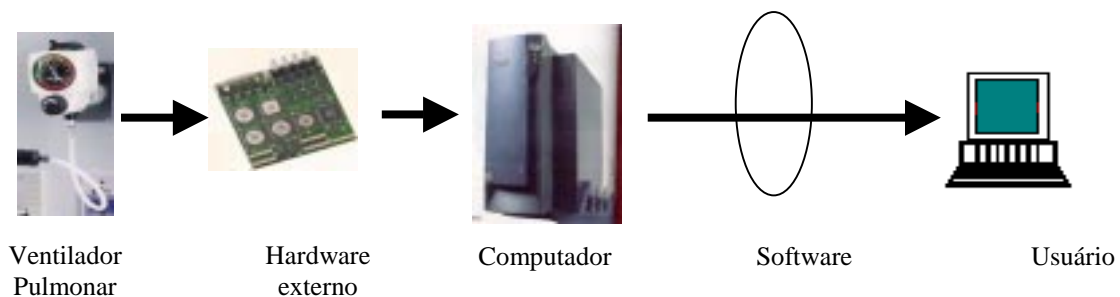


Figura 1 – Esquema em blocos do funcionamento do projeto



Esses dois blocos podem ser divididos em vários módulos para que se torne mais fácil a implementação, podendo-se fazer uma análise melhor do projeto. Todos os módulos são interconectados sequencialmente, com o *hardware* dividido em sensores, que irão fazer a transformação direta das grandezas especificadas; tratamento do sinal transmitido pelos sensores; multiplexação entre os diversos sinais para que apenas um seja transmitido a cada vez; conversão A/D (Analógico/Digital), que deve ocorrer para que o dado possa ser tratado pelo microcomputador e a comunicação entre esse *hardware* e o computador. O diagrama dos diversos módulos do *hardware* é apresentado na figura 2.

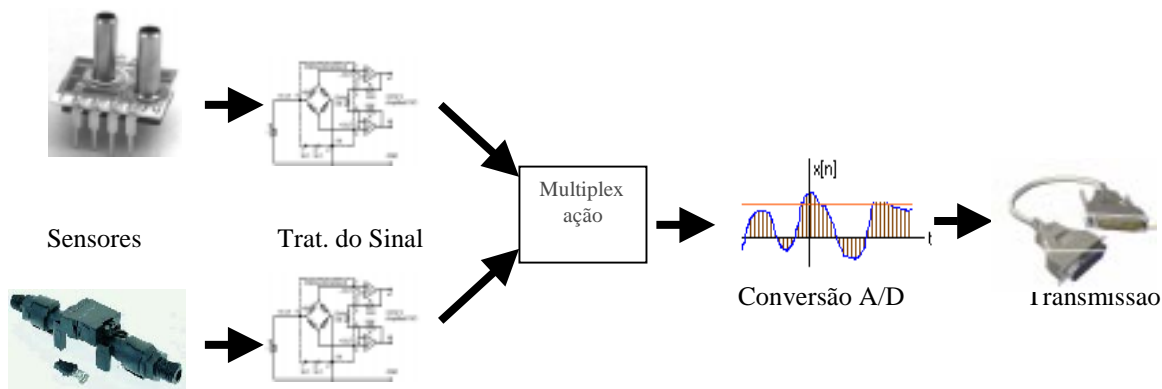


Figura 2 – Diagrama em blocos do sistema de *hardware* desenvolvido

O *software*, por sua vez, é dividido em três blocos: recepção dos dados do *hardware*, cálculo das diversas variáveis e apresentação gráfica. Esse diagrama em blocos é mostrado na figura 3 e a interligação entre todos os módulos de *hardware* e *software* está na figura 4.

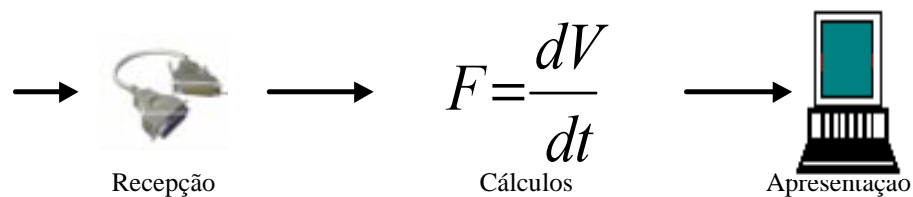


Figura 3 – Diagrama em blocos da parte de *software* a ser desenvolvida

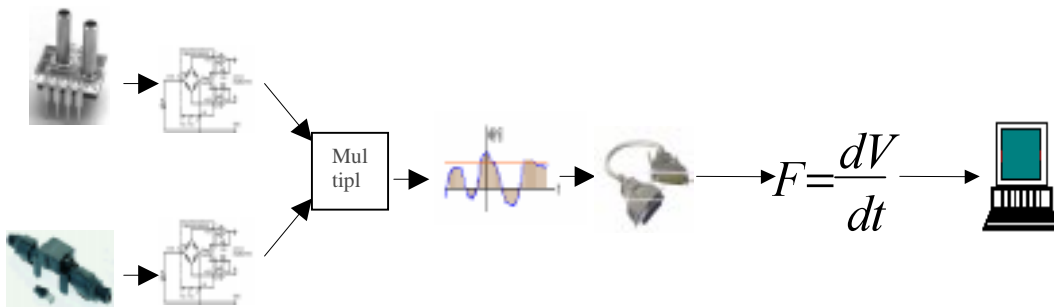


Figura 4 – Diagrama em blocos completo do sistema de análise

## 2.2 ESTUDO TEÓRICO

Como visto na figura 1, o sistema todo pode ser dividido em duas partes principais onde cada uma tem seus módulos específicos. Neste estudo teórico são apresentados os módulos de *hardware*.

### 2.2.1 HARDWARE

O *hardware* do sistema foi definido em 5 módulos funcionais sequenciais nos quais se executa a leitura de variáveis de medida, tratamento destas variáveis, conversão destas do formato analógico para o digital e transmissão para o computador. Essas tarefas são executadas pelos sensores, amplificadores operacionais (tratamento do sinal), conversores A/D e um meio de transmissão, respectivamente. Na sequência é explicada a relação de cada bloco no sistema funcional, como e porque estes são utilizados.

#### 2.2.1.1 Sensores

Na ventilação pulmonar existem três grandezas fundamentais que necessitam serem medidas: pressão, volume e fluxo. Destas, duas são intrinsecamente interligadas já que o volume é a integração do fluxo de ar que passa pelo sensor [TOM - 81]. Desta forma, decidiu-se por utilizar apenas sensores para duas grandezas (pressão e fluxo) e calcular a outra de forma computacional.

As grandezas a serem medidas podem ser definidas da seguinte forma:

- pressão  $\rightarrow$  é a força exercida sobre uma determinada área em um dado instante.
- fluxo  $\rightarrow$  é a taxa de deslocamento de uma massa em um meio, ocasionada por uma diferença de pressão.

Essas duas grandezas analógicas, como pode ser visto por suas definições, não contém nenhuma variação direta em relação a grandezas elétricas. Em projetos eletrônicos de monitoração de variáveis ambiente nestas condições, há sempre a necessidade da conversão da grandeza a ser medida em uma variável elétrica e, desta forma, poder-se iniciar tratamento deste sinal que foi convertido.

Normalmente a pressão é medida através de *strain gauges*, que são transdutores que tem como variável de medida a alteração da resistência. Estes partem do princípio de que quando um objeto está sobre pressão, sofre uma deformação em sua estrutura básica, fenômeno mostrado na figura 5.

Como não é possível medir a resistência elétrica de um material, faz-se necessário um circuito auxiliar no qual ocorra a variação de outra grandeza elétrica relacionada à resistência. Normalmente é utilizada a variação da tensão para se medir a variação destes sistemas. Transdutores deste tipo são chamados de moduladores, pois modulam a energia de outra fonte para produzir um sinal proporcional a entrada na saída.

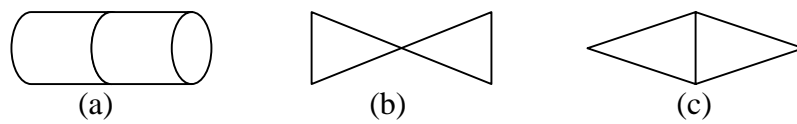


Figura 5 – Desenho de um extensômetro: (a) repouso; (b) tração; (c) compressão

Já os transdutores de fluxo de ar podem ser desenvolvidos de quatro formas diferentes: transdutor de rotação, ultrasônico, convecção térmica e através da diferença de duas pressões.

Na primeira forma, a medição do fluxo é feito através de um método parecido com o utilizado para efetuar o cálculo da velocidade radial de um dado objeto, utilizando uma hélice que, girando com a força do fluxo, corta um raio infravermelho colocado em seu caminho, de forma que haverá um sinal de saída no receptor do infravermelho no qual a frequência dos pulsos gerados é proporcional a velocidade de rotação da hélice e esta por sua vez é compatível com o fluxo de ar que a impulsiona. Um esquema desta forma de medição é mostrado na figura 6.

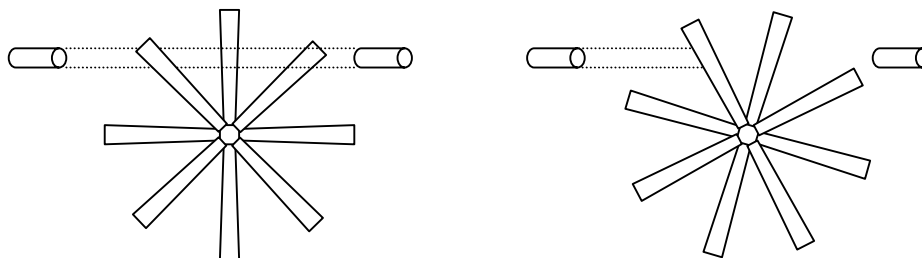


Figura 6 – (a) raio do sensor passando entre as hélices (b) raio cortado pela hélice

O transdutor de ultrassom se utiliza do mesmo princípio sendo que o valor de saída da frequência é proporcional a três grandezas: fluxo, temperatura e composição, não sendo ideal para o caso proposto.

A convecção termal se baseia no aquecimento de um elemento sensor de temperatura através de uma corrente elevada, este processo é descrito pela Lei de Joule, com o fluxo de ar passando diretamente sobre o elemento sensor esta temperatura tende a diminuir fazendo com que o sensor de temperatura tenha uma resposta diferente e, desta forma, torna possível a medição da temperatura.

No último caso, o sensor é desenvolvido a partir da definição física de fluxo que é: a diferença entre duas pressões sobre uma resistência. A partir de um tubo onde a resistência é conhecida coloca-se um sensor de pressão em cada extremidade e, calcula-se o fluxo a partir da diferença entre as saídas destes dois sensores.

### 2.2.1.2 Conversão A/D

Como dito anteriormente o processo de conversão A/D é uma das fases mais importantes em um projeto onde se envolve o uso de um computador, pois a maioria das grandezas do mundo real é medida de uma forma analógica e contínua, sendo que normalmente para se efetuar as operações computacionais necessita-se de um dado amostrado digitalmente.

Existem diversos métodos utilizados para se efetuar a conversão A/D. Na sequência serão mostrados os três métodos mais utilizados atualmente nos dispositivos comerciais.

#### 2.2.1.2.1 Método da Tensão em Rampa

Esse método é o mais simples e, também, o menos usado, sendo normalmente utilizado em instrumentos de medidas, como voltímetros digitais.

O método de conversão da tensão em rampa funciona pelo seguinte esquema [KHA – 84], e, para melhor compreensão, compare com a figura 7:

1. é aplicado um sinal de entrada analógico, para ser digitalizado, na entrada do comparador;
2. um nível lógico alto “1” é aplicado no terminal de início, iniciando o processo de conversão executando as três seguintes funções:

- 2.1 apaga o contador, removendo, desta maneira, qualquer informação nele contida, da operação anterior.
- 2.2 Coloca o circuito biestável R-S no estado Q, ativando, assim a porta AND A1.
- 2.3 Inicia o gerador de tensão em rampa, através da porta AND A2.
3. A tensão de referência do gerador de rampa é alimentada na segunda entrada do comparador e esta tensão aumenta linearmente.
4. Neste meio tempo, os pulsos do marcador de tempo fixo são retidos no contador através de uma porta AND A1 e começam a incrementá-lo.
5. Quando a tensão em rampa atinge e excede ligeiramente a tensão da entrada analógica, a saída do comparador muda de “0” para “1”.
6. Isto restaura o circuito biestável para o lado  $\sim Q$ , desativando, desta maneira, a porta AND A1 e evitando que os pulsos do marcador de tempo incrementem, posteriormente, o contador.
7. A saída em “1” do comparador também desativa a porta AND A2, que, por sua vez, DESLIGA o gerador de tensão em rampa, retornando, assim, a saída do comparador para um “0”, em preparação para a próxima operação de conversão.
8. O contador contém o valor equivalente em digital do sinal de entrada analógica.

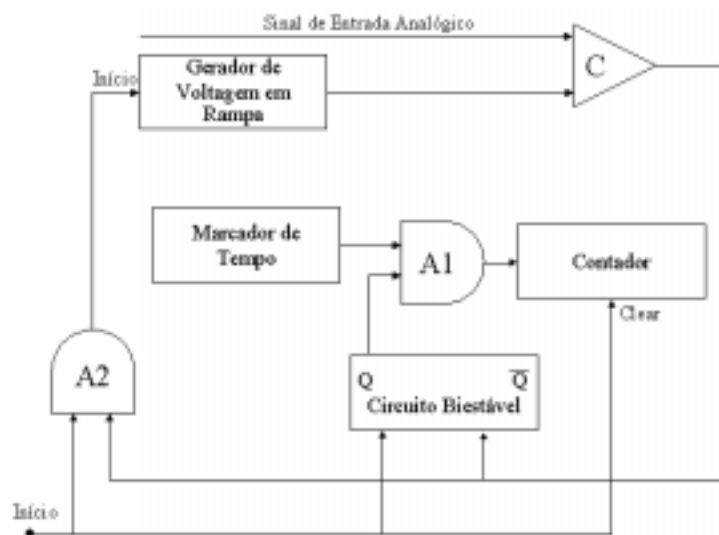


Figura 7 – Configuração lógica para a técnica da Tensão em rampa

Para uma melhor compreensão do circuito acima descrito, abaixo é mostrado um algoritmo de conversão:

1. O conversor recebe o valor analógico, um pulso zera o contador, e segue-se o seguinte laço:

2. Soma-se um 1 bit;
3. Converte o valor em conversor D/A;
4. Compara o valor de entrada com o valor do conversor D/A;

Se o valor do conversor é maior ou igual termina-se a conversão;

Senão continua-se até ser satisfeito esse critério.

Este método tem a desvantagem de o contador ter que alcançar a contagem máxima em um período de tempo extremamente curto, em decorrência de uma entrada analógica máxima.

#### 2.2.1.2.2 Método da Aproximação Sucessiva

Esse método é o mais amplamente usado na conversão A/D, sendo utilizado na maioria dos conversores A/D atuais [MAL – 85].

O método da aproximação sucessiva obedece a seguinte seqüência, para compreensão compare com a figura 8:

1. SD (Sequenciador Digital) envia um “1” na linha MSB (Bit Mais Significativo). Isto, também, ativa a porta A8. A entrada para o DAC (*Digital/Analogic Converter* – Conversor Digital/Analógico) é 10000000 em binário.
2. A saída do DAC é introduzida no ajustador do sinal de prova (ASP). O ASP emite um sinal de prova que é a metade da escala.
3. Se o sinal de entrada analógica for menor que o sinal de prova, então o comparador emite um “0”; isto é alimentado na segunda entrada de A8. A saída de A8 é “0”, portanto o bit 8 (o MSB) de RS (Registrador Sequencial) continua um “0”.
4. Se o sinal de entrada analógica for maior que este sinal de prova, o comparador emite um “1” que ativa A8 completamente. Neste caso, um “1” vai ser carregado na posição de bit 8 do RS.
5. A saída “1” de A8 é, também, alimentada no ASP. Isto diz ao ASP para somar uma metade do valor.
6. Agora, o SD emite um “1” na próxima linha de saída inferior. A entrada para o DAC é 01000000 e a saída é a metade do valor da saída anterior.

7. ASP vai ajustar a saída maior ou menor, dependendo se a saída da operação anterior estava em “1” ou em “0”.
8. Este processo é repetido com SD emitindo um “1” digital em cada linha de saída subsequente em cada etapa sucessiva. Portanto, em cada etapa sucessiva, o processo resulta em uma aproximação que fica mais perto do sinal de entrada analógica.
9. Após 8 fases do processo, o RS conterá a aproximação em binário do sinal de entrada analógica.

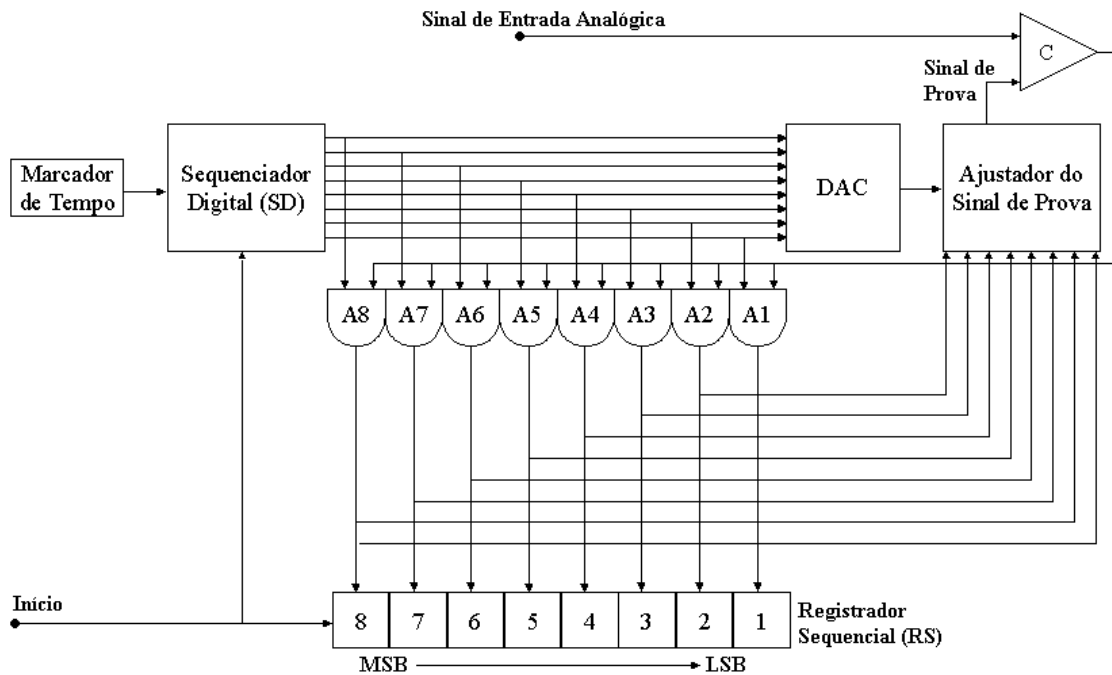


Figura 8 – Esquema lógico para o método da aproximação sucessiva

Também, para melhor compreensão do método visto acima, na sequência é descrito um algoritmo:

1. conversor recebe o valor analógico, e segue-se o seguinte laço:
  2. Um sequenciador digital joga um nível lógico “1” no MSB, e um conversor D/A converte esse valor e compara com o sinal analógico;
 

Se for maior ele armazena “0” em um registrador sequencial;

Senão armazena-se “1”.
  3. Passa-se para o próximo MSB até se chegar ao LSB (bit menos significativo), o valor digital aproximado estará no registrador sequencial

### 2.2.1.2.3 Método *Flash* ou Conversão Paralela

Esse método de conversão é o mais veloz de todos, e também o mais dispendioso tanto em custo quanto em componentes utilizados. O conversor *flash* se utiliza de diversos comparadores ligados em uma de suas entradas a uma parte de um divisor resistivo, fazendo com que cada comparador corresponda a um nível de tensão diferente. Na outra entrada dos comparadores é ligado diretamente o sinal de entrada. Todos os comparadores que tenham um nível de tensão do divisor menor que o sinal de entrada serão ativados e, desta forma, o sinal de saída corresponderá ao bit do maior comparador ativado. O esquemático de um conversor flash é mostrado na figura 9.

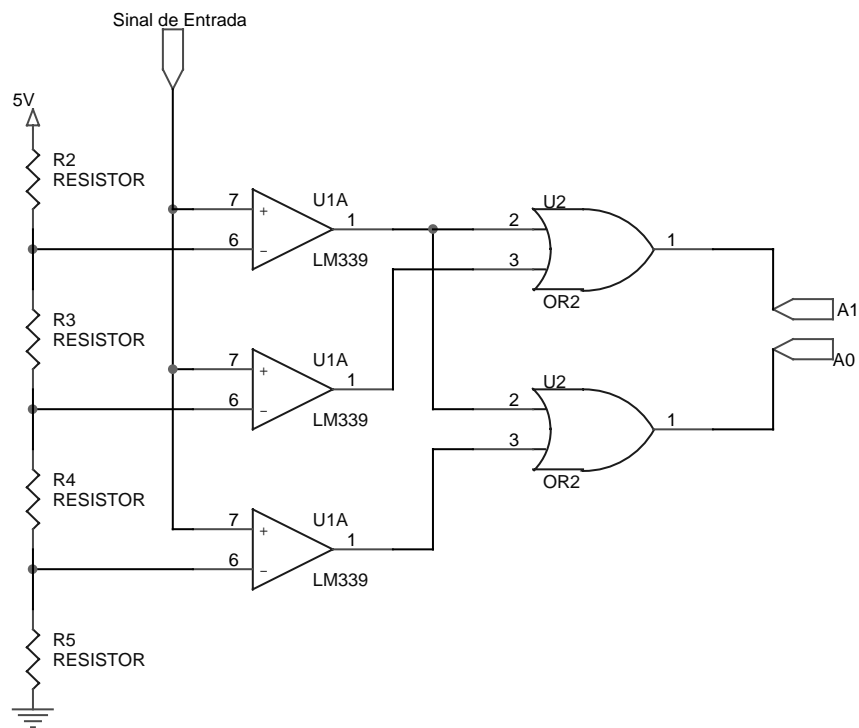


Figura 9 – Esquemático de um conversor A/D do tipo flash

Após a fase de conversão o sinal em si já estará digitalizado e pronto para ser transmitido ao computador, restando definir de que forma será feita essa transmissão. Esta definição será vista na sequência.

### 2.2.1.3 Meios de Transmissão

Após todo o tratamento do sinal analógico e sua digitalização pode-se manipular os dados no computador, bastando para isso a transmissão dos dados do circuito de *hardware*



desenvolvido para o computador. Atualmente existem duas formas distintas de se fazer a comunicação com um dispositivo computacional: a transferência de dados serial ou a paralela.

A comunicação serial, como seu nome diz, baseia-se na transmissão dos bits da informação de forma serial, ou em sequência. Para isto ela precisa de apenas uma linha de transmissão e outra de recepção e, além disso, existem algumas outras linhas para o controle da transmissão entre dois equipamentos [KRI – 00].

As principais vantagens da comunicação serial são: baixo custo, maior distância de transmissão, menor sensibilidade a ruídos externos. A comunicação serial é ideal para a transmissão de dados através de dispositivos microcontrolados, onde é necessário a maior minimização possível de um projeto.

Já a comunicação paralela se baseia na transmissão paralela de todos os bits que compõem um byte. Normalmente a porta paralela do PC tem três registradores ligados a esta: registradores de dados, *status* e controle da comunicação paralela.

A principal vantagem da comunicação paralela é a menor complexidade do *hardware* a ser desenvolvido mas em maior espaço físico. Também pode-se citar a velocidade de transmissão dos bytes como uma vantagem da transmissão paralela. O principal problema desta forma de comunicação é a sua pequena distância e baixa imunidade a ruídos.

Existem diversas outras formas de comunicação com o computador: slot de expansão, placa de rede, *wireless*, além das abordadas neste tópico. Decidiu-se por não descrevê-las aqui porque se acredita que estas formas de comunicação são definidas em um nível mais alto de complexidade ao do circuito proposto. Neste projeto optou-se por desenvolver a comunicação de forma paralela diminuindo-se assim o tempo necessário para implementação da comunicação entre o *hardware* e o *software* desenvolvidos.

## 2.3 ESPECIFICAÇÃO DE *HARDWARE*

Segundo a norma F-1100 [ASTM - 90], um analisador deve ser desenvolvido seguindo o modelo apresentado na figura 10. Este modelo descreve o desenvolvimento de um pulmão artificial seguindo algumas características de uma pessoa, tanto doente quanto saudável, para que possibilite a medida real dos parâmetros do ventilador. Um detalhe importante é que o ambiente externo onde estiver localizado o modelo de pulmão artificial deverá estar o mais próximo possível da CNTP (Condições Normais de Temperatura e Pressão; 22°C e 1 atm).

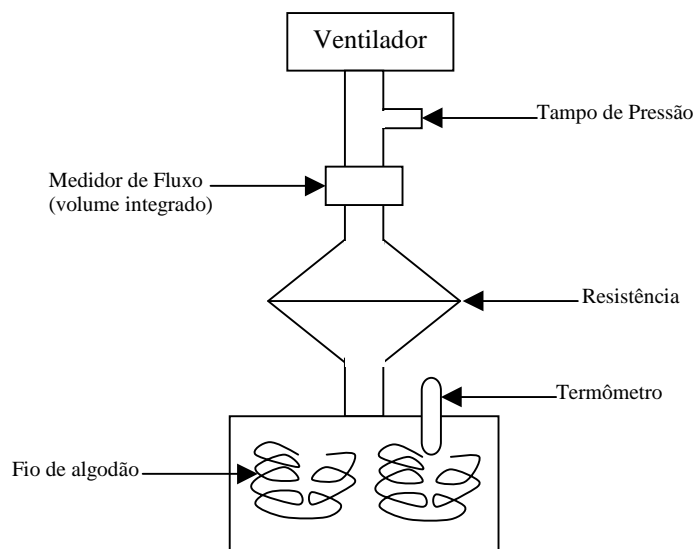


Figura 10 – Modelo de pulmão artificial para testes

Como dito anteriormente o sistema é dividido em dois blocos principais o *hardware* e o *software*. O *hardware* tem como função principal a detecção das grandezas fundamentais do sistema (fluxo e pressão) o tratamento destes sinais para que se tornem identificáveis e a conversão A/D, responsável pela transformação do sinal analógico para um sinal digital que será recebido pelo *software* presente no computador. Todo este diagrama de *hardware* é mostrado na figura 2. O sistema proposto deve ser de baixo consumo de energia para que se torne o mais portátil possível. Além disso o sistema deve ter uma boa repetibilidade de seus dados, pois será utilizado para a análise de equipamentos de alto risco em hospitais e, por estes motivos, cada componente do sistema deve preencher alguns requisitos mínimos, que são mostrados na tabela 1.

Tabela 1 – Requisitos de *Hardware*

<b>Componente</b>	<b>Requisitos Mínimos</b>
Sensor de pressão	deve atuar na faixa de 0 – 100 cm H <sub>2</sub> O / 600mmHg
Sensor de fluxo	Deve ter uma faixa de atuação de 0 – 120 l/min
Conversor A/D	Mínimo de 8 bits para que tenha uma conversão precisa, além disso deve possuir entradas multiplexadas para leitura de diversos sinais.

Como dito anteriormente os ventiladores pulmonares são equipamentos de missão crítica, não podendo estarem descalibrados ou em mau funcionamento. Desta forma, qualquer projeto que envolva o desenvolvimento de um ventilador pulmonar ou outro equipamento no auxílio da ventilação pulmonar, deve ter todos seus componentes rigidamente avaliados. Novamente frisa-se que os componentes a serem utilizados, como sensores e conversor devam ser analisados com o máximo cuidado e estes componentes é que virão a definir o sucesso do desenvolvimento. Baseado-se nesta análise, os principais componentes escolhidos são apresentados na sequência.

### 2.3.1 SENSOR DE PRESSÃO

O sensor de pressão tem papel fundamental na análise de ventiladores pulmonares, pois através dele pode-se medir a pressão do ar exercida dentro do pulmão. Esta informação tem grande relevância quanto analisada junto com a complacência pulmonar, que esta relacionada ao valor da elasticidade do pulmão. Quanto mais complacente o pulmão, maior poderá ser a pressão exercida sobre ele sem ocorrer danos a estrutura pulmonar conhecida como alvéolo. Se for aplicada uma pressão acima do valor de complacência do pulmão, esta poderá acarretar sérios danos ao órgão [MAI - 92].

Segundo a norma NBR-13763 [BRA - 96], na realização da análise de ventiladores pulmonares os testes devem ser realizados com três valores distintos de pressão, 4,9, 6,86 e 9,8 KPa.. Depois, deve-se avaliar a pressão no tempo de 100 respirações e verificar a exatidão do equipamento. Para os testes de fluxo de ar deve-se gerar um fluxo de 50L/min sem PEEP (*Positive End Expiratory Pressure* – Pressão Expiratória Positiva Final)e a pressão não deve ultrapassar o valor de 0,49KPa [MUC - 96].

Para estes fins foi escolhido o sensor de pressão o NPC1210 da NovaSensor. O NPC1210 é um sensor ideal para monitoração de baixas pressões, incluindo nestes:

ventilação, monitoração de fluxo de ar, respiradores, etc [LUC - 02]. Este é encapsulado em um pequeno invólucro DIP-8 e suas principais características são apresentadas na tabela 2.

Tabela 2 – Características Principais do NPC1210

<b>Símbolo</b>	<b>Parâmetro</b>	<b>Margem</b>	<b>Unidade</b>
$V_{cc}$	Tensão de alimentação	0-50	MiliVolts
$T_A$	Temperatura	-40 até +125	Celsius
FSO	Linearidade	0.5%	
$I_i$	Impedância de entrada	4	KiloOhm
$P_{ov}$	Pressão Máxima	5	Psi

### 2.3.2 SENSOR DE FLUXO

Os sensores de fluxo utilizados em projetos de ventilação pulmonar tem como particularidade a grande faixa de atuação, pois o fluxo de ar pode variar muito rapidamente e de forma acentuada. Com este intuito, surge a necessidade de um sensor que trabalhe com uma faixa de escala maior que as escalas utilizadas habitualmente, entre 0 e 20 LPM (litros/minuto).

O sensor utilizado para este fim no projeto a ser desenvolvido é fabricado pela Data Instruments, sendo da série AWM700. Essa série é especial para equipamentos que necessitam de uma grande escala. As principais características dos sensores da família AWM700 são apresentados na tabela 3.

Tabela 3 – Características Principais dos sensores da linha AWM700

<b>Símbolo</b>	<b>Parâmetro</b>	<b>Margem</b>	<b>Unidade</b>
$V_{cc}$	Tensão de alimentação	8 até 15	Volts
$T_A$	Temperatura	-25 até +85	Celsius
$P_D$	Potência Dissipada	<85	MiliWatts
Fe	Escala de fluxo	0-200	Litros/Minuto
$T_{wr}$	Tempo de resposta	6	miliSegundos

### 2.3.3 CONVERSOR A/D

Os conversores A/D e D/A são componentes especiais em projetos que envolvem a comunicação entre o mundo externo e o computador, pois são a ponte que liga estes dois mundos.

Na realização do projeto será necessário a utilização de um conversor que tenha pelo menos duas entradas distintas de informação, cada uma representada por um sensor descrito acima. Tendo em vista esses fatores foi definido como componente conversor o ADC0808.

O ADC0808 é um conversor analógico/digital com resolução de 8 bits e saída paralela, desenvolvido utilizando-se o método da aproximação sucessiva para efetuar a conversão [NAT - 99]. As principais características do ADC0808 são:

- 8 entradas multiplexadas
- Saída em *tri-state*
- Pino de fim de conversão e interrupção
- *Clock* externo
- Velocidade de conversão de 100 $\mu$ s
- Resolução máxima de 75  $\mu$ V.

As principais características físicas do ADC0808 são apresentadas na tabela 4.

Tabela 4 – Características Principais do ADC0808

Símbolo	Parâmetro	Margem	Unidade
V <sub>cc</sub>	Tensão de alimentação	4,5 até 6,5	Volts
T <sub>A</sub>	Temperatura	-65 até +150	Celsius
P <sub>D</sub>	Potência Dissipada	875	MiliWatts
F <sub>c</sub>	Frequência de <i>Clock</i>	10-1280	KiloHertz
t <sub>ws</sub>	Pulso mínimo de <i>start</i>	100-200	NanoSegundos

## 2.4 ESPECIFICAÇÃO DE SOFTWARE

O *software* a ser desenvolvido utilizará como sistema operacional o Windows 9X/ME. Este sistema foi o escolhido por ser atualmente o mais utilizado em desktops pessoais e dispositivos portáteis. Futuramente será estudado uma forma de implementar o *software* no sistema operacional Palm OS, o que permitiria a utilização do *software* em equipamentos menores e mais fáceis de serem carregados.

Como compilador será utilizado o Borland C++ Builder, por ser o ambiente de desenvolvimento rápido em C mais amigável ao programador. O *software* não terá uma grande complexidade, tendo que captar os dados transmitidos, calcular as informações que serão apresentadas além das transmitidas, como volume, complacência, etc, e apresentar estes dados de uma forma fácil ao usuário. Além disso, haverá a criação de um arquivo de log o que possibilitará o estudo de dados após a verificação do ventilador. Neste arquivo deverão ser armazenados, além dos dados colhidos pelo *software*, dados do ventilador como marca, modelo, número de série e comentários sobre as últimas amostras. Todos esses dados fornecidos pelo usuário. A interface principal do *software* é apresentada na figura 11.

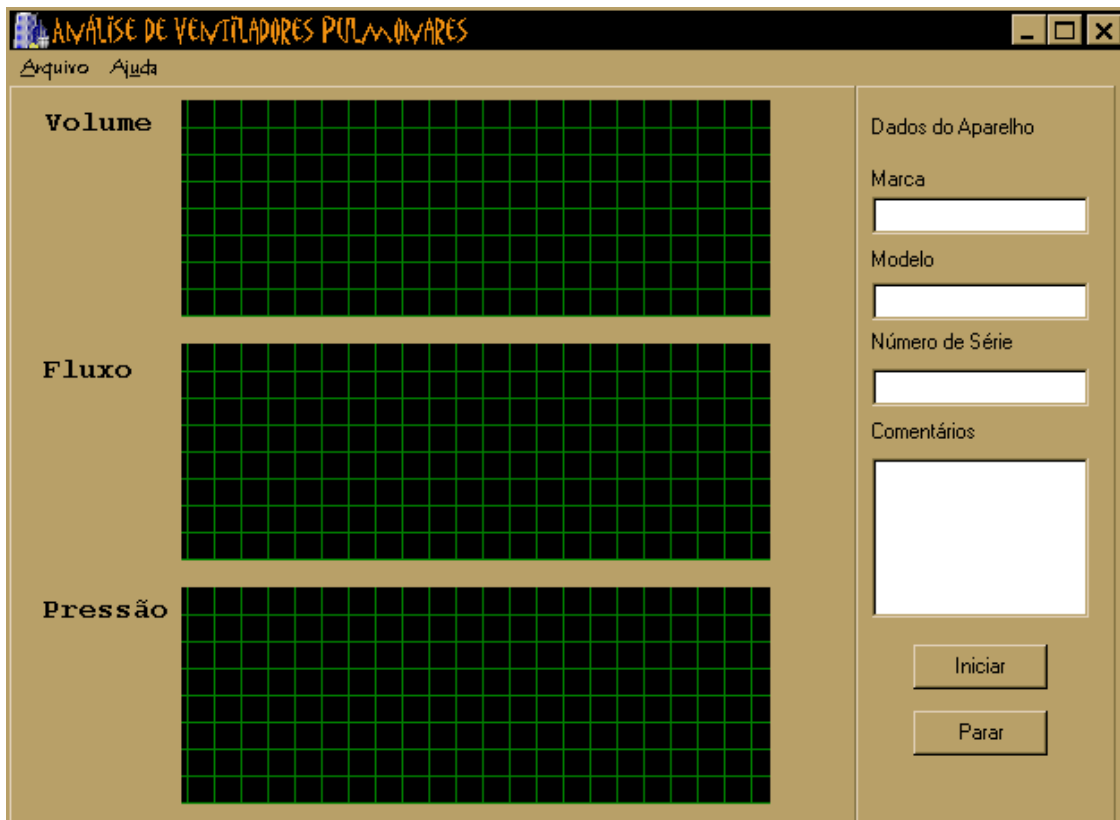


Figura 11 – Tela principal do *software* a ser desenvolvido

A partir do momento que o usuário acione o botão iniciar, o programa executará um ciclo de leitura dos sensores, cálculo de variáveis, apresentação visual ao usuário e criação de arquivo de log. Esse fluxograma é mostrado na figura 12, esse ciclo só é interrompido no momento em que o usuário acionar o botão parar.

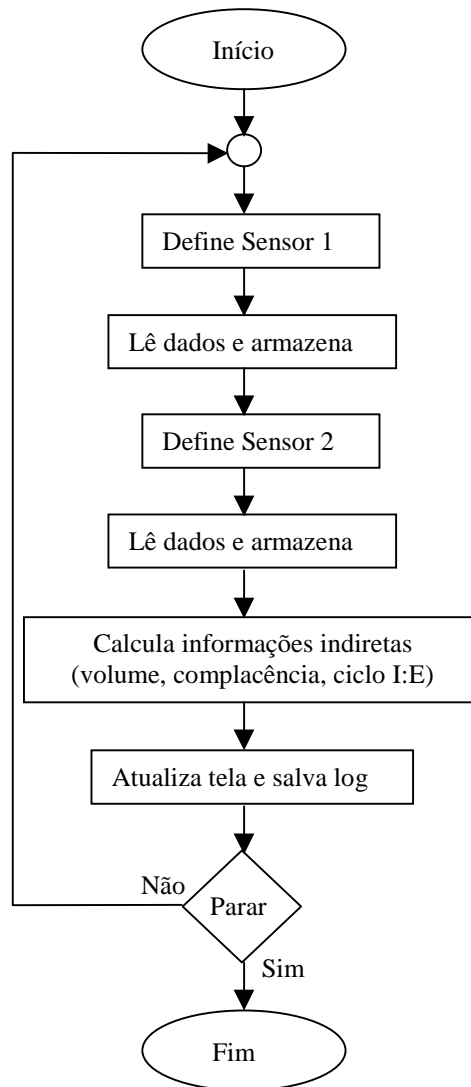


Figura 12 – Fluxograma de funcionamento básico do *software*

Haverá três métodos de entrada de dados no *software*, e cada um desses métodos é efetuado por um agente externo distinto. Além disso o sistema terá algumas saídas padrões de ação conforme a entrada dos dados, por exemplo, a entrada de um novo valor de pressão deverá atualizar o gráfico de pressão para a visualização do usuário. Assim quando o usuário preenche as informações do equipamento estas deverão estar no log do sistema. O fluxo de dados do *software* é apresentado na figura 13.

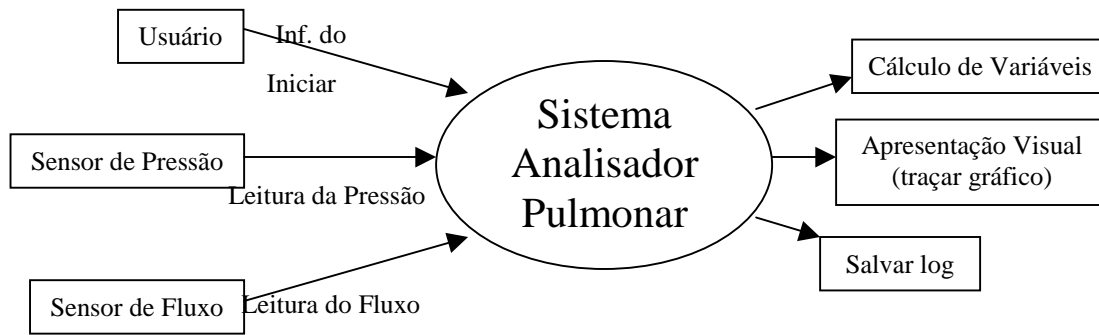


Figura 13 – DFD de nível 0 do sistema proposto



## 2.5 ESPECIFICAÇÃO DE VALIDAÇÃO DO PROJETO

A fase de testes de um projeto é de vital importância na realização deste. Através de testes no circuito que se está implementado é possível a análise e melhoria do mesmo. Como o projeto proposto é dividido em vários blocos funcionais, descrito no capítulo 2, a realização dos testes deverá ocorrer a partir do desenvolvimento de cada etapa. Os testes que deverão ser realizados por bloco funcional são apresentados na tabela 5.

Tabela 5 – Atividades de teste do sistema proposto

Fase do Projeto	Descrição do Teste	Resultados Esperados
Sensores	Deverá ser efetuado testes com relação a variação do sensor apenas na grandeza a ser analisada sem interferência por outros fatores como temperatura.	Os sensores se comportarão de forma ideal, respondendo apenas estímulos da grandeza a ser mensurada
Tratamento do Sinal	Teste de linearidade da saída e relação entre a saída e a grandeza a ser medida. Deve-se tornar a tensão em uma relação mais próxima possível a grandeza medida	Após o tratamento, o sinal de saída dos dois sistemas será linear correspondente a grandeza mensurada
Conversão A/D	Na conversão deverão ser efetuados testes de multiplexagem de entradas, tempo de conversão e saída digital	Multiplexagem entre entradas ideal, com tempo de conversão baixo e sem ruído nas saídas.
Transmissão/ Recepção	Início da fase de <i>software</i> , testes de recepção de informações sem perda de dados, tanto pela transmissão quanto por <i>buffer overflow</i>	Não deverá haver problemas com ruído em pequenas distâncias. Também o <i>software</i> deverá precaver-se de <i>buffer overflow</i>
<i>Software</i>	Deverão ser efetuados testes no <i>software</i> para o cálculo de grandezas não medidas através de sensores. Além disso, deverá ser testada a rotina de salvamento em log.	<i>Software</i> implementado sem problemas com relação a ponto flutuante (cálculos) e arquivos gerados com sucesso.

Quando todos os módulos estiverem desenvolvidos, terá início a fase de teste em conjunto a qual já se poderá verificar a exatidão do sistema desenvolvido. Desta exatidão dependerá o funcionamento do equipamento, pois como dito anteriormente em diversas ocasiões, este sistema deverá analisar equipamentos de missão crítica ao ser humano.

## 2.6 MÓDULO EXTRA

O módulo extra a ser desenvolvido tem o intuito de prover uma forma de alerta a pessoa que está monitorando o equipamento. Através da especificação de parâmetros limites de cada uma das grandezas monitoradas. Além disso ficará gravado todo o formato de onda apresentado em tela da grandeza que ultrapassou o valor máximo especificado. O fluxograma do módulo extra proposto é apresentado na figura 14.

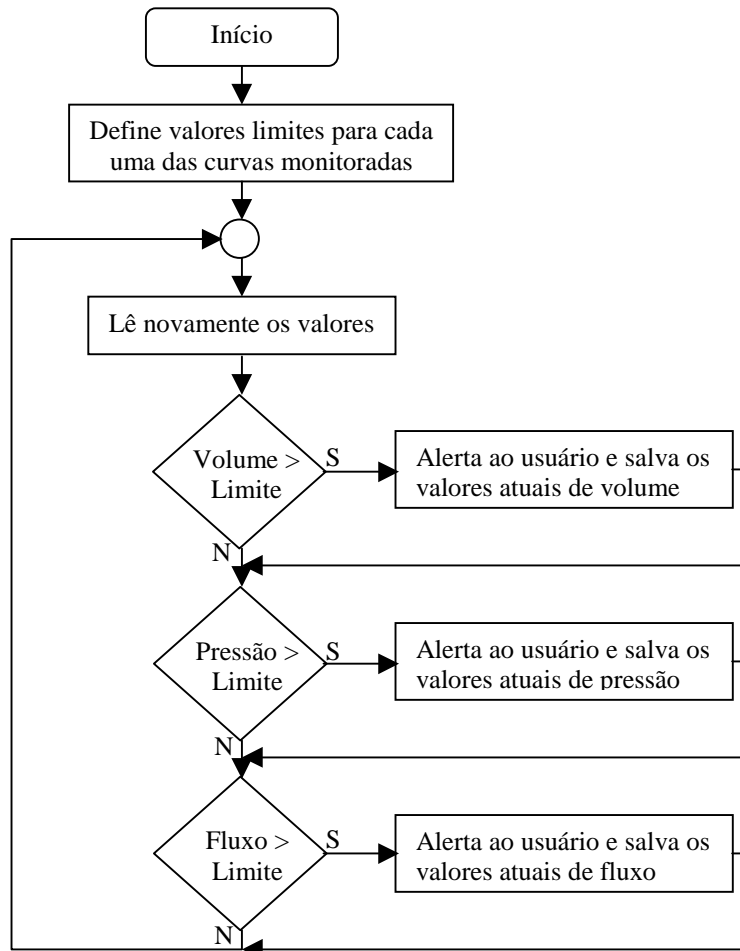


Figura 14 – Fluxograma do módulo extra a ser desenvolvido

## 3. PROJETO

Nesta etapa de projeto descreveremos detalhadamente, o projeto baseado na especificação definida na etapa anterior.

### 3.1 *HARDWARE*

Como desenvolvido durante a especificação, o trabalho foi dividido em dois módulos maiores, que são a etapa de *software* e a de *hardware*. Este último módulo, que será descrita agora, tem o intuito de transformar algumas das diversas grandezas físicas que se fazem presente no processo respiratório em uma forma possível de ser lida pelo computador.

A primeira etapa de todo esse processo é iniciada com a conversão da grandeza física qualquer para uma grandeza elétrica, normalmente a tensão. Essa conversão é realizada por dispositivos conhecidos como sensores.

#### 3.1.1 Sensores

Os sensores tem uma grande importância no projeto em si, pois é através destes que será possível a aquisição dos sinais necessários. Desta forma, visou-se a utilização de componentes com graus de precisão dentro das normas referenciadas na especificação. São utilizados no projeto dois sensores diferentes, um para aquisição da pressão e outro para o fluxo.

##### 3.1.1.1 Sensor de Pressão

Como sensor de pressão foi utilizado o Motorola MPX50D. Este sensor de pressão é desenvolvido como uma ponte de Wheatstone sendo utilizado como elemento sensor um *strain gauge* (extensômetro).

A partir dos dados acima, pode-se constatar que o sensor tem uma saída diferencial variável conforme a pressão aplicada. Para a ligação deste com o circuito é utilizado um amplificador de instrumentação, tendo este como principais funções, a retirada de ruídos de

modo comum e a transformação de um sinal diferencial em um sinal referente ao terra do circuito. O esquema de ligação entre o sensor e o amplificador de instrumentação é apresentado na figura 15.

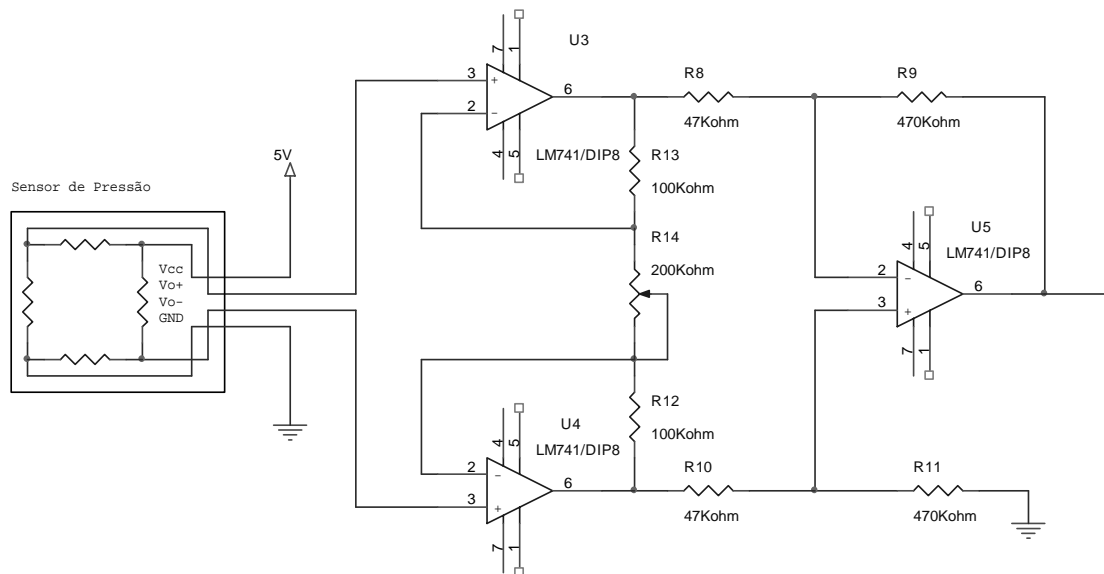


Figura 15 – Conexão do Sensor de Pressão ao Amplificador de Instrumentação

Este circuito tem um ganho variável de 1 até 20 vezes o valor de entrada do sensor, este estágio tem esta pequena amplificação para que seja possível a atenuação do ruído.

### 3.1.1.2 Sensor de Fluxo

Para a utilização do sensor de fluxo, inicialmente, foi cogitado os dispositivos da série AWM700, que são sensores ideais para o uso neste aplicação. Visto que o custo de compra deste dispositivo era muito alto, optou-se por uma opção alternativa.

Foi utilizado para elemento sensor de fluxo, um dispositivo utilizado em equipamentos de anestesia em hospitais. Sendo assim, não tem-se dados conhecidos sobre a sua estrutura, sabe-se apenas que ele trabalha como um transdutor de rotação, fazendo com que o sinal de saída tenha frequência proporcional ao fluxo de ar passante.

Tendo em vista que não é possível converter a saída em frequência para um valor binário, compreensível ao computador, e também por ser mais difícil a alteração da frequência de sinais, A etapa de entrada faz a conversão frequência-tensão através de um conversor comercial, LM2907.

O esquema de desenvolvimento da etapa de entrada para o sensor de fluxo é apresentado na figura 16.

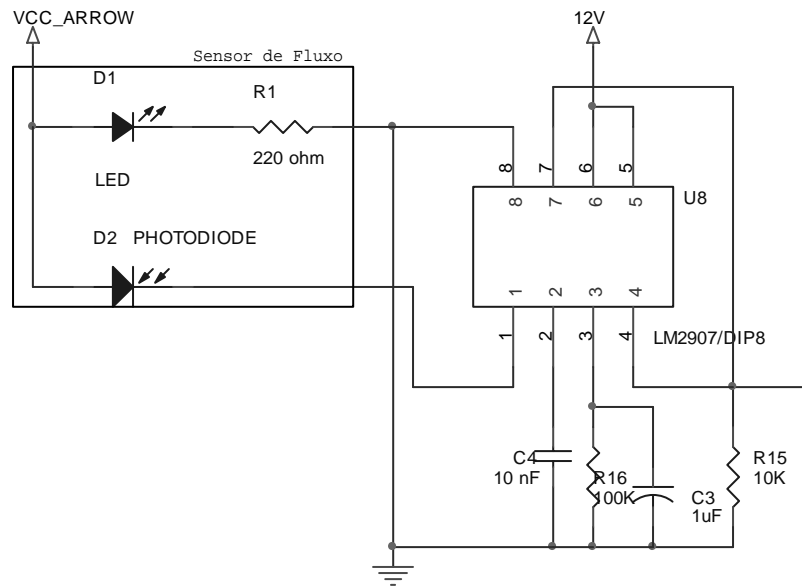


Figura 16 – Etapa de Entrada do Sensor de Fluxo

Como não se sabe ainda a variação de frequência em relação ao fluxo, visto que o sensor é desconhecido. Este esquema tem uma saída de 1 volt para cada 67 Hz de variação na frequência, conforme as análises a serem realizadas durante a fase de implementação, esta variação poderá ser alterada, bastando para isso alterar o valor do capacitor C4.

Após este módulo é necessário o tratamento do sinal.

### 3.1.2 Tratamento do Sinal

O módulo de tratamento do sinal tem como principal função a linearização das curvas, igualando o máximo valor da grandeza física com o maior valor binário possível ao conversor, isto é, a máxima saída da etapa do sensor deverá ser igual ao valor de referência positiva do conversor.

Além disso, em alguns projetos, o tratamento do sinal constitui a fase de retirada de sinais indesejados, através de filtros de sinais. Esta metodologia não deverá ser necessária, visto que normalmente os sinais de saída não deverão ser constituídos de diversas frequências.

No projeto, é utilizado como elemento para o tratamento do sinal, um amplificador operacional montado de forma não-inversora, com seu ganho variável entre 1 até 20 vezes

o valor do sinal original, permitindo com isso o reajuste do equipamento. O circuito do amplificador não-inversor é mostrado na figura 17.

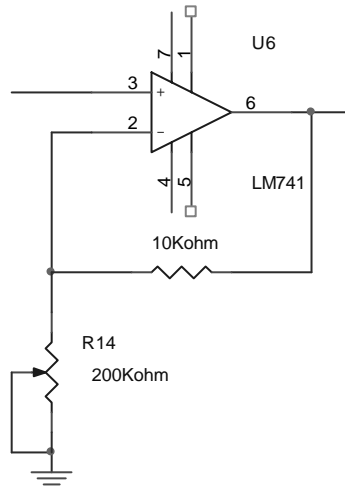


Figura 17 – Amplificador não-inversor

Após esta fase, o sinal já foi adquirido e tratado, tornando ideal para o próximo módulo, que é a conversão A/D.

### 3.1.3 Conversão A/D

O módulo de conversão A/D tem grande importância no projeto visto que é a única maneira de interfaceamento entre os módulos de entrada de dados e o computador. A partir do momento que o tratamento do sinal foi realizado e a tensão, correspondente a grandeza física a ser medida, estiver variando entre as tensões de referência, tanto positiva quanto negativa, será possível a conversão desta tensão dentro da escala de bits do ADC (*Analogic-Digital Converter* / Conversor Analógico-Digital). Neste módulo figura-se com grande destaque o ADC definido para realizar esta operação.

Como conversor A/D será utilizado o ADC0808, visto que preenche os principais requisitos como: velocidade e precisão. Além disto, tem uma característica que torna-o muito interessante ao projeto, que é a possibilidade de se conectar até oito entradas de sinais diferentes. Desta forma é necessário apenas a conexão de cada um dos dispositivos em uma entrada diferente que o próprio ADC efetuará a multiplexação dos sinais. Para a definição de qual dos dois sinais será convertido em um dado momento, será utilizado um bit de controle da porta paralela, este por sua vez será controlado por *software*.

O esquemático da conversão A/D é apresentado na figura 18. Para que o conversor possa funcionar adequadamente é necessário um sinal de 10KHz em seu pino 10, este sinal é gerado a partir de um circuito astável utilizando um 555. Este circuito de *clock* para o ADC é apresentado na figura 19.

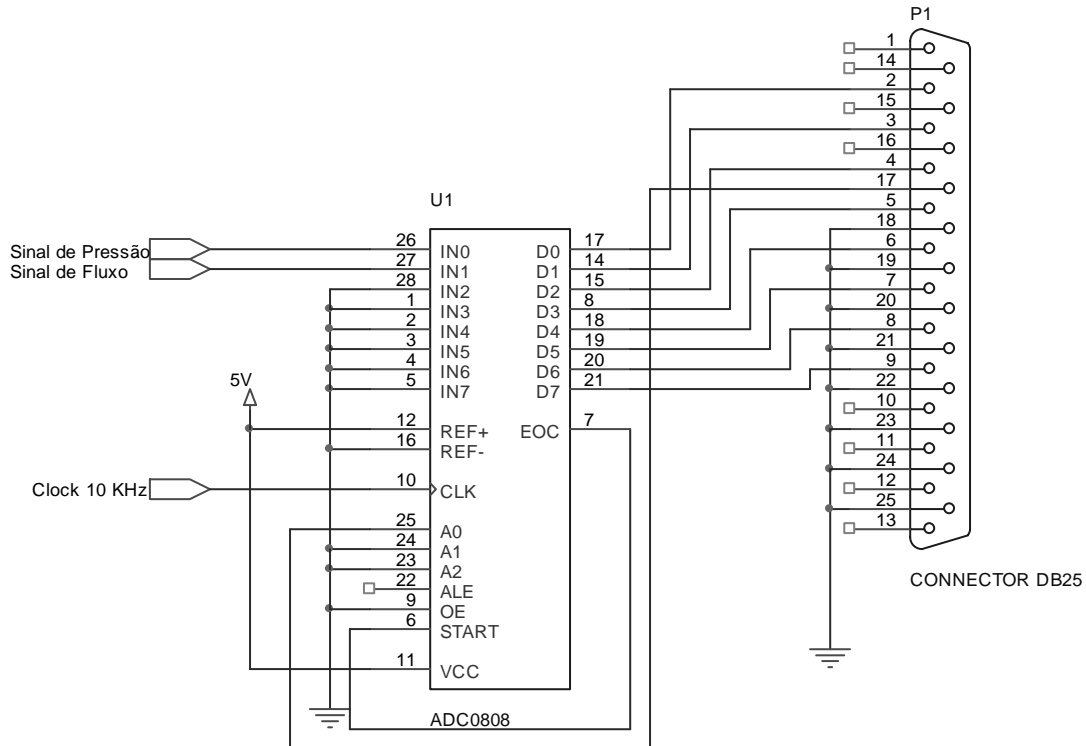


Figura 18 – Esquema de ligação do circuito de conversão A/D

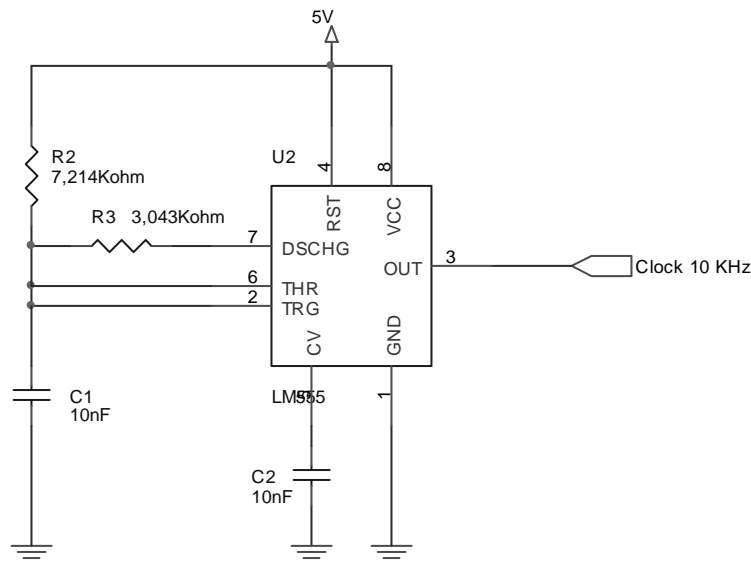


Figura 19 – Gerador de *clock* de 10 KHz para o ADC0808

### 3.1.4 Transmissão

Como definido durante a especificação a transmissão consistirá apenas da conexão entre o ADC e a interface paralela do computador, visto que os dois trabalham com o mesmo número de bits, não havendo qualquer incompatibilidade neste processo.

Inicialmente, foi definido como escopo do projeto os sistemas operacionais Windows 9X/ME. Esses sistemas permitem o acesso direto aos endereços de memória, realizando instruções em *assembler* de escrita e leitura de dados. Desta forma, o acesso a porta paralela se torna mais simples, visto que pode-se utilizar códigos já conhecidos para acesso.

No decorrer do projeto, se o cronograma permitir, deverá se tentar uma implementação também para Windows 2000. Esta implementação será realizada utilizando-se um *device driver*, que são os únicos componentes do sistema que podem ter acesso ao hardware. Para o acesso a porta paralela será utilizado um *device driver freeware*, conhecido como UserPort, ele permite o acesso a qualquer endereço de memória especificado em software.

É necessário que a porta paralela esteja configurada como PS/2, visto que é o único meio de comunicação bidirecional direta da porta, não ocorrendo comunicação nos outros três modos existentes. O software realizará uma verificação logo na inicialização para detectar se a porta está configurada corretamente.



## 3.2 SOFTWARE

O *software* deverá efetuar a parcela de interfaceamento com o usuário, proporcionando ao cliente uma interface mais agradável, é que permita uma melhor observação do equipamento analisado. Além disso, o software deverá permitir que o usuário possa verificar um histórico dos equipamentos já analisados.

Para descrever o projeto de *software* iniciar-se-á descrevendo-se as interfaces de tela, definindo quais informações o usuário poderá enviar e quais retornos ele terá. Após essa descrição serão mostrados os fluxogramas do programa a ser desenvolvido.

### 3.2.1 Itens de Tela

A tela principal do software é mostrado na figura 20, é a partir dela que o usuário poderá interagir com os diversos módulos do software.

Esta é dividida em dois frames principais, um com informações de saída, no caso os gráficos das propriedades, e o outro com entrada das informações e botões de acionamento da monitoração.

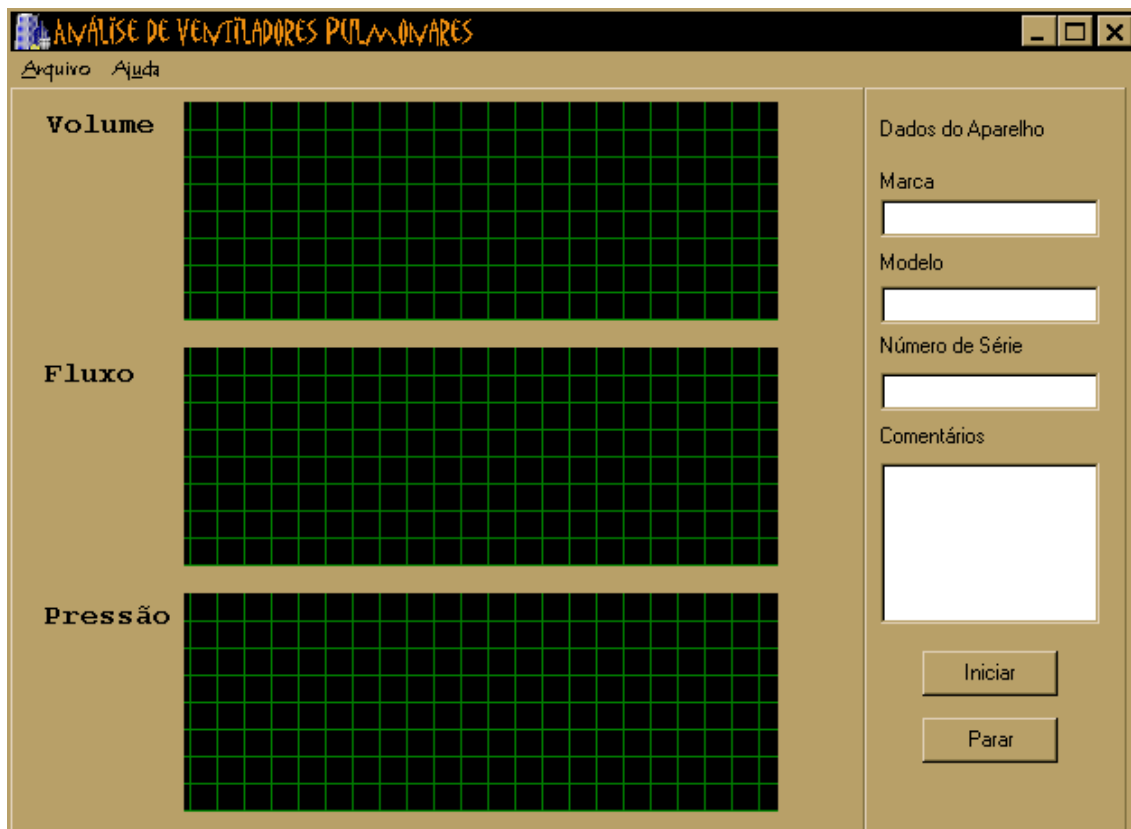


Figura 20 – Tela de apresentação do software

### 3.2.1.1 Frame 1 – Saída das informações

É composto por três gráficos, cada um apresentando a curva de uma propriedade, que serão: fluxo, volume e pressão. Os gráficos terão no eixo das coordenadas a informação de tempo e no eixo das abcissas conterão o valor amostrado pelos sensores ou o valor calculado, tendo uma apresentação em linha. O eixo de tempo conterá trezentos pontos fixos e será atualizado quando forem preenchidos esses trezentos pontos, o eixo das propriedades variará conforme a propriedade, este valor de variação é apresentado na tabela 6.

Tabela 6 – Valores máximo e mínimo para as diversas propriedades

Propriedade	Valor mínimo	Valor máximo	Unidade
Pressão	0	600	mmHg
Fluxo	0	120	l/min
Volume	0	8	l

### 3.2.1.2 Frame 2 – Entrada de Dados

Neste frame ter-se-á os campos que o usuário poderá preencher para ter um log mais completo e fácil de consultar.

Os campos e suas propriedades são apresentados na tabela 7.

Tabela 7 – Campos de entrada de informações sobre o equipamento

Campo	Nome	Tipo de dado	Máximo (caracteres)
Marca	EdtMrc	Alfanumérico	10
Modelo	EdtMdl	Alfanumérico	10
Número de Série	EdtNsr	Alfanumérico	18
Comentários	MmCmt	Alfanumérico	--

Além dos campos de entrada de informações, este frame contém os botões de ação que farão com que o software inicie o ciclo de leitura dos sensores-atualização da tela e interromperão este ciclo. Quando o ciclo é interrompido o software retornará ao cliente, conforme os sinais adquiridos, se o equipamento está sem problemas ou qual parâmetro está descalibrado.

### 3.2.2 Menus

O software conterá apenas dois menus principais: arquivo e ajuda. O menu arquivo conterá as ações que o usuário poderá efetuar, estas ações são apresentadas na tabela 8.

Tabela 8 – Comandos do Menu Arquivo

<b>Ação</b>	<b>Comentário</b>
Iniciar	Deverá desencadear o mesmo ciclo que o botão presente na interface principal.
Parar	Deverá desencadear o mesmo ciclo que o botão presente na interface principal.
Definir Parâmetros	Permitirá ao usuário definir diversos parâmetros necessários a análise do equipamento, através destes será possível uma melhor avaliação.
Verificar Log	Verifica no diretório do software os logs já salvos e apresenta uma tabela com todos, onde o usuário escolherá qual será visualizado.
Sair	Finalizará o programa, realizando a parada do software se ele estiver em execução.

Os parâmetros que poderão ser definidos para a análise do equipamento são, na figura 21, temos um exemplo da tela que capturará esses dados:

- Volume Máximo Alcançado
- Volume Mínimo Alcançado
- Pressão Máxima Alcançada
- Pressão Mínima Alcançada
- Fluxo Máximo Alcançado
- Fluxo Mínimo Alcançado

**PARÂMETROS A SEREM ANALISADOS**

<input type="checkbox"/> Tempo de Respiração	<input type="text"/>	segundos
<input type="checkbox"/> Tempo de Inspiração	<input type="text"/>	segundos
<input type="checkbox"/> Tempo de Expiração	<input type="text"/>	segundos
<input type="checkbox"/> Volume Máximo	<input type="text"/>	litros
<input type="checkbox"/> Volume Mínimo	<input type="text"/>	litros
<input type="checkbox"/> Pressão Máxima	<input type="text"/>	cmH2O
<input type="checkbox"/> Pressão Mínima	<input type="text"/>	cmH2O
<input type="checkbox"/> Fluxo Máximo	<input type="text"/>	litros/minuto
<input type="checkbox"/> Fluxo Mínimo	<input type="text"/>	litros/minuto

OK Cancel

Figura 21 – Tela de definição de parâmetros para Análise

O menu de ajuda conterá dois comandos, como pode ser visto na tabela 9:

Tabela 9 – Comandos do Menu Ajuda

Ação	Comentário
Índice	Conterá um help básico de como deve ser utilizado o software e suas principais telas.
Sobre	Tem as informações de versão e autor do software.

### 3.2.3 Arquivos de Log

Como dito anteriormente, o software deverá efetuar um log em cada análise que ele realizar. O formato de nome do arquivo de log está apresentado na tabela 10, já na tabela 11 teremos como serão gravadas as informações neste arquivo.

Tabela 10 – Definição do padrão de nome dos arquivos de log

Posição	Informação	Tamanho (caracteres)	Comentário
1°	Modelo	0-10	Modelo do equipamento informado pelo usuário (opcional)
2°	Núm. de Série	0-10	Número de série do equipamento informado pelo usuário (opcional)
3°	Data	8	Data do momento em que foi iniciado o ciclo de análise
4°	Hora	4	Hora do momento em que foi iniciado o

			ciclo de análise
5 <sup>o</sup>	.LAVP	5	Extensão do arquivo, com valor fixo, para definir como arquivo de log do software

Tabela 11 – Formato de gravação das informações no arquivo de log

<b>Linha</b>	<b>Informação</b>	<b>Intervalo (caracteres)</b>
1 <sup>a</sup>	*	1-50
2 <sup>a</sup>	Marca	0-10
2 <sup>a</sup>	Modelo	11-20
2 <sup>a</sup>	Número de Série	21-38
3 <sup>a</sup>	Comentários	1-50
4 <sup>a</sup>	Comentários	1-50
5 <sup>a</sup> -até o final do arquivo	Pressão	1-3
5 <sup>a</sup> -até o final do arquivo	Volume	4-6
5 <sup>a</sup> -até o final do arquivo	Fluxo	7-9
5 <sup>a</sup> -até o final do arquivo	Hora	10-17

Junto com o arquivo de log, vai ser gravado um arquivo de erro que armazenará os valores que excederam o limite proposto pelo usuário, este arquivo terá diversas semelhanças com o arquivo de log, sendo que seu nome é o mesmo apresentado na tabela 5 com a alteração da extensão para ERR. Também as linhas de informação do equipamento serão mantidas, isto se deve a necessidade de se consultar o arquivo de erro sem verificar o arquivo de log para saber qual o equipamento. Os campos que serão gravados no arquivo de erro são apresentados na tabela 12.

Tabela 12 – Formato de gravação das informações no arquivo de erro

<b>Linha</b>	<b>Informação</b>	<b>Intervalo (caracteres)</b>
1 <sup>a</sup>	*	1-50
2 <sup>a</sup>	Marca	0-10
2 <sup>a</sup>	Modelo	11-20
2 <sup>a</sup>	Número de Série	21-38
5 <sup>a</sup> -até o final do arquivo	Grandeza física	1-6
5 <sup>a</sup> -até o final do arquivo	Valor atingido	7-10
5 <sup>a</sup> -até o final do arquivo	Valor limite estipulado	11-14
5 <sup>a</sup> -até o final do arquivo	Hora	10-17

### 3.2.4 Ciclo de execução

O software se baseia em um loop, de início e término definidos pelo usuário, de aquisição, do arquivo de log e a apresentação visual ao usuário, como pode ser visto na figura 22.

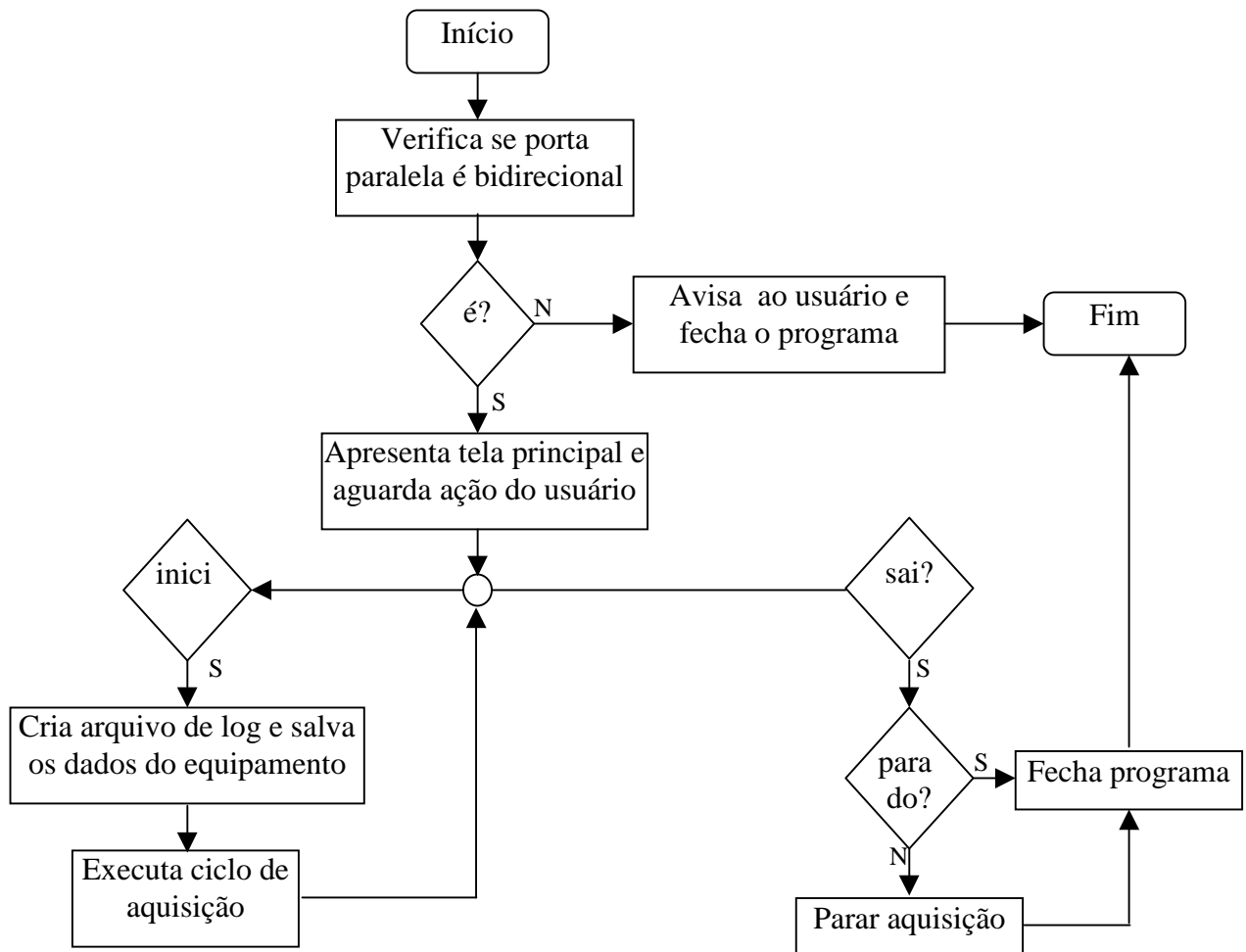


Figura 22 – Ciclo de Processamento do Software

O ciclo de aquisição mostrado dentro do fluxograma acima, tem como tarefas a definição de qual sensor vai ser lido; a recepção do dado requerido enviado pelo conversor A/D; os cálculos das variáveis não medidas, como o fluxo, tempo de inspiração e expiração; a atualização dos diversos gráficos pertencentes a interface de saída; e a comparação com os valores limite para as diversas grandezas, salvando em arquivo se estes foram ultrapassados, para uma melhor compreensão de todas as tarefas que devem ser feitas o ciclo de aquisição é descrito no fluxograma apresentado na figura 23.

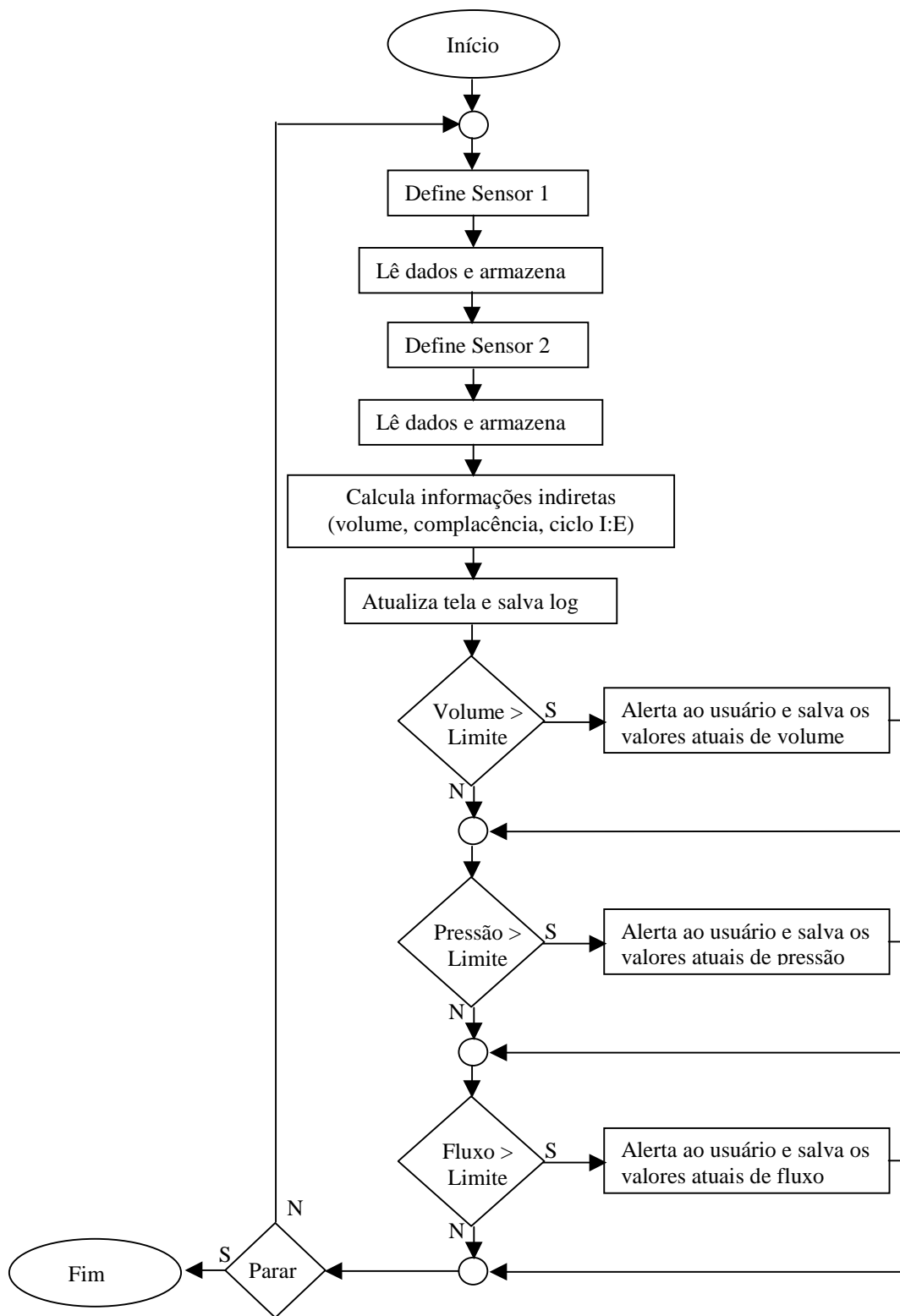


Figura 23 – Fluxograma do Ciclo de Aquisição

## 4. IMPLEMENTAÇÃO

Nesta fase do projeto, serão descritos os resultados obtidos no decorrer da execução da implementação do projeto.

Serão apresentados aqui as curvas obtidas na análise de cada sensor, os testes com o conversor, os circuitos desenvolvidos e o código do *software*.

Apresentar-se-á a implementação, seguindo a ordem acima proposta, não necessariamente seguindo os blocos funcionais dispostos na primeira seção.

### 4.1 CURVA DOS SENSORES E TRATAMENTO DO SINAL

O primeiro passo para a utilização de sensores para aquisição de sinais de grandeza físicas é o levantamento da curva de resposta. Isto é, deve-se verificar a faixa de amplitude de resposta em relação a grandeza aplicada e a sensibilidade do sensor.

Com estes dados dispostos inicia-se a fase de desenvolvimento do *hardware* para tratamento deste sinal e, com isto, deixa-lo em uma faixa de amplitude na qual necessita-se trabalhar.

Por último, deve-se refazer novamente a curva de resposta em relação a todo o circuito desenvolvido, desta forma, pode-se verificar a confiabilidade do circuito desenvolvido.

Na sequência são apresentados os resultados decorrentes da aplicação deste três passos para o desenvolvimento do sistema. Será apresentado primeiro os dados obtidos em relação ao sensor de pressão e subseqüentemente os dados do sensor de fluxo.

#### 4.1.1 Sensor de Pressão

Como dito na seção de projeto foi escolhido como sensor de pressão o MXP50D da Motorola.

Este sensor tem como característica principal ser de pressão diferencial. Com isto, uma das entradas foi deixada em aberto, para que as medidas de pressão fossem realizadas relativas a pressão ambiente.



Na sequência, foi realizado uma avaliação com o intuito de desenhar a curva de resposta do sensor em relação a pressão, esta curva é apresentada na tabela 13, a curva também é apresentada na figura 24 para uma melhor compreensão de suas características.

Tabela 13 – Relação pressão X saída do sensor

Pressão (cmH <sub>2</sub> O)	Saída (mV)
0	46,5
5	47,2
10	47,9
15	48,7
20	49,5
25	50,2
30	51
35	51,7
40	52,5
45	53,2
50	54,1
55	54,8
60	55,9

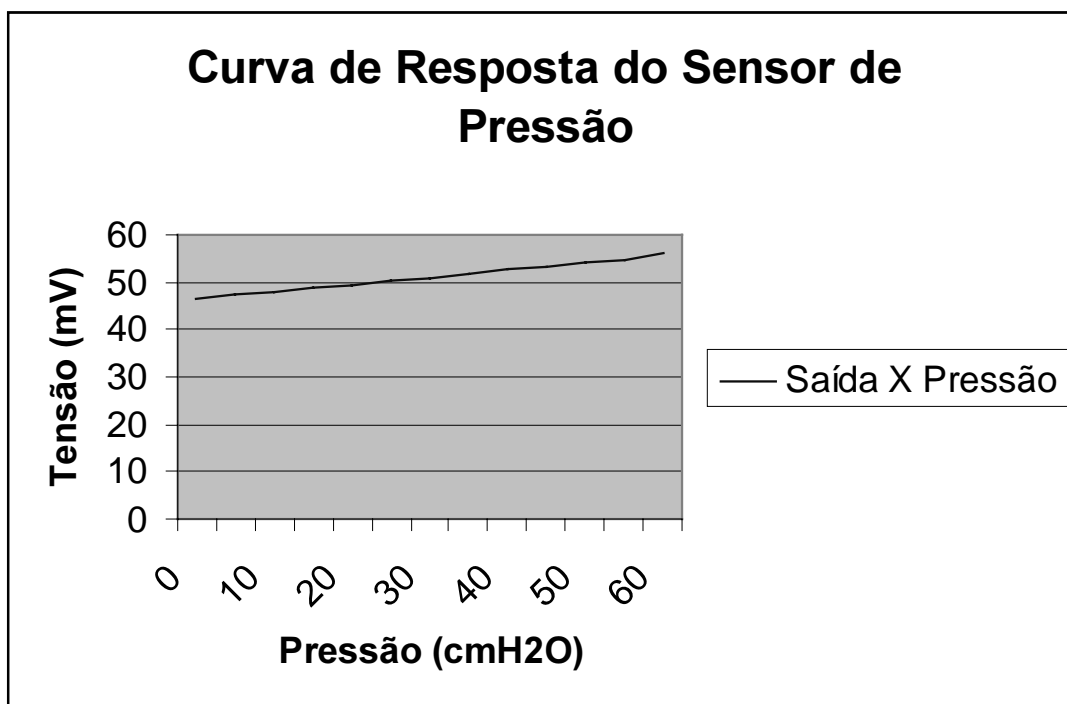


Figura 24 – Curva de resposta do Sensor de Pressão

Primeiramente foi desenvolvido um amplificador de instrumentação, pois o sinal de saída do sensor era flutuante, isto é, tinha uma saída sem o referencial de terra, na sequência foi implementado um somador para a retirada do sinal de *offset*.

O sinal de *offset* é a resposta quando não se está aplicando pressão nenhuma. Com a utilização de um somador com uma tensão de igual amplitude mas inversamente proporcional ao sinal de *offset* se terá uma saída igual a zero quando não houver pressão aplicada.

O intuito do circuito foi ser o mais maleável possível, desta forma, foram implementados os ganhos e ajustes de *offset* com *trimpots*, permitindo-se assim, que a calibração seja feita rapidamente e, com isto, tenha-se uma melhor performance do sistema.

O circuito implementado está em anexo. Com todos os passos acima descritos implementados, pode-se, novamente, refazer a curva do sinal de tensão em relação a pressão exercida, esta curva é apresentada na figura 25. Os valores apresentados no gráfico estão dispostos na tabela 14 para que possa-se verificar todos os valores.

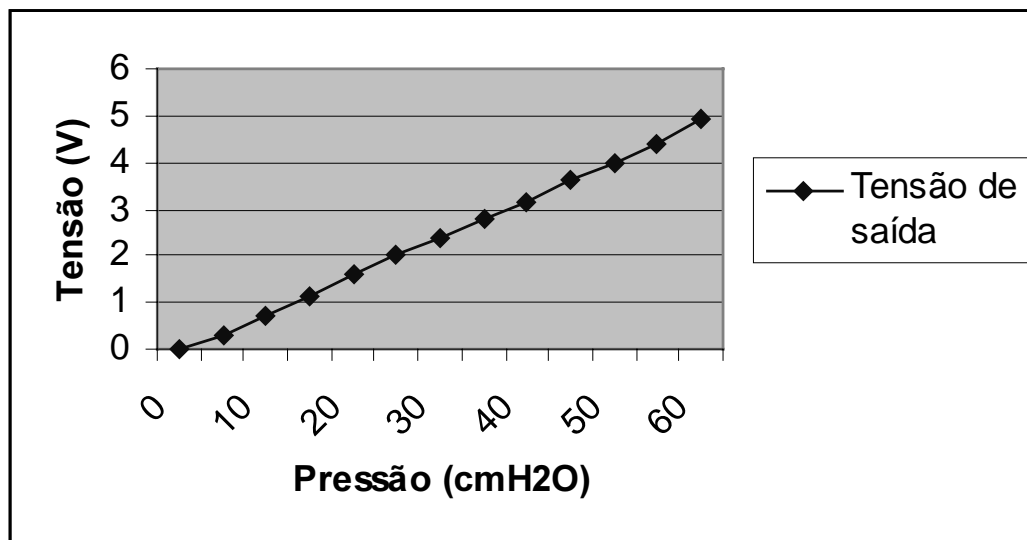


Figura 25 – Curva de resposta do circuito de tratamento do sinal de Pressão

Tabela 14 – Valores do sinal do circuito

Pressão (cmH <sub>2</sub> O)	Tensão (V)
0	0
5	0,308
10	0,6886
15	1,144
20	1,5994
25	2,0108
30	2,3474
35	2,816
40	3,1724
45	3,6014
50	3,971
55	4,3824
60	4,9324

Após a implementação do circuito para análise da pressão foi implementado a conversão do fluxo.

#### 4.1.2 Sensor de Fluxo

O sensor de fluxo utilizado foi cedido pelo Hospital de Clínicas e se caracteriza por ser um sensor do tipo turbina, com isto, tem-se um sinal de saída em frequência conforme o fluxo aplicado sobre o sensor.

Foi definido como limite superior da escala 120 SLPM, e, com isto, foi necessário utilizar as dependências do HC para caracterização da curva de resposta do sensor, assim como, o auxílio de um *software* analisador que traçava a curva de fluxo.

Foi definido um intervalo de escala de 5 SLPM para caracterização da curva de resposta. Este valor foi definido para que se tivesse uma boa precisão na curva de resposta e ao mesmo tempo não fosse necessário um grande número de medições.

Os valores de frequência encontrados para o sensor utilizado são apresentados na tabela 15, já a curva de resposta é apresentada na figura 26.

Tabela 15 – Valores de resposta do sensor de fluxo

Fluxo (SLPM)	Frequência Sensor (Hz)
0	0
5	25,15
10	70,11
15	114,2
20	137,1
25	172
30	211,6
35	248,1
40	285,3
45	330,2
50	356
55	385,5
60	431,9
65	468,5
70	506,8
75	543,1
80	585
85	622,3
90	652
95	674
100	723
105	758
110	796
115	824
120	855

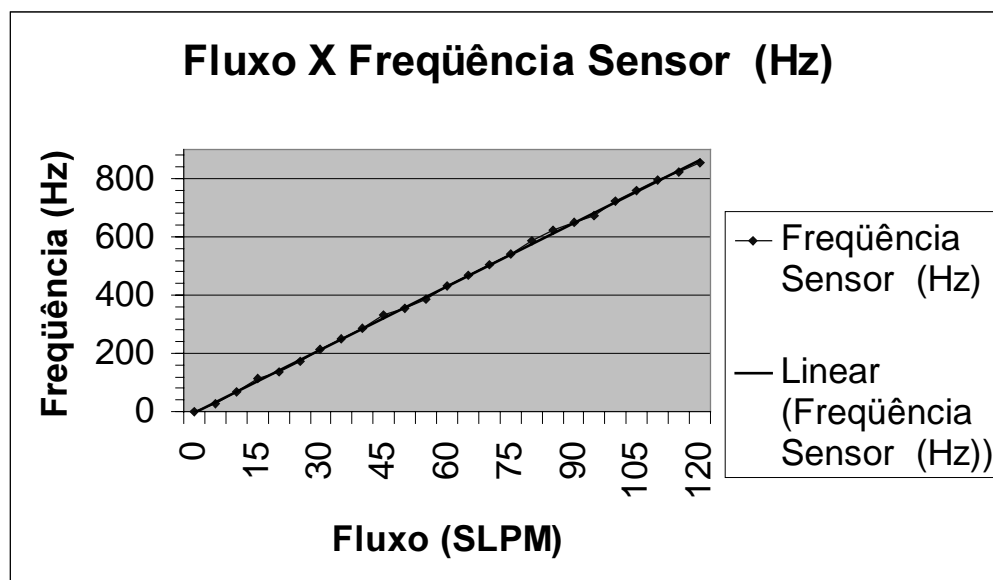


Figura 26 – Curva de resposta do sensor de fluxo

Como definido na seção de projeto foi utilizado um conversor frequência/tensão para que se pudesse utilizar o sinal obtido pelo sensor da forma desejada, fazendo com que ele pudesse ser convertido pelo ADC e captado pelo computador.

O LM2907 é um componente que tem sua faixa de atuação variável conforme a tensão de alimentação e alguns componentes analógicos, resistores e capacitores. Para se calcular os valores de resposta deve se utilizar a seguinte equação:

$$V = f * V_{cc} * R * C$$

Para se definir o valor de R, utilizou-se:

V = 5 Volts, tensão máxima para o ADC

F = 855 Hz, a frequência do maior valor da escala

V<sub>cc</sub> = 9 Volts, tensão de alimentação

C = 10 nF, valor fixo do capacitor.

Com estes valores, pode-se obter R = 56 KΩ. O circuito implementado para tal conversão é apresentado em anexo.

Após a implementação do circuito, foi aferido os seus valores de saída em relação ao fluxo, estes valores são apresentados na tabela 16 e sua curva é apresentada na figura 27.

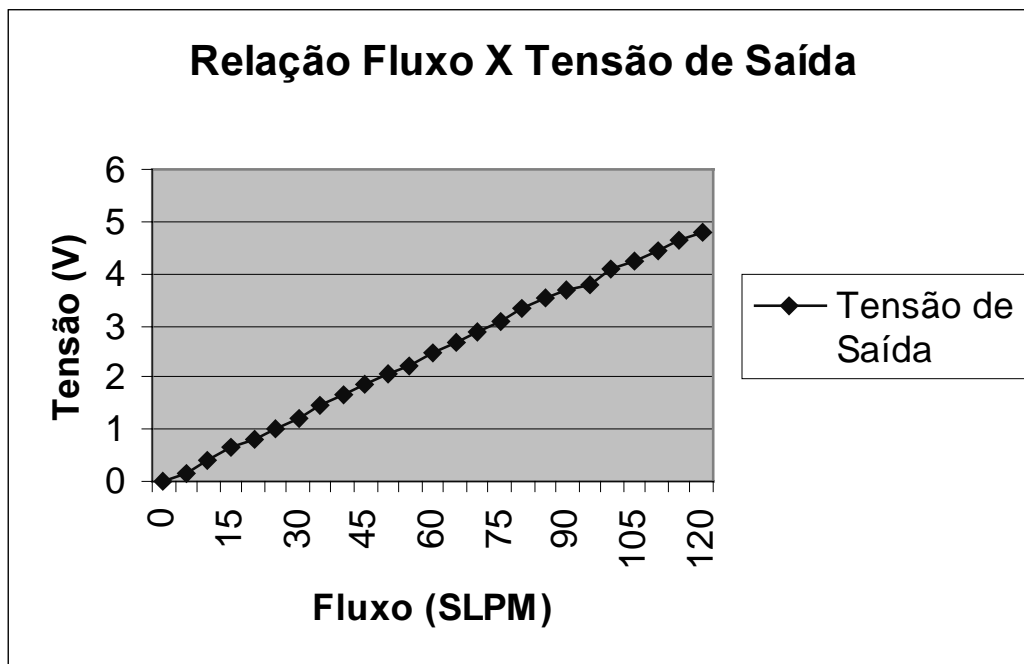


Figura 27 – Curva de resposta do circuito conversor Frequência/Tensão

Tabela 16 – Valores obtidos para o circuito conv. F→ T

Fluxo (SLPM)	Tensão Saída (V)
0	0
5	0,164
10	0,422
15	0,668
20	0,8
25	1
30	1,23
35	1,44
40	1,64
45	1,89
50	2,05
55	2,2
60	2,46
65	2,66
70	2,88
75	3,07
80	3,31
85	3,53
90	3,66
95	3,79
100	4,06
105	4,26
110	4,45
115	4,62
120	4,78

Com estes dois circuitos pode-se efetuar a aquisição dos parâmetros necessários para a análise do aparelho.

Na seqüência para se realizar o processamento no computador é necessário a sua conversão e amplificação, este passo é apresentado na seqüência.

## 4.2 CONVERSÃO A/D E TRANSMISSÃO

Para a conversão A/D foi implementado o mesmo circuito apresentado na seção projeto com a alteração do circuito de *clock*.

O circuito de *clock* anteriormente proposto ocasionava uma realimentação nos valores de referência do ADC, desta forma o conversor A/D tinha um sinal de saída que não correspondia ao sinal esperado.

Para a correção deste problema foi implementado um circuito de *clock* através de um cristal, fazendo com que o sinal fosse muito mais confiável. O diagrama esquemático do circuito implementado é apresentado na figura 28.

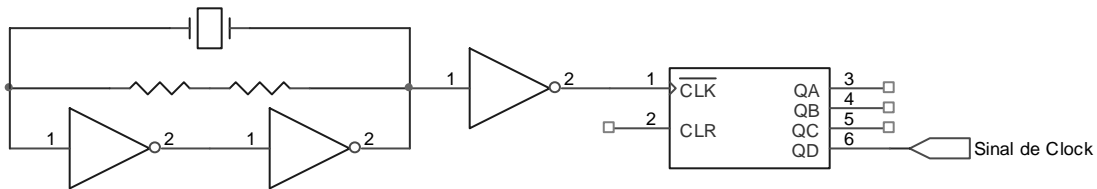


Figura 28 – Circuito de clock a cristal implementado

Após a implementação deste circuito foram realizados novos testes e verificou-se a resolução do problema.

O *clock*, anteriormente com 10 KHz, foi alterado para uma frequência de 380 KHz aproximadamente, o que torna a conversão mais rápida.

A transmissão foi implementada da forma prevista não ocorrendo nenhum problema quanto a isto.

## 4.3 SOFTWARE

Como anteriormente definido, o software foi implementado em C++ Builder.

Foi adicionado ao menu principal uma forma de o usuário poder verificar seus arquivos de log anteriormente gravados, tornando, desta forma, mais amigável a visualização das análises anteriormente realizadas. A tela de visualização de logs é apresentada na figura 29.

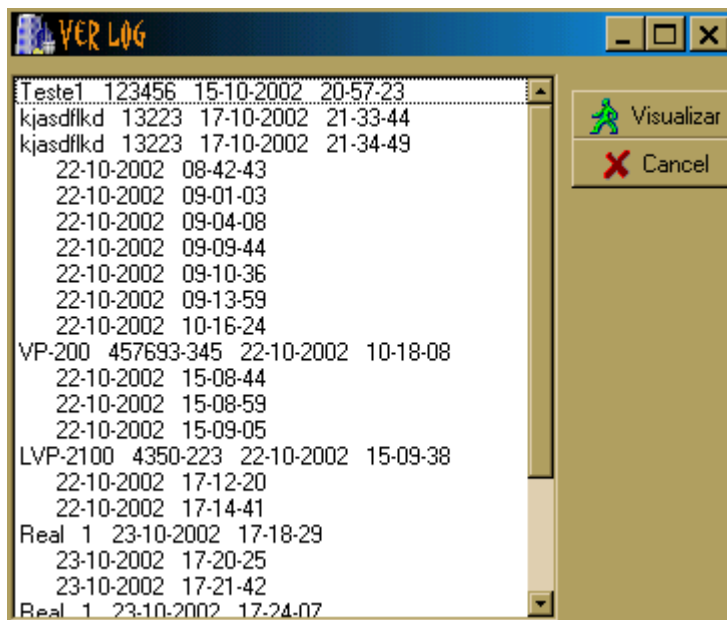


Figura 29 – Tela para seleção de log a ser visualizado

Na figura 30 é apresentado um arquivo de log anteriormente feito, com isto o usuário poderá rever a análise de aparelhos para detecção de problemas.

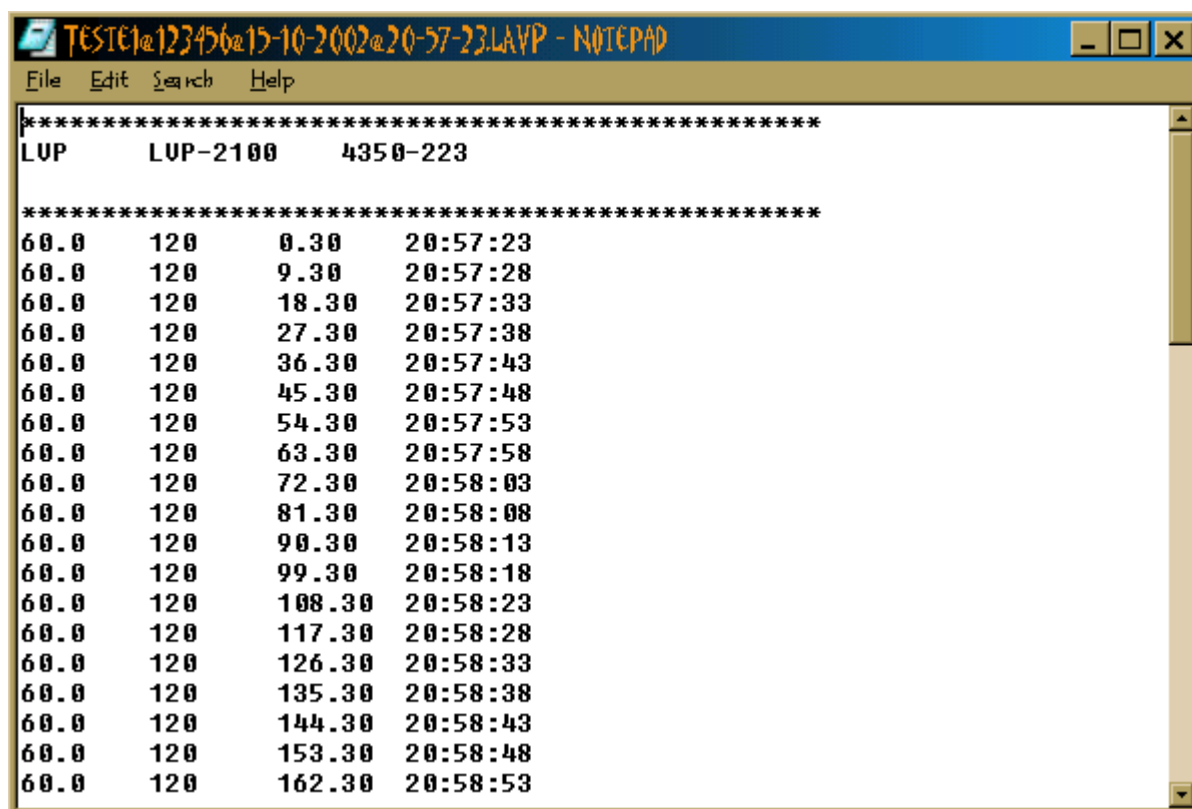


Figura 30 – Tela de um log anteriormente gravado



Além disto, também é apresentada ao final da análise, uma tela resumo com os principais dados obtidos e com o número de vezes em que o aparelho ultrapassou os limites estipulados, desta forma, o usuário poderá tomar medidas corretivas.

Também foram desenvolvidas três classes especiais. Uma para comunicação com a porta paralela tendo portabilidade para sistemas WIN9X, NT e 2000. E as outras duas para funcionarem na análise do ventilador. A análise destas três classes são apresentadas em anexo.

Como o software é desenvolvido em C++ Builder, e este compilador se utiliza de inúmeros arquivos para manutenção dos objetos e eventos, não é apresentado o código-fonte desenvolvido.

## 5. RESULTADOS

Nesta seção, serão descritos os resultados obtidos com a partir da realização dos testes de validação e da utilização do projeto junto a um respirador pulmonar.

### 5.1 TESTES DE VALIDAÇÃO

Os testes de validação foram realizados em conjunto à implementação do projeto, visto que, estes compreendiam verificações de módulos isolados.

Os resultados obtidos a partir da realização dos testes foram os já previstos durante a fase de especificação; as curvas dos sensores eram de grande linearidade permitindo uma menor implementação ao módulo de tratamento de sinal; o conversor A/D teve uma boa taxa de resposta em relação a multiplexação de entradas e valores na saída; no software, não ocorreu nenhum problema de ponto flutuante permitindo uma grande aproximação dos valores calculados para os valores reais.

Visto todos estes pontos foi definido um teste de utilização do aparelho *in vivo* em um ventilador Bennett MA-1 que será detalhado na sequência.

### 5.2 TESTE DE UTILIZAÇÃO

Definiu-se a realização de um teste utilizando-se o AVP e um ventilador pulmonar para verificação de suas curvas durante o processo de ciclagem do ventilador. Para isto o AVP foi conectado a um respirador Bennett MA-1 do Hospital de Clínicas de Curitiba.

Foi iniciado o processo de ciclagem com os parâmetros mostrados na tabela 17.

Tabela 17 – Parâmetros definidos

Parâmetro	Valor	Unidade
Fluxo	80	SLPM
Volume	1,2	l
Pressão	19	cmH <sub>2</sub> O

Em seguida, foi conectado ao ventilador o AVP e este ao PC. Foi inicializado o software. Definiu-se os dados de entrada, marca, modelo, etc, e iniciou-se a aquisição dos

dados no software. A tela principal com as curvas de resposta obtidas pelo software são apresentadas na figura 31.

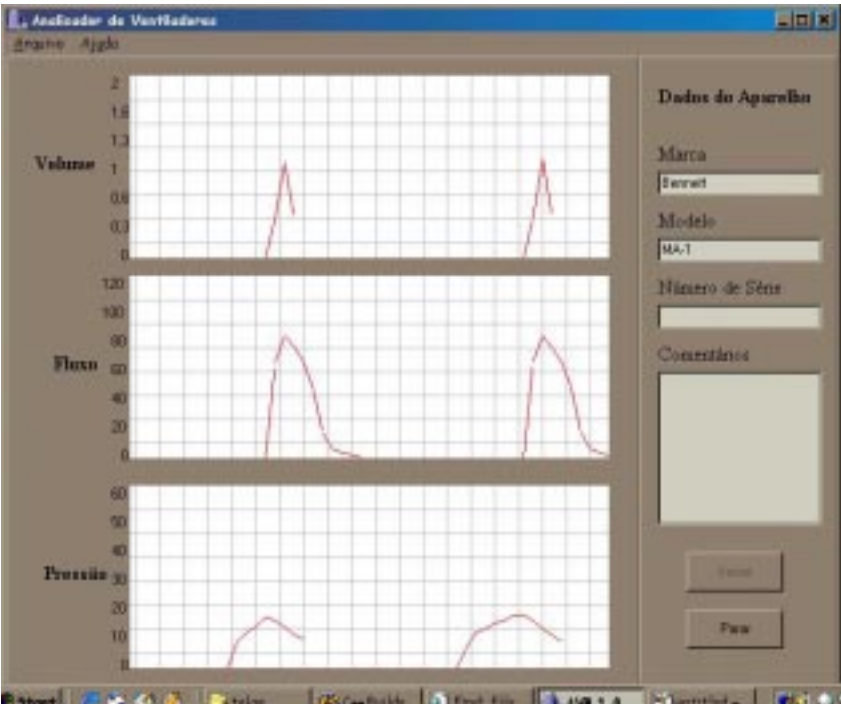


Figura 31 – Curvas obtidas pelo AVP

Como pode-se verificar nas curvas apresentadas os valores de fluxo, volume e pressão definidos no ventilador são os picos de máximo das curvas. Esta afirmação fica ainda mais visível se verificarmos o relatório final do AVP que é apresentado na figura 32, seus valores também são tabulados na tabela 18.

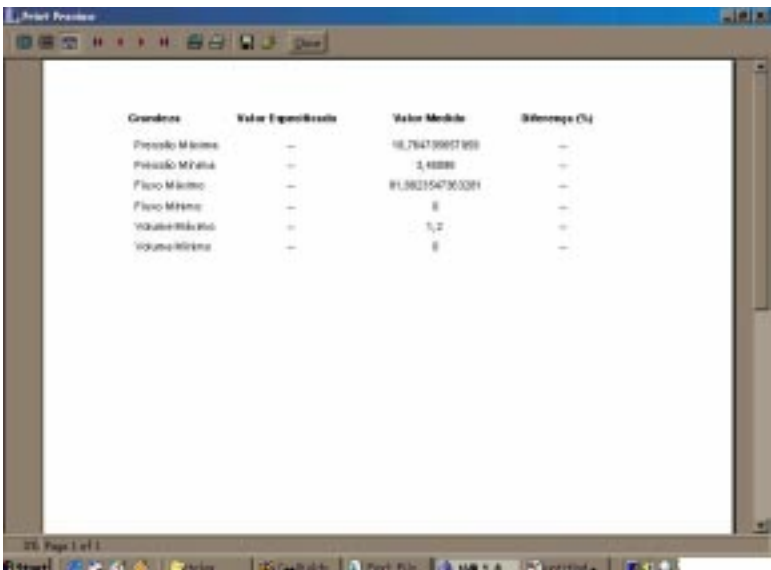


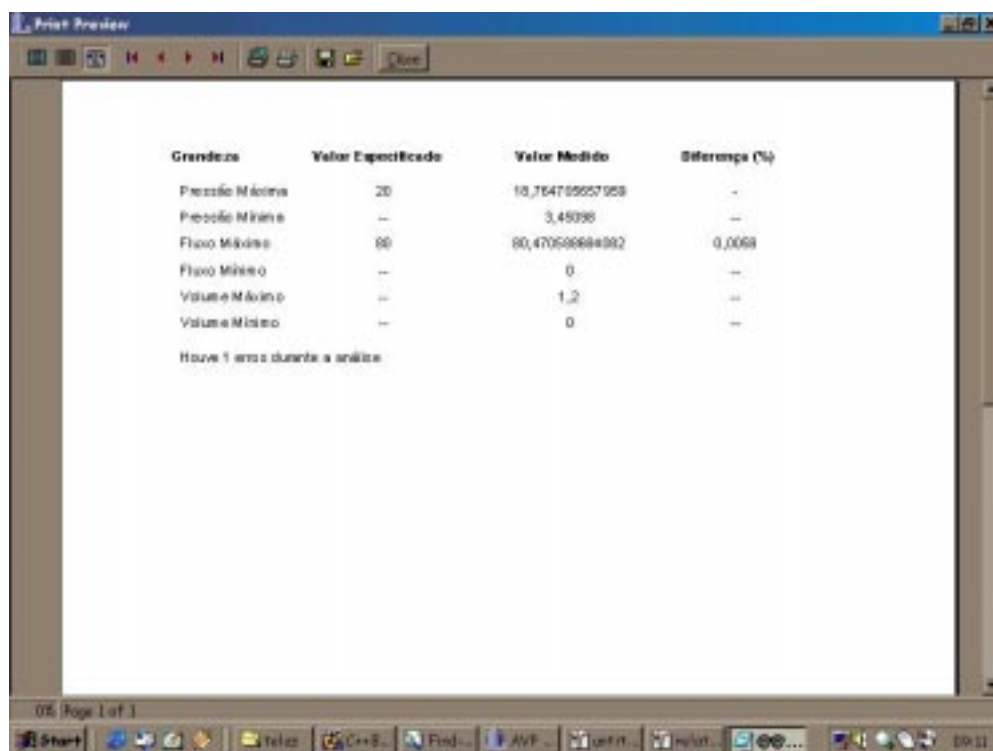
Figura 32 – Relatório final do AVP

Tabela 18 – Valores amostrados pelo AVP

Parâmetro	V. Mínimo	V. Máximo	Unidade
Fluxo	0	81,88	SLPM
Volume	0	1,2	l
Pressão	3,45	18,76	CmH <sub>2</sub> O

Na seqüência foi definido um limite máximo de 80 SLPM para o fluxo, o intuito deste teste foi verificar a apresentação do relatório com a especificação do erro e demonstrar a gravação do arquivo de erro. Estes dois itens são apresentados nas figuras 32 e 33 respectivamente.

No relatório de erro é apresentada a diferença entre o valor amostrado e o valor limite, no teste realizado verificou-se uma relação muito pequena entre limite e valor, o que um técnico normalmente desconsideraria como problema no equipamento.



The screenshot shows a 'Print Preview' window with a table containing the following data:

Grandezas	Valor Especificado	Valor Medido	Diferença (%)
Pressão Máxima	20	18,764769657958	-
Pressão Mínima	--	3,45098	--
Fluxo Máximo	80	80,47050884082	0,0068
Fluxo Mínimo	--	0	--
Volume Máximo	--	1,2	--
Volume Mínimo	--	0	--

Below the table, the text 'Houve 1 erro durante a análise' (There was 1 error during the analysis) is displayed.

Figura 32 – Relatório apresentado o limite de fluxo ultrapassado

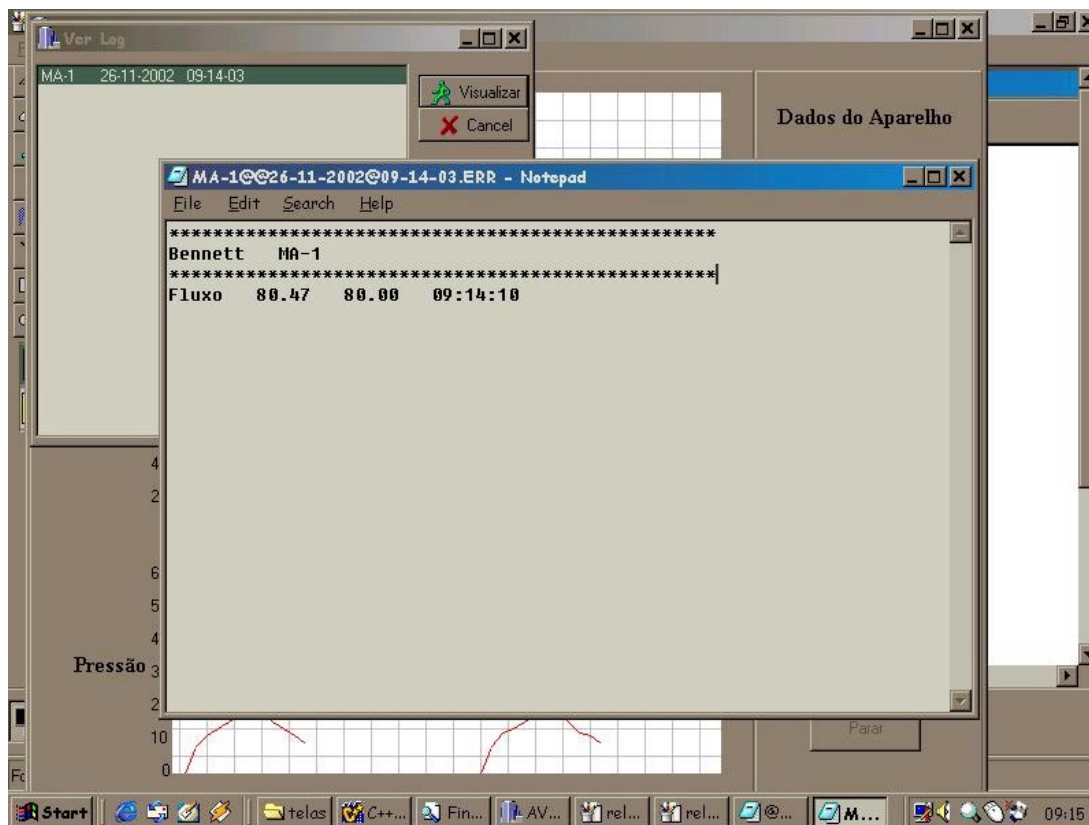


Figura 33 – Log de erro salvo para futura consulta do técnico

Os resultados apresentados aqui são bastante satisfatórios e animadores para o desenvolvimento de uma aplicação mais complexa.

## 6. CONCLUSÃO

Este trabalho teve o intuito de descrever o projeto de um analisador de ventiladores pulmonares.

O aparelho desenvolvido tem uma grande utilidade prática junto as equipes de engenharia dos hospitais, visto que, os aparelhos ventiladores pulmonares são equipamentos de alto risco, e desta forma, devem estar com todos os seus parâmetros de uma forma correta, para que deste modo auxilie e não prejudique o processo de atendimento de pessoas em casos críticos, onde há a necessidade de auxílio no processo respiratório.

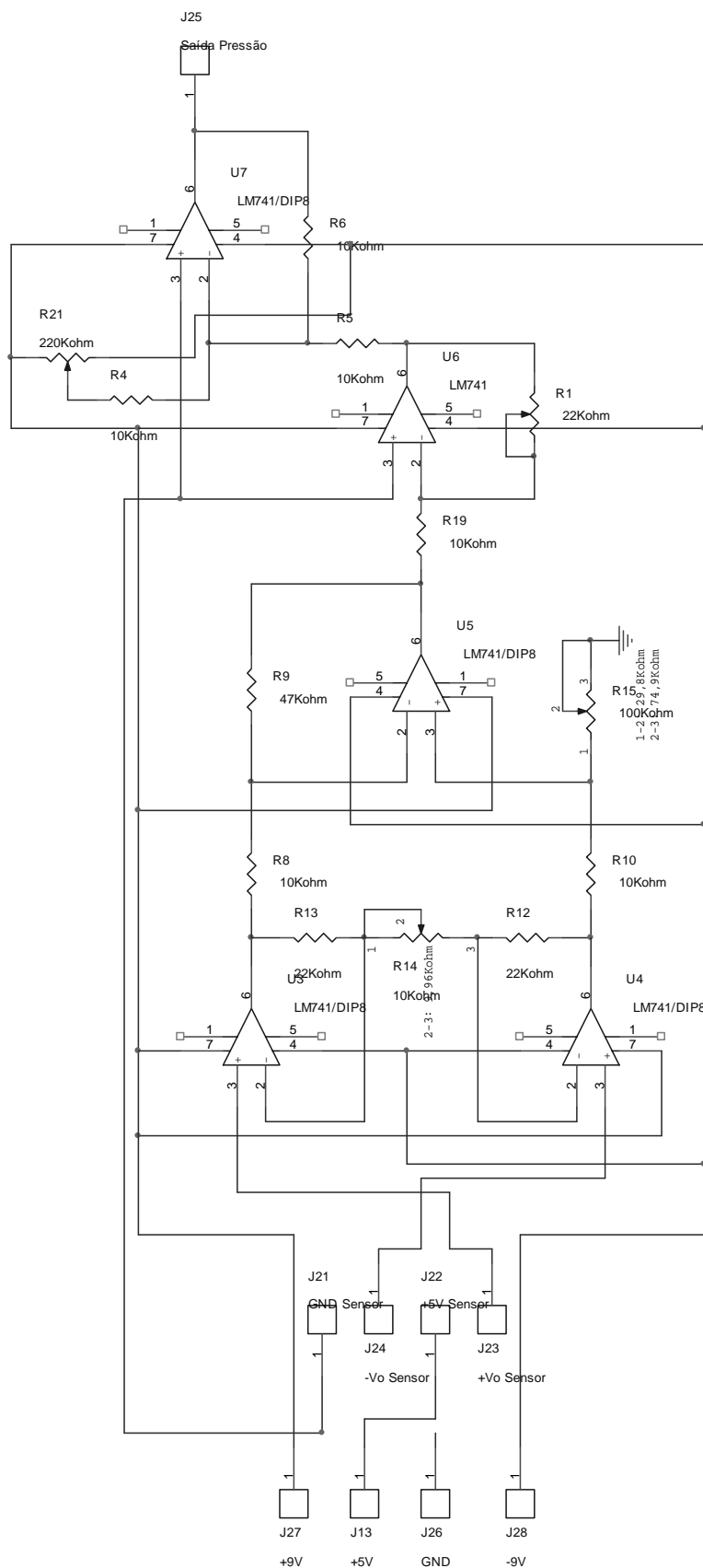
Existem diversos melhoramentos que poderão ser desenvolvidos conforme a necessidade, deste a melhora dos sensores, até a integração de um módulo que verifique a oxigenação, ou mesmo, um melhoramento de software como o desenvolvimento deste para outras plataformas.

Finalmente, espera-se a realização de testes de desempenho mais rigorosos para determinar sua validade e com isto permitir a sua utilização de uma boa forma.

## 7. REFERÊNCIAS BIBLIOGRÁFICAS

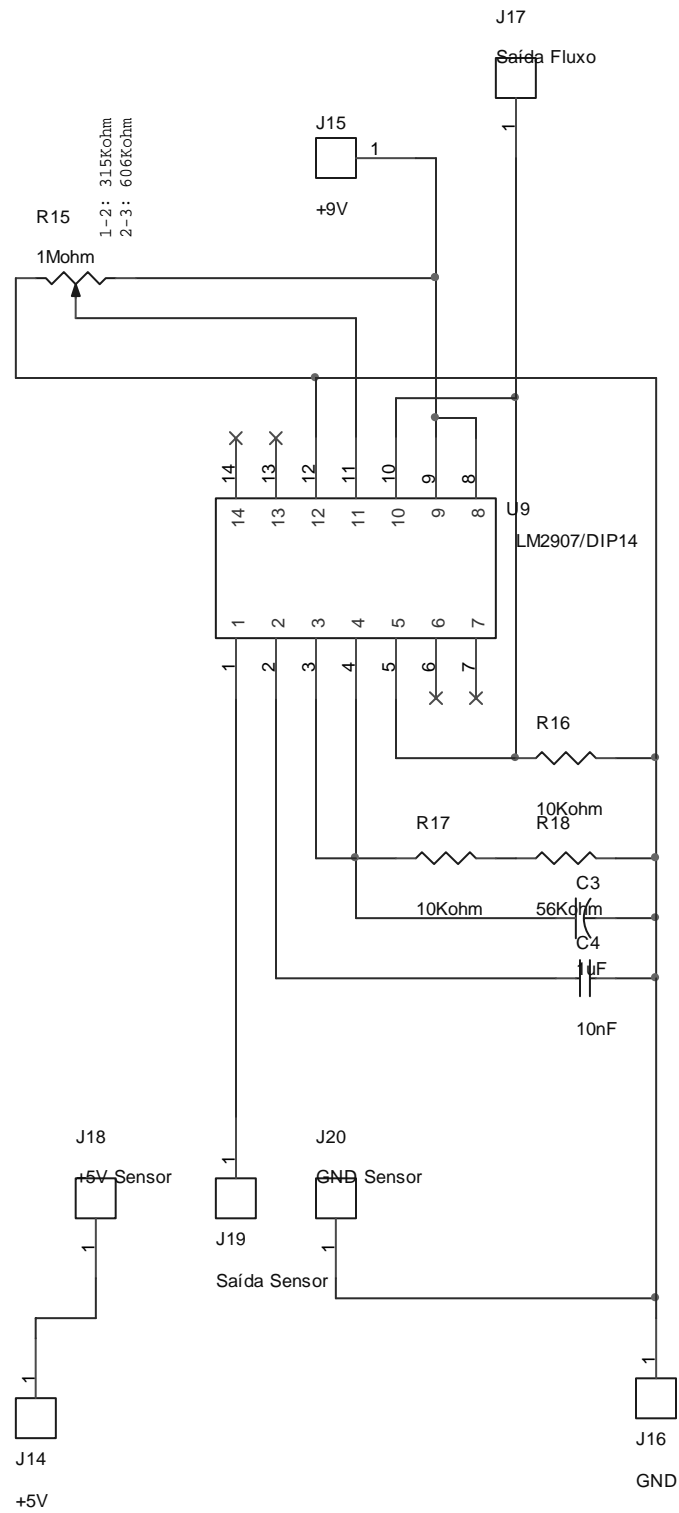
- [ASTM - 90] – MATERIALS, American Society for Testing and; Standard Specification for Ventilators Intended for Use in Critical Care; ASTM; 1990.
- [BRA - 96] – BRASIL/ABNT; Ventiladores Pulmonares para uso medicinal; NBR 13763; 1996.
- [KHA - 84] – **Khambata**, Adi J.; Microprocessadores, Microcomputadores: Volume 1; Editora Campus.
- [KRI - 00] – **Heidenstrom**, Kris; [k@heidenstrom.gen.nz](mailto:k@heidenstrom.gen.nz); Nova Zelândia
- [LUC - 02] – LUCAS Novasensor; Datasheet: NPC-1210 Low Pressure Series; Lucas Novasensor; 2002.
- [MAI - 92] – MAIA, José A. e EMMERICK, João C.; Ventilação Pulmonar Artificial; Editora Ateneu; 1992.
- [MAL - 85] – **Malvino**, A.P.; Microcomputadores e Microprocessadores; Editora McGraw-Hill; 1985.
- [MUC - 96] – MUCHAGATA, L.S. **II Consenso Brasileiro de Ventilação Mecânica.** Disponível por WWW em [http://www.cardiol.br/esquina/manualc/X\\_VENTILA%C3%87AO\\_MECANICA.pdf](http://www.cardiol.br/esquina/manualc/X_VENTILA%C3%87AO_MECANICA.pdf). (20 Dez. 2001 ).
- [NAT - 99] – NATIONAL Semicondutores; Datasheet: ADC0808/ADC0809: 8 bit  $\mu$ P Compatible A/D Converters with 8-Channel; National Semicondutores; 1999.
- [SEV - 00] – SEVERO, Leonardo S.; Avaliação do Desempenho dos Ventiladores Pulmonares; XVI Congresso Brasileiro de Engenharia Biomédica;2000.
- [STO - 99] – STOCK, M. Christine; Suporte Ventilatório Mecânico; Editora Manoli;2ª edição; 1999.
- [TOM - 81] – TOMPKINS, Willis S. e WEBSTER, John G.; Design of Microcomputer-based Medical Instrumentation; Editora Prentice-Hall; 1981.

## 8. ANEXOS

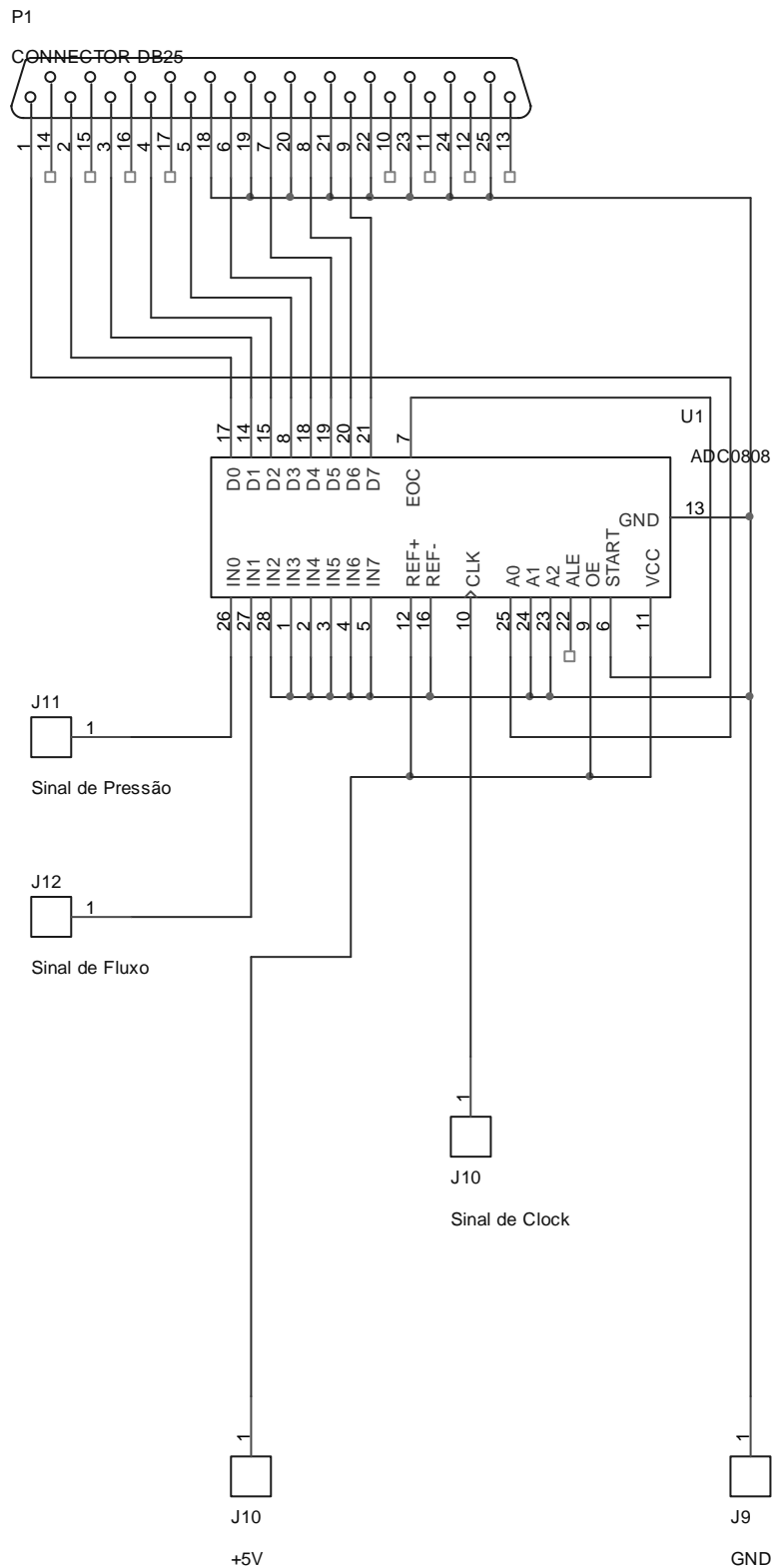


## Anexo 1 – Esquemático do Circuito Sensor de Pressão





Anexo 2 – Esquemático do Sensor de Fluxo



Anexo 3 – Esquemático para conversão A/D e conexão ao computador

```

class cParalela{
public:
    cParalela(int &bidir);
    void setaDados(unsigned char dado);
    unsigned char leDados(void);
    void setaStatus(unsigned char dado);
    unsigned char leStatus(void);
    void setaControle(unsigned char dado);
    unsigned char leControle(void);
private:
    void outportb(unsigned int portid, unsigned char value);
    unsigned char inportb(unsigned int portid);
};

```

Anexo 4 – Protótipo da classe para comunicação através da porta paralela

```

class cRespirador{
public:
    //construtor
    cRespirador(AnsiString sMarca, AnsiString sModelo, AnsiString sSerie, AnsiString
sComent, cParametros parametros, int &porta);

    int fazerMedicao(void);
    void getMedicao(float &fPres, float &fFlux, float &fVol);
    void getMax(float &fPres, float &fFlux, float &fVol);
    void getMin(float &fPres, float &fFlux, float &fVol);
    void comparaMax(float fPres, float fFlux, float fVol);
    void comparaMin(float fPres, float fFlux, float fVol);
    int getAmostras(void){ return amostras;};
    int getErro(void){ return iErro;};
    float getParam(int qual){ return parame.getParam(qual);};
    ~cRespirador();
private:
    //metodos
    void gravaLog(float fPres, float fFlux, float fVol);
    void gravaErro(float fMed, float fLim, int param);

    //atributos
    cParametros parame;
    cParalela *comunicar;
    //atributos que contem os valores atuais das variaveis analisadas
    float fPresAtual, fFluxAtual, fVolAtual;

    //atributos q armazenam os valores maximos e minimos
    float fPresMax, fFluxMax, fVolMax;
    float fPresMin, fFluxMin, fVolMin;

    //atributos para realizar a media dos valores
    float fSomaPressao, fSomaFluxo, fSomaVolume;
    int amostras;
};

```

```

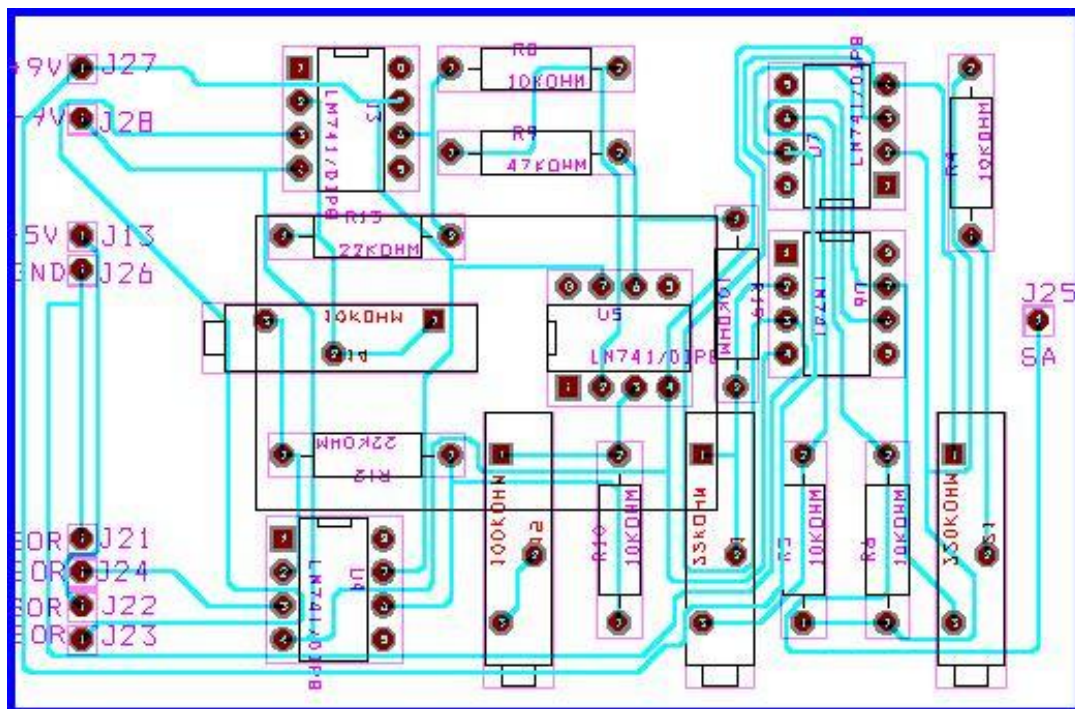
int iErro;

//atributos fisicos do aparelho analisado
String sMarca, sModelo, sComentarios, sSerie;

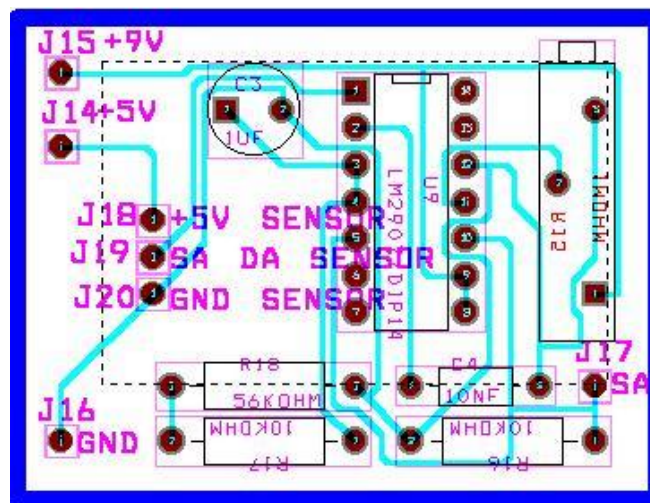
//atributo para o arquivo a ser gravado
String sNomeArq;
HANDLE fileLog;
ofstream *fileErro;
};

```

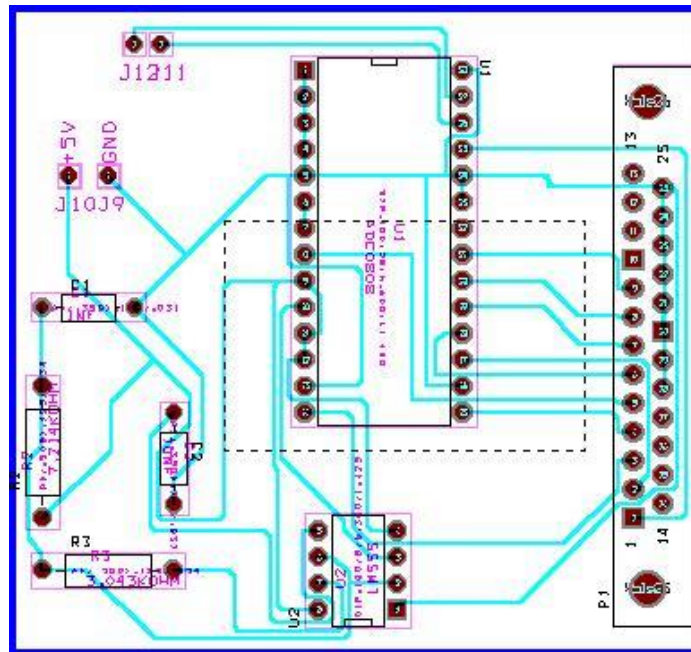
Anexo 5 – Protótipo da classe que simula o respirador



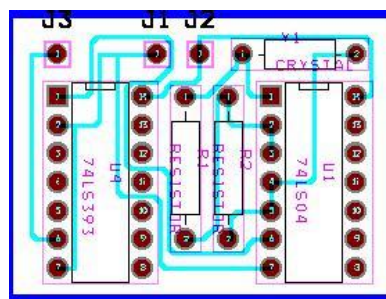
Anexo 6 – Placa de circuito impresso do sensor de pressão



Anexo 7 - Placa de circuito impresso do sensor de fluxo



Anexo 8 - Placa de circuito impresso do conversor A/D



Anexo 9 – Placa de circuito impresso do gerador de *clock*