UNIVERSIDADE FEDERAL DE PERNAMBUCO
GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

SISTEMA DE AQUISIÇÃO, ARMAZENAMENTO, TRATAMENTO E VISUALIZAÇÃO DE SINAIS NEUROLÓGICOS DE COBAIAS EM EXPERIMENTOS DE "SPREADING DEPRESSION"

TRABALHO DE GRADUAÇÃO

Aluno: Arthur Umbelino Alves Rolim (auar@cin.ufpe.br)
Orientador: Manoel Eusébio de Lima (mel@cin.ufpe.br)

Recife, 03 de abril de 2007.

Assinaturas

Este Trabalho de Graduação é resultado dos esforços do aluno Arthur Umbelino Alves Rolim, sob a orientação do professor Manoel Eusébio de Lima, conduzido no Centro de Informática da Universidade Federal de Pernambuco. Todos abaixo estão de acordo com o conteúdo deste documento e os resultados deste Trabalho de Graduação.

_____

Arthur Umbelino Alves Rolim

_____

Manoel Eusébio de Lima

Agradecimentos

Agradeço primeiramente, a Deus que me ajudou a percorrer este longo caminho da graduação e força de vontade para não desistir e chegar até o fim.

Agradeço também toda minha família. Meus pais, Antonio e Elinete, que sempre me incentivaram e apoiaram do começo ao fim, meus irmãos Dirlane, sempre me incentivou e ajudou quando precisei, e Rodrigo, minha dupla *default* de trabalho, que todas as cadeiras que pagamos juntos, sempre fez os trabalhos comigo, me ajudando muito nesse caminho.

Quero agradecer a todos aos colegas de universidade que me ajudaram a chegar até aqui: Ana Elizabeth, Martinelli, Marcelo, Adelma, Chico, Millena, Lamberto, Williams, Firma, Pablo, Lauro Moura, Carol, Geraldo, BPE, Tadeu, Marcio, Bengt, Emanoel, Leornado Lira, os 2 Carlos, Nancy, Jeane, Vitão, Farley, Pedro Lages, Paulo Piauí, Sylvinha, Ulliane, Laís, Joabe, Allan, Thierry, Carina, Augusto, Gabriel, Pedro Osandy, Cristiano, Abner, Adson e todos os outros alunos do CIn.

E finalmente, aos professores do CIn e CTG, Manoel Eusébio, Hélio Magalhães, Frederico Dias, Silvio Meira, André Santos, Patrícia Tedesco, José Dias e todos os outros que contribuíram para minha formação, meus agradecimentos.

Resumo

Spreading Depression é um fenômeno no qual ocorre uma queda acentuada na amplitude da atividade elétrica espontânea do córtex cerebral quando estimulado artificialmente. O comportamento destas ondas quando estudados podem diagnosticar doenças de causas não muito bem compreendidas, como a epilepsia e a enxaqueca. Então, um sistema que realizasse a maioria dos passos da experiência de maneira automatizada e sem a intervenção humana, aumentaria em muito a qualidade do resultado, além de deixar os cientistas livres para executar outras tarefas mais nobres, em paralelo, aumentando sua produtividade.

Palavras-chaves: spreading depression, depressão alastrante, captura de sinais biológicos.

Abstract

Spreading Depression is a phenomenon with an accented fall in the amplitude of the spontaneous electric activity of the cerebral cortex, when stimulated artificially. The behavior of these waves when studied may diagnosis illnesses of unknowns causes, as epilepsy and migraine. Then, a system that carried through the almost all experience, automatized and without the intervention human being, would increase the quality of the result, and leave the scientists free to execute other tasks, in parallel, increasing their productivity.

Keywords: spreading depression, biological signal capture, embed systems.

Índice

1. INTRODUÇÃO 9

1.1 MOTIVAÇÃO 10
1.2 OBJETIVOS 10
1.3 ORGANIZAÇÃO DO DOCUMENTO 11

2. SINAIS BIOLÓGICOS 12

2.1 CONVERSORES ANALÓGICO/DIGITAL (A/D) 13
2.2 CARACTERÍSTICAS DO SINAL 15
2.3 APLICAÇÕES DO PROCESSAMENTO DIGITAL 15
2.4 FILTRAGEM DIGITAL DE SINAIS BIOLÓGICOS 17
2.4.1 FIR 18
2.4.2 IIR 19

3. "SPREADING DEPRESSION" 21

3.1 EXPERIÊNCIA DE "SPREADING DEPRESSION" ATUALMENTE NA UFPE 23

4. SISTEMA SDVIEW 25

4.1 ARQUITETURA 26
4.2 CAMADA DE CAPTURA 28
4.2.1 AD8555 28
4.2.2 ARM LPC2101 29
4.2.2.1 MÁQUINA DE ESTADOS 31
4.3 CAMADA DE COMUNICAÇÃO 33
4.3.1 PACOTES 33
4.3.2 O PROTOCOLO 34
4.3.2.1 CONFIGURAÇÃO DA CAMADA DE CAPTURA 35
4.3.2.2 ENVIO DE DADOS PARA CAMADA DE SOFTWARE 37
4.4 CAMADA DE SOFTWARE 39
4.4.1 TECNOLOGIAS USADAS 39
4.4.2 CASOS DE USO 40
4.4.3 DESCRIÇÃO DA ESTRUTURA DO SOFTWARE 41

5. RESULTADOS OBTIDOS 47

5.1 CAPTURA 47

5.2 COMUNICAÇÃO **48**
5.3 SOFTWARE **48**

6. CONCLUSÃO E TRABALHOS FUTUROS **50**

REFERÊNCIAS **51**

Índice de figuras

Figura 1 – Conversão de um sinal analógico para digital 13
Figura 2 - Esquema de um conversor A/D 14
Figura 3- Filtro digital 18
Figura 4 -Estrutura lógica de um filtro FIR[9] 19
Figura 5 - Estrutura lógica de um filtro IIR[10] 20
Figura 6. Atividade elétrica durante um spreading depression 21
Figura 7 - Experimento de "spreading depression" 23
Figura 8 - Estrutura do sistema SDView 25
Figura 9 - Arquitetura SDView 26
Figura 10 - Esquemático funcional do AD8555[17] 29
Figura 11 - Diagramas de blocos do ARM 31
Figura 12 - Máquina de estados do ARM[12] 32
Figura 13 - Estrutura de um pacote 33
Figura 14 - Estrutura do HEADER 34
Figura 15 - Visão completa do pacote 34
Figura 16 – Fluxo de mensagens para configuração 36
Figura 17 – Fluxo de mensagens para envio de dados 37
Figura 18 - Exemplo de gráfico usando o ZedGraph 40
Figura 19 - Casos de uso 40
Figura 20 - Pacote de comunicação 42
Figura 21 - Pacote de armazenamento 44
Figura 22 - Pacote de gráficos 45
Figura 23 - Fachada do software 46
Figura 24 - Protótipo de captura 47
Figura 25 - Loopback na serial 48
Figura 26 - Software SDView 49

Índice de Tabelas

Tabela 1 - Constantes do protocolo de comunicação..... 38

1. Introdução

Pesquisas têm sido feitas durante anos em buscas das razões que podem causar doenças, decorrentes de possíveis disfunções cerebrais, como a epilepsia e enxaqueca. Alguns cientistas têm tentado identificar possíveis razões para o aparecimento destas anomalias através de estudos do comportamento das ondas cerebrais de vítimas destas disfunções.

Uma das linhas de pesquisa, ora em andamento no Departamento de Nutrição da UFPE é o estudo do fenômeno denominado *spreading depression* (depressão alastrante).

Spreading depression é um fenômeno no qual ocorre uma queda acentuada na amplitude da atividade elétrica espontânea do córtex cerebral quando estimulado artificialmente. Essa depressão se propaga a áreas vizinhas e, após certo período de tempo, retorna ao quadro normal. O comportamento destas ondas é então analisado para diagnosticar doenças de causas não muito bem compreendidas, como a epilepsia e a enxaqueca.

Em laboratório estes experimentos são em geral analisados em cobaias. Atualmente a captura dos impulsos cerebrais durante a experiência é executada ainda de maneira quase manual, desde a calibração dos aparelhos eletrônicos utilizados na captura até a captura e análise dos sinais neurológicos. Este processo demanda um longo período de tempo para sua realização, exigindo captura constante de dados por parte dos pesquisadores, o que pode acarretar erros durante a coleta.

Um sistema eletrônico automatizado que realizasse a maioria dos passos da experiência sem a intervenção humana, aumentaria em muito a qualidade do resultado, além de deixar os cientistas livres para executar outras tarefas mais nobres, em paralelo, aumentando sua produtividade. Além de deixar muito mais confiáveis os resultados gerados da experiência, pois os dados seriam visualizados e armazenados de maneira digital, sem perda de qualidade nos dados armazanados.

1.1 Motivação

A implementação de um sistema capaz de realizar um experimento de análise de s*preading drepession* praticamente automatizado será um grande avanço nas pesquisas realizadas pelo Departamento de Nutrição da Universidade Federal de Pernambuco. Com os dados adquiridos eles poderão analisar o resultado da experiência de maneira mais confiável e digital, o que eles permitirá diagnosticar novos comportamentos cerebrais e possíveis tratamentos para doenças que não sabemos ao certo a causa e origem, desse modo, contribuindo para o estudo e tratamento de doenças relacionadas ao cérebro.

Dentro deste contexto, e numa linha voltada para saúde, de interesse do Grupo de Engenharia de Computação, numa área diretamente relacionada a tratamento de sinais digitais, decidimos desenvolver juntos com o Departamento de Nutrição da UFPE uma infra-estrutura capaz de auxiliar pesquisadores no diagnóstico de doenças celebrais e de criar um novo nicho de pesquisa interdisciplinar para nosso curso de Engenharia da Computação.

1.2 Objetivos

Este trabalho de graduação tem como objetivos o desenvolvimento de um sistema eletrônico que permita entre outros:

- Capturar sinais cerebrais de cobaias;
- Converter sinais analógicos em digitais;
- Estudo de filtros digitais para correção de sinais biológicos;
- Transferir sinais como dados para PC;
- Traçar gráficos a partir dos dados obtidos.

Enfim montar uma infra-estrutura mínima, um protótipo, que possa permitir aos pesquisadores desenvolver de forma mais precisa seus diagnósticos e ao mesmo tempo, formar engenheiros de computação com formação no tratamento de sinais de pequena intensidade, sua amplificação, conversão A/D e tratamento digital.

1.3 Organização do documento

Este trabalho está organizado da seguinte maneira:

- O capítulo 2 apresenta uma introdução aos sinais biológicos, seus tipos, como ocorre sua captura e uma breve introdução à filtragem digital de sinais biológicos.
- O capítulo 3 fala sobre o fenômeno de *spreading depression* e como as experiências estão ocorrendo no departamento de nutrição da UFPE.
- No capítulo 4 é apresentada a estrutura do sistema proposto como solução automatizada das experiências realizadas na UFPE de *spreading depression*.
- O capitulo 5 mostra os resultados obtidos a partir da implementação do sistema proposto no capítulo 4.
- O capítulo 6 apresenta uma conclusão do sistema proposto e futuros trabalhos e melhorias para o sistema.
- Referências utilizadas para desenvolvimento deste trabalho de graduação.

2. Sinais biológicos

A Medicina usa o termo sinal para indicar qualquer indicador objetivo de alterações patológicas de natureza fisiológica ou morfológica (por exemplo, um aumento na força de um reflexo, uma coloração típica de uma lesão dérmica, uma variação na temperatura corporal).

O termo sinal biológico se refere mais especificamente às variações temporais que ocorrem em alguma forma de energia no corpo humano, como resultado de seu funcionamento. Deste modo, por exemplo, o ECG caracteriza a evolução temporal dos potenciais de ativação elétrica do tecido muscular cardíaco e o EEG caracteriza a evolução temporal da somatória dos potenciais de membrana dos neurônios.

Na engenharia é entendido como sinal qualquer evento que carregue informação, e é nessa direção que os sinais biológicos passaram a ser definidos.

Alguns variam no tempo de forma lenta (por exemplo, a temperatura interna), outras de forma muito rápida (por exemplo, a vibração das cordas vocais). Além disso, a energia física presente em um sinal biológico pode ser de natureza elétrica (como o ECG e o EEG) ou não elétrica (térmica, mecânica ou fotônica).

Com a evolução tecnológica, a análise desses sinais biológicos passou a ser realizado por computadores. Para que isso aconteça, entretanto, é necessário transformar o sinal original, que geralmente é analógico por natureza, em números discretos (dígitos), que é a maneira com que o computador digital trabalha com quantidades numéricas discretas. Isso equivale a tomar medidas de amplitude do sinal, em intervalos regulares, e converter essas medidas em números digitais com certo grau de precisão. Esse processo é chamado de digitalização, ou amostragem do sinal no tempo, como mostrado na Figura 1. Esta conversão é realizada por um circuito eletrônico especializado (geralmente uma placa que se acopla internamente ao microcomputador), chamado conversor analógico-digital, ou conversor A/D.

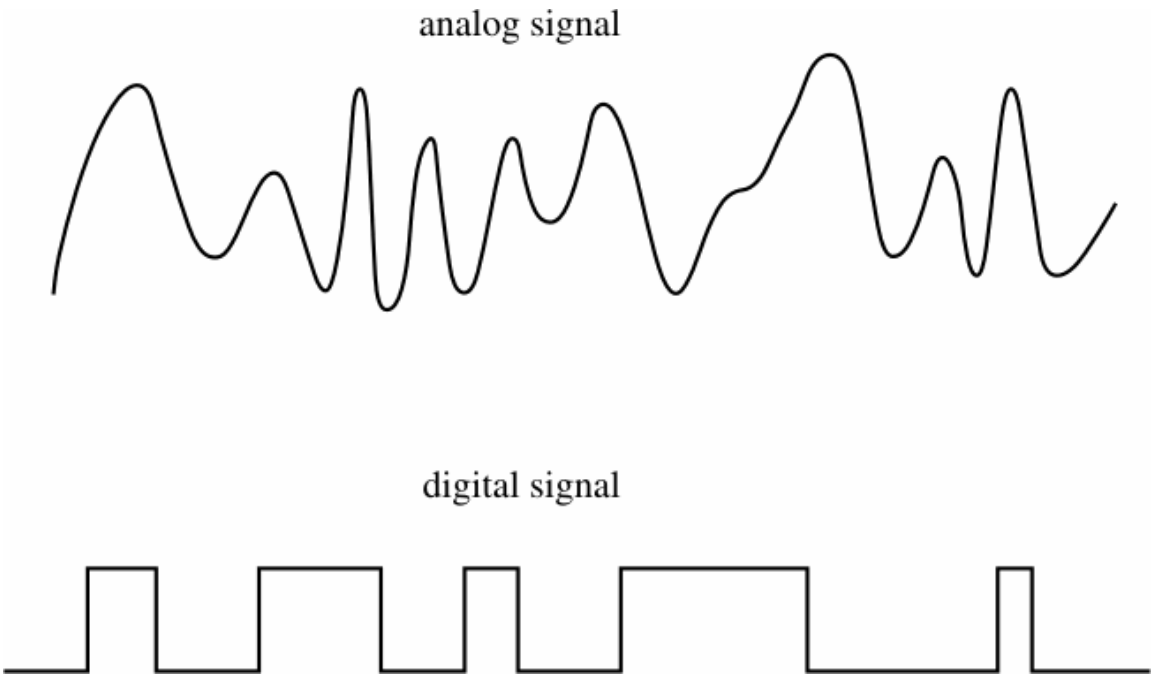

Figura 1 – Conversão de um sinal analógico para digital

2.1 Conversores Analógico/Digital (A/D)

Conversores A/D são a base de todo o interfaceamento eletrônico entre o mundo analógico e o mundo digital. Estes circuitos estão presentes na grande maioria dos instrumentos de medida atuais e são os responsáveis pelo aumento significativo nos níveis de precisão e exatidão assim como o barateamento e popularização de instrumentos de medida digitais.

Grandezas analógicas são capturadas com uma certa freqüência e o seu valor em um dado instante do processo é discretizado, como ilustrado na Figura 2.

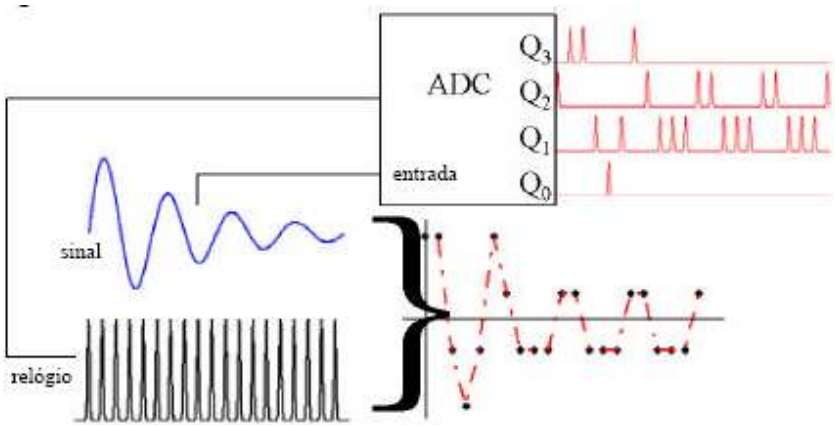

Figura 2 - Esquema de um conversor A/D

Características importantes e comuns aos conversores:

- **Faixa dinâmica:** é a faixa de amplitude de operação do sinal analógico (em geral uma tensão) dentro da região de trabalho (linear) do conversor. O sinal de entrada deve ser condicionado de forma a possibilitar sua máxima utilização dentro dessa faixa dinâmica. Os conversores D/A e A/D apresentam na prática uma faixa dinâmica de 0,1 a 10V.
- **Resolução:** é a menor quantidade que pode ser convertida dentro da faixa dinâmica do sinal de entrada. É especificada pelo número de *bits* do conversor. São encontrados na prática conversores com resoluções de 8 a 20 *bits*. Ex.: Resolução de 12 *bits* ⇒ significa que o conversor consegue diferenciar sinais com amplitude $1/2^{12}$ do valor total da faixa dinâmica. Para uma faixa dinâmica de 5V, a menor amplitude que pode ser resolvida é: $5/2^{12}$ = 0,00122 ⇒ 1,22 mV.
- **Tempo de conversão:** é o tempo necessário para se obter o valor na saída a partir do momento em que o sinal de entrada foi aplicado e iniciado o processo de conversão. Depende da estrutura do circuito utilizado e da sua resolução. De modo geral, quanto maior a resolução, maior o tempo de conversão. Encontram-se, na prática, tempos de conversão variando desde

alguns segundos até menor que nanosegundos. Este tempo é importante para definir a máxima freqüência possível a ser convertida a partir de um sinal de entrada variante no tempo.

- **Erro de linearidade:** expressa o desvio do resultado de conversão de uma reta ideal. É expresso em com mais ou menos uma porcentagem do valor total ou em número de *bits*. Ex: Um erro de linearidade de mais ou menos 0,4% equivale a uma linearidade de mais ou menos um *bit* num conversor de oito *bits*.

2.2 Características do sinal

Para se conseguir determinar a velocidade de conversão a ser utilizada devemos analisar, além da maior freqüência presente no sinal, também a periodicidade com que queremos registrar o sinal em função do tempo. A esse respeito, podemos considerar que existem dois tipos de sinais:

- **Sinais estáticos:** quando o tempo não é considerado;
- **Sinais dinâmicos:** quando são medidos e processados em relação ao tempo.

Tomando o exemplo da pressão sanguínea: a pressão sistólica e diastólica instantâneas, são sinais estáticos. A mensuração da pressão sanguínea de forma contínua (registro da pressão), ou uma medida estática freqüentemente repetida (por exemplo, de uma em uma hora), são sinais dinâmicos.

2.3 Aplicações do processamento digital

O processamento digital dos sinais biológicos oferece muitas vantagens e uma maior flexibilidade em relação aos sistemas analógicos. Como, após o processo de conversão A/D, um sinal temporal passa a ser representado por um conjunto de valores numéricos discretos na memória do computador, é muito fácil e

conveniente realizar diversas manipulações e transformações matemáticas sobre esse conjunto de números. Assim, diversas funções que só podem ser conseguidas por meio de circuitos eletrônicos caros e especializados em sistemas de registro analógico, são obtidas e simuladas sem dificuldades por meios puramente lógicos (por software).

Algumas dessas funções são:

- Filtragem de ruídos, isto está relacionado à eliminação de interferências de baixa ou alta freqüência, tais como artefatos de movimento, cabos mal conectados, influência da rede elétrica comum, etc.
- Amplificação, refere-se ao aumento da amplitude absoluta do sinal.
- Integração refere-se à soma cumulativa de variações de um sinal.
- Diferenciação lida com a medida da velocidade de variação de um dado sinal.
- Obtenção das freqüências senoidais puras, ou harmônicas, que integram um sinal complexo. Essa análise, denominada de espectral ou de Fourier, permite isolar componentes de significado fisiológico em um sinal, tais como as ondas alfa, beta, gama e delta em um EEG, ou as periodicidades de flutuação da freqüência cardíaca registrada a longo prazo.
- Síntese de algumas formas de registro a partir de outras: por exemplo, um vetocardiograma pode ser obtido a partir de um registro de ECG simples de três derivações (sem necessidade de comprar um vetocardiógrafo); ou o mapeamento espacial de sinais, como, por exemplo, a topografia cerebral de EEG.
- O reconhecimento e interpretação automática de sinais. tais como nos novos equipamentos de ECG que são capazes de realizar, automaticamente, diversas medidas de amplitude, duração, e propor interpretações diagnósticas com grande acurácia.

Como o sinal biológico digitalizado pode ser armazenado de forma econômica, na forma de arquivos de computador este pode também ser transmitido confiavelmente através de redes de computadores, não importando a distância e pode ser exibido e impresso de inúmeras formas e maneiras.

2.4 Filtragem digital de sinais biológicos

Normalmente, sinais biológicos têm baixa freqüência, que são contaminadas de maneira intrínseca por ruídos biológicos, o ECG, por exemplo, pode ser corrompido pelo ruído de algum músculo ao redor do coração, e também pelos instrumentos utilizados durante a captura dos sinais.

Um ruído adicional é causado por causa da discretização do sinal e durante a conversão analógico-digital. Esses ruídos são amplificados e em alguns casos podem alterar os resultados finais. [7]

Para eliminar os ruídos, filtros são incorporados ao sistema e podem ser classificados quanto aos seus limites de corte em:

- Filtros passa-baixa (*Low Pass Filter* ou LPF) são utilizados para rejeitar freqüências que estão acima da freqüência de corte desejada do sinal desejado, como os sinais biológicos são de natureza de freqüência baixas esses filtros são muito utilizados para filtragem dos ruídos que os sinais possam possuir.
- Filtros passa-faixa permitem a passagem das freqüências de uma certa faixa e rejeita (atenua) as freqüências fora dessa faixa.[21]
- Filtros passa-alta (*High Pass Filter*) permitem a passagem das frequências altas com facilidade, porém atenua (ou reduz) a amplitude das frequências abaixo de frequência de corte.[22]

Os filtros digitais são representados por equações discretas que aplicadas a um sinal digital de entrada x(n), geram uma saída y(n), de acordo com a Figura 3, com freqüências dentro de certa faixa selecionada. Os mais utilizados são os do tipo FIR[9] (*Finite Impulse Response*), ou IIR[10] (*Infinite Impulse Response*).

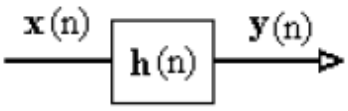

Figura 3- Filtro digital

A seguir temos uma breve introdução sobre as características dos filtros digitais mais utilizados atualmente, equações genéricas e esquema de blocos. Com essas informações teremos uma noção básica de como eliminar os ruídos mais comuns aos sinais biológicos depois de discretizados.

2.4.1 FIR

Um filtro FIR[9] ou *Finite Impulse Response* é um tipo de filtro digital caracterizado por uma resposta ao impulso que se torna nula após um tempo finito, em contraste com os filtros IIR[10].

Um filtro FIR[9] digital genérico terá uma saída dada pela fórmula:

$$y(n) = h_0 x(n) + h_1 x(n-1) + ... + h_P x(n-P)$$

Onde *P* é a ordem do filtro, *x*(*n*) o sinal de entrada, *y*(*n*) o sinal de saída e h_i são os coeficientes do filtro.

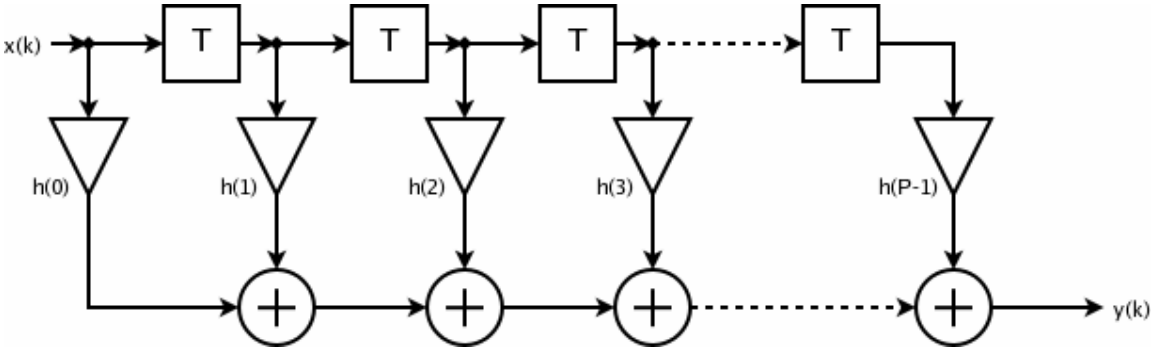

Figura 4 -Estrutura lógica de um filtro FIR[9]

Propriedades:

- São inerentemente estáveis.
- Não usam realimentação. Em conseqüência os erros de arredondamento não se acumulam.
- Podem ter fase linear.
- Podem ter fase mínima.

2.4.2 IIR

Um filtro IIR[10] ou *Infinite Impulse Response* é um filtro digital com resposta ao impulso de duração infinita.

Um filtro IIR[10] digital genérico terá uma saída dada pela fórmula:

$$y_n = b_0 x_n + b_1 x_{n-1} + \cdots + b_N x_{n-N} + a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_M y_{n-M}$$

Onde os a e b são os coeficientes do filtro. E a ordem é o maior número entre os valores de M e N.

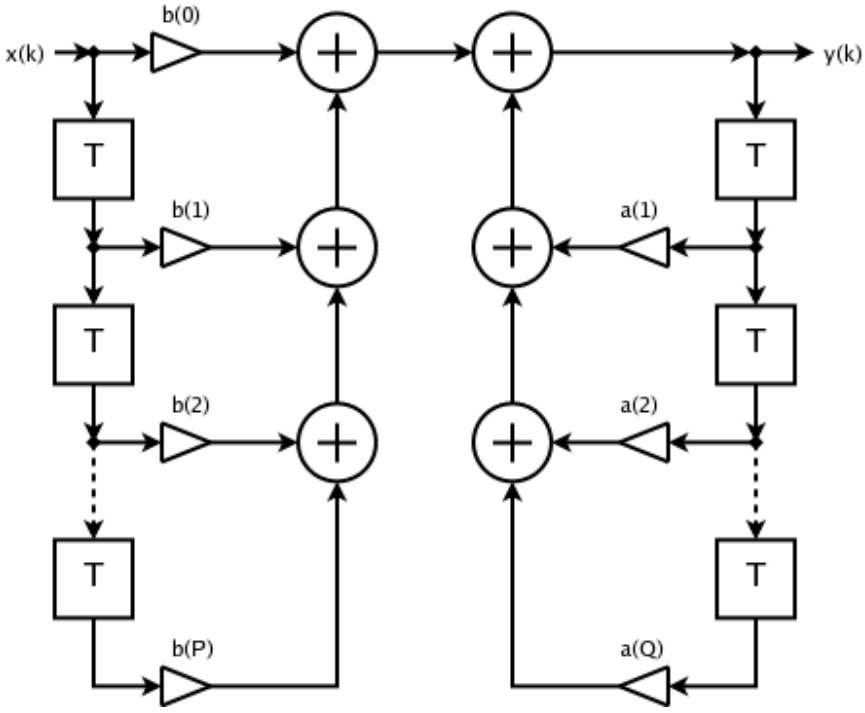

Figura 5 - Estrutura lógica de um filtro IIR[10]

Propriedades:

- Filtros FIR[9] e IIR[10] podem cumprir as mesmas exigências, mas o IIR[10] com uma ordem menor, logo precisa de menor poder computacional.
- São filtros instáveis, apesar de serem construídos para serem estáveis.

Hamer CF, Ifeachor EC e Jervis BW em um trabalho publicado chamado "*Digital filtering of physiological signals with minimal distortion*" [25], fazem um estudo sobre a utilização de filtros digitais FIR e IIR, que são usados em muitos processos de filtragem biológica, no trabalho demonstram que apenas utilizando os filtros digitais citados acima, é possível garantir uma distorção mínima do sinal capturado. Assim, demonstrando que o uso de filtros digitais durante a captura de sinais biológicos é altamente recomendado para validação dos resultados.

3. “Spreading Depression”

A *spreading depression*, descoberta por Aristides Leão[20], é um fenômeno no qual ocorre uma queda acentuada na amplitude da atividade elétrica espontânea do córtex cerebral quando estimulado artificialmente; essa depressão se propaga a áreas vizinhas e, após certo período de tempo, retorna ao quadro normal.

O estímulo pode ser provocado por um toque, choque elétrico ou substância química. No entanto, até hoje não se sabe qual é o mecanismo que origina naturalmente o fenômeno.

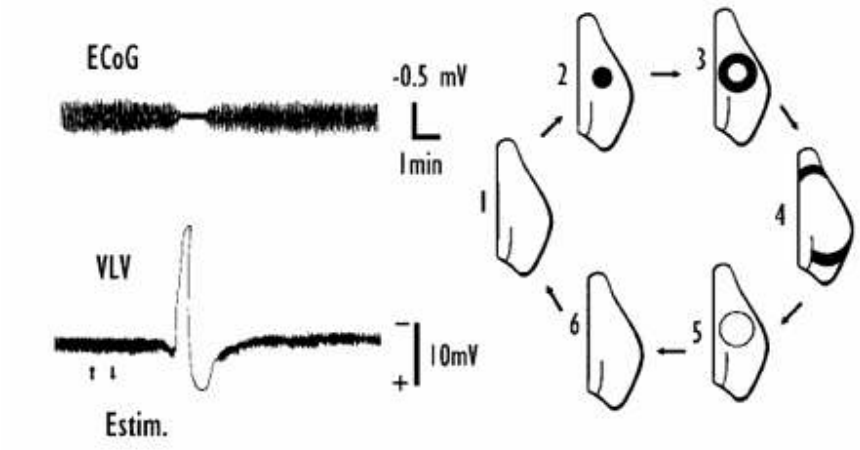

Figura 6. Atividade elétrica durante um *spreading depression*

O termo *depression* indica que a amplitude do EEG em uma determinada região cortical se torna temporariamente reduzida, isto é, a diferença de potencial elétrico entre aquele ponto cortical e um outro ponto vizinho tende para zero, quando a região é invadida pelo fenômeno.

Assim, o *spreading depression* foi descrita originalmente como uma “onda” reversível e propagável de redução da atividade elétrica cerebral (tanto a espontânea como a provocada), acompanhada do aparecimento de uma “variação lenta de voltagem” (VLV) na região do cérebro invadida pelo fenômeno, em

resposta à estimulação elétrica, mecânica ou química de um ponto da superfície cerebral. Tanto a depressão do EEG, quanto a VLV, características do *spreading depression* se propagam de forma concêntrica a partir do ponto estimulado, atingindo gradualmente regiões corticais mais e mais distantes, enquanto a área inicialmente deprimida começa a se recuperar, como ilustrado na Figura 6.

Como regra geral, a recuperação completa do EEG é obtida após 5 a 10 minutos. Ao contrário do EEG, a VLV tem características do tipo “tudo ou nada”, ou seja, a sua presença, com uma “forma de onda” bem definida, com início e fim fáceis de identificar, sempre indica a existência do *spreading depression*.

Por isso, a VLV é muito usada para se calcular a velocidade com que o fenômeno se propaga pelo tecido nervoso. Surpreendentemente, em todos os vertebrados em que já se registrou a DA (desde peixes até mamíferos), a sua velocidade de propagação tem se mostrado notavelmente baixa (alguns mm/min), em comparação com a ordem de grandeza da propagação dos impulsos nervosos em axônios (até dezenas de m/s).[3]

Aristides Leão[20] descobriu o fenômeno quando pesquisava a epilepsia experimental, nos Estados Unidos. Em um experimento em 1943, colocou uma fileira de eletrodos na superfície cortical do cérebro de um coelho, como ilustrado na Figura 7. O primeiro eletrodo estimularia a região, enquanto os outros registrariam as ondas elétricas. Dessa forma, ele reparou que, ao ser estimulada, a região sofria uma depressão intensa e duradoura de oscilações elétricas espontâneas, que se expandia lentamente e, ao fim de cerca de 15 minutos, retornava à normalidade.

Figura 7 - Experimento de "spreading depression"

Com o desenvolvimento das pesquisas, Leão[20] comprovou que o fenômeno não se restringe ao córtex, e pode ser identificado em outras estruturas neurais.

Outros pesquisadores se dedicaram então à pesquisa da depressão em retinas isoladas de animais, sobretudo sapos e pintos. Observaram que essas áreas também apresentavam o efeito da *spreading depression*, em forma de ondas, perceptíveis a olho nu.

Em busca de respostas, muitos centros de pesquisa ainda se dedicam aos estudos da *spreading depression*, sobretudo na República Tcheca, nos Estados Unidos e no Brasil. No Brasil, particularmente no Laboratório de Neurofisiologia da UFRJ que leva o nome de Aristides Leão e no Laboratório de Fisiologia da Nutrição Naíde Teodósio (LAFINNT), do Departamento de Nutrição da UFPE.

3.1 Experiência de “spreading depression” atualmente na UFPE

Na UFPE, no Centro de Nutrição, pesquisadores realizam experiências de *spreading depression* utilizando ratos albinos de laboratório como cobaias. As variações das ondas neurais das cobaias são capturadas, amplificadas e depois

são impressas em papel para que os pesquisadores possam analisar e tirar suas conclusões.

Nestes experimentos são utilizados vários equipamentos para a amplificação do sinal, desta maneira durante a transição entre a cobaia e a impressora o sinal pode ter adicionado a ele um ruído, que pode dificultar a análise dos dados após a experiência.

Também, durante as execuções das experiências é necessário que uma pessoa fique constantemente acompanhando todo o processo de aquisição de dados, para que haja uma produção confiável de resultados., Este processo, porém pode chegar até seis horas, consumindo muito tempo útil de um pesquisador, além da fadiga que pode ocasionar eventualmente erros de leituras de equipamentos.

Os dados resultantes da experiência são armazenados em computadores com a ajuda do software AxoScope 9[11], porém esse não provê muitos recursos para a análise dos mesmos, apenas mostra os dados que foram capturados e mesmo assim de maneira genérica, não de maneira especifica como a experiência demanda.

Neste cenário, um processo automatizado e totalmente digital poderia aumentar um muito à produtividade destas experiências, desde a captura até a visualização. Agilizando e melhorando a resultados das experiências.

4. Sistema SDView

O SDView foi uma solução proposta para o Departamento de Nutrição da UFPE para a automatização da experiência de *spreading depression*. Conforme, descrito na seção anterior, as experiências são executadas de maneira praticamente manual e os resultados ainda não são corretamente armazenados de maneira digital.

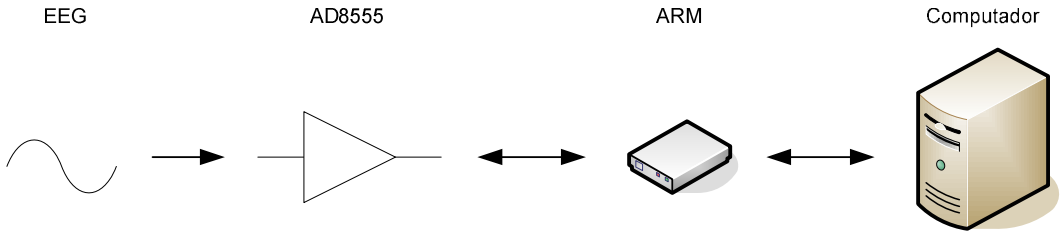

Figura 8 - Estrutura do sistema SDView

O sistema proposto, ilustrado na Figura 8, neste trabalho cria uma solução que vai desde a captura do sinal nos eletrodos fixados para a captura dos sinais EEG, sua amplificação, conversão analógico-digital e sua transferência para um computador onde esses dados serão armazenados e visualizados.

Nas próximas seções há uma descrição do sistema e após os resultados obtidos da implementação do mesmo.

4.1 Arquitetura

A arquitetura do sistema é ilustrada na Figura 9. Nela, é apresentada as camadas que o sistema possui para realizar a transferência dos dados da camada de captura, passando pela de camada comunicação e finalmente chegando na camada de software.

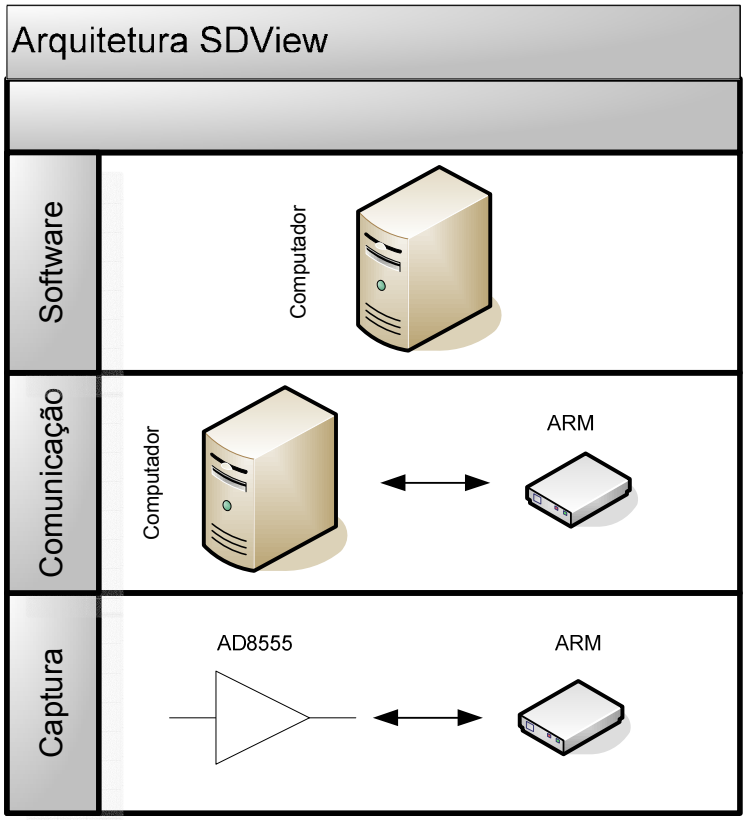

Figura 9 - Arquitetura SDView

O sinal EEG capturado por eletrodos no cérebro da cobaia sofre uma amplificação e um aumento no nível DC, controlado digitalmente, pois é na ordem de µ volts, assim não podendo ser capturado diretamente pelo conversor A/D.

Após, amplificado o sinal passará pelo conversor analógico-digital de um microprocessador ARM[12]. Neste microprocessador, através de um conversor

A/D de 10 *bits* com a taxa de amostragens de 1KHz, o sinal passará de analógico para discreto podendo assim ser manipulado digitalmente. O microprocessador ainda tem o papel de controlar a amplificação do sinal de maneira digital.

No microprocessador o sinal será empacotado é transferido por conexão serial a um computador onde os dados serão desempacotados armazenados e visualizados pelo software do sistema. O software tem a função de controlar o microprocessador para a captura ocorrer da maneira esperada.

Nas próximas seções será apresentada em detalhes a arquitetura mencionada acima. A arquitetura é dividida em três partes: captura, comunicação e software, que serão explicadas individualmente para desta maneira conseguir mostrar com maiores detalhes toda a estrutura do sistema.

4.2 Camada de captura

A camada de captura do sistema SDView é composto de: um amplificador com ganho e off-set do sinal de maneira digital, o AD8555[17], e um microprocessador ARM[12]. Estes dispositivos trabalhando em conjunto conseguem controlar a captura do sinal EEG gerado pelo rato albino durante a experiência de spreading depression.

4.2.1 AD8555

Este amplificador foi utilizado durante o experimento, pois permite alterar facilmente o ganho e o off-set do sinal capturado, assim facilitando a captura do sinal EEG. Ainda amplifica no modo diferencial[24], protegendo o sinal contra possíveis ruídos que possam acompanhá-lo. O ganho pode ser programado para variar entre 70 até 1280 vezes na tensão de entrada e o off-set, através de uma conexão serial com o microprocessador.

Este circuito integrado pode captura dois sinais ao mesmo tempo, mas para a experiência demanda captura quatro sinais para realização das análises, então dois circuitos AD8555[17] serão utilizados simultaneamente para a captura dos sinais.

Este componente ajuda a diminuir os vários ajustes que são realizados manualmente durante as experiências, assim dando maior agilidade ao processo de captura.

O esquemático do amplificador está ilustrado na Figura 10.

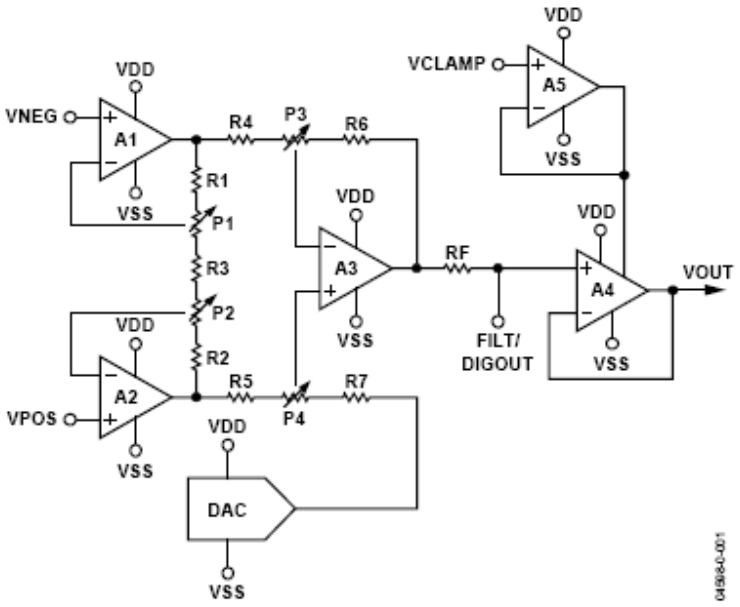

Figura 10 - Esquemático funcional do AD8555[17]

4.2.2 ARM LPC2101

O modelo do microprocessador utilizado neste sistema foi o ARM LPC2101[18], diagrama de blocos na figura 11, da Philips, baseados na CPU 16-*bit*/32-*bit* ARM7TDMI-S. A principal característica que nos fez adotar esse microprocessador foi o fato dele possuir quatro conversores analógico-digitais de 10 *bits*, precisamente o que precisamos para capturar os sinais biológicos com precisão. O modelo ainda funciona com a freqüência de 60 MHz que para captura de sinais biológicos, que são lentos, é mais que o suficiente.

Características gerais do microprocessador:

- Arquitetura *Load-Store*: as instruções somente processarão (soma, subtração, etc) valores que estiverem nos registradores e sempre armazenarão os resultados em algum registrador.
- Instruções fixas de 32 *bits* de largura (com exceção das instruções *Thumb* compactas de 16 *bits*) alinhadas em 4 *bytes* consecutivos da memória, com execução condicional, com poderosas instruções de carga e

armazenamento de múltiplos registradores, capacidade de executar operações de deslocamento e na ULA com uma única instrução executada em um ciclo de *clock*.

- Formato de instruções de três endereços (isto é, os dois registradores operandos e o registrador de resultado são independentemente especificados).
- 15 registradores de 32 *bits* para uso geral.
- Manipulação de periféricos de I/O como dispositivos mapeados na memória com suporte a interrupções.
- Conjunto de instruções aberto a extensões através de co-processador, incluindo a adição de novos registradores e tipos de dados ao modelo do programador.
- *Pipelines* de três e cinco estágios.

Na camada de captura ele tem o papel de configurar o AD8555[17] e captura com a freqüência correta os sinais convertidos pelos conversores A/D que ele possui.

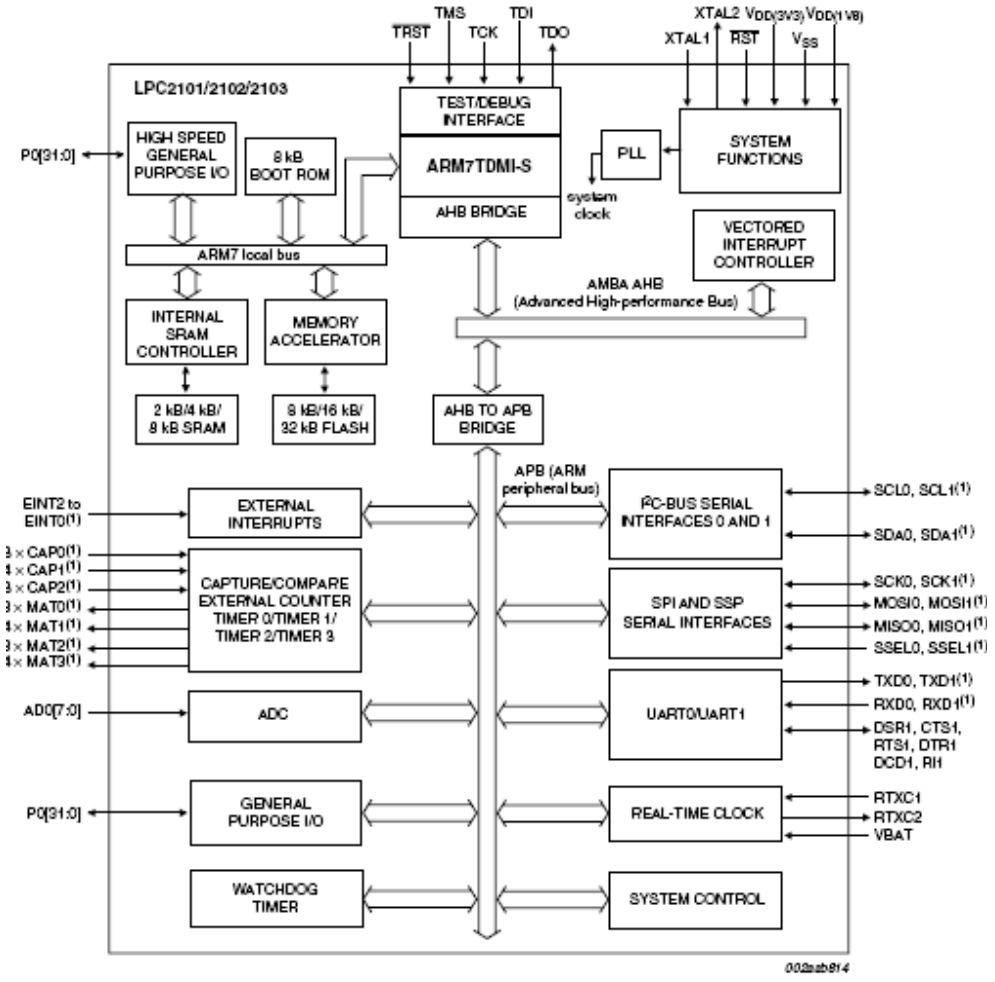

Figura 11 - Diagramas de blocos do ARM

Como pode ser visto no diagrama na Figura 10, a arquitetura fornecida pelo microprocessador ARM é bastante completa para resolvermos problemas de diversas naturezas na área de embarcados, inclusive neste trabalho mencionado.

4.2.2.1 Máquina de estados

Devido a simplicidade na apuração dos dados no sistema em estudo, não foi preciso, no estágio atual de desenvolvimento a utilização de um sistema operacional ARM[12]. Em vez desta estratégia, uma máquina de estados foi

desenvolvida. A máquina é composta por três estados e se encontra ilustrada na figura 12:

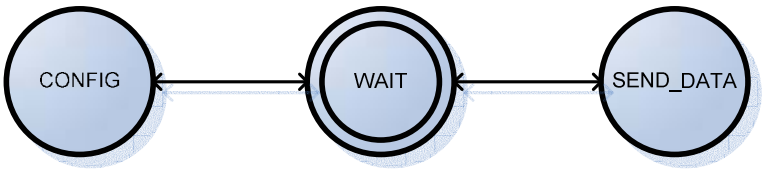

Figura 12 - Máquina de estados do ARM[12]

Descrição dos estados:

- WAIT: estado inicial do microprocessador, após ser ligado. Espera uma mensagem do computador para realizar alguma ação.
- CONFIG: estado que configura o microprocessador, de acordo com as instruções enviadas pelo software.
- SEND_DATA: estado que envia os dados capturados de experiência para o software.

A descrição da transição entre os estados é descrita na seção 4.3.2, onde está descrita a estrutura do protocolo.

4.3 Camada de comunicação

Entre a captura do sinal, especificada na seção 4.2, e o software, que será especificada na seção 4.4., uma camada de comunicação foi projetada obedecendo a um protocolo de comunicação para um perfeito acoplamento entre os dois subsistemas. Esta camada é formada por uma série de componentes abaixo relacionados:

4.3.1 Pacotes

Para troca de informações entre o computador e o microprocessador foram estabelecidos pacotes de 16 *bits*, os quais são transferidos através de uma conexão serial. O pacote do sistema consiste de dois *bytes*. Os primeiros 10 *bits* são reservados para dados (“DATA”) do pacote e os últimos seis *bits* para o “HEADER”, como ilustrado na figura 13. O tamanho de dez *bits* foi escolhido para a parte “DATA”, pois os conversores analógico-digitais do microprocessador são de dez *bits*.

Figura 13 - Estrutura de um pacote

Descrição do pacote:

- DATA: corresponde aos dados que foram capturados pelo microprocessador depois de sua amplificação ou a configuração desejada para captura.
- HEADER: ficou dividido em duas partes, os quatro primeiros *bits* representam o “CODE” usado na camada de protocolo e os dois últimos

bits representam o “CHART” como quatro sinais vão ser capturados pelo sistema foi necessário um campo no pacote para identificação de cada um, como ilustrado na figura14.

Figura 14 - Estrutura do HEADER

Visão completa do pacote:

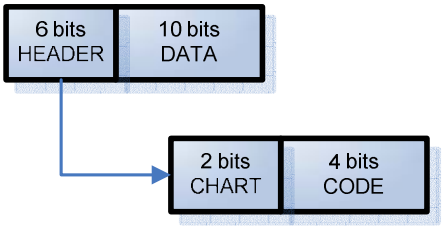

Figura 15 - Visão completa do pacote

4.3.2 O protocolo

“Protocolo é um padrão que especifica o formato de dados e as regras a serem seguidas, durante uma transferência de informação” [12]. Então, foi desenvolvido um protocolo para estabelecer uma comunicação entre a camada de captura e a camada de software, de maneira organizada e robusta.

Como citado acima, o microprocessador possui apenas uma máquina de estados para descrever todo seu comportamento e a transição entre os estados é administrada pelo software do sistema hospedeiro através de um protocolo que ambos os lados obedecem para haver uma comunicação.

O microprocessador quando ligado está sempre no estado de WAIT, esperando instruções de como deve agir. A partir desse estado inicial ele pode ir tanto para o estado CONFIG como para o estado SEND_DATA, dependendo das mensagens trocadas como o software obedecendo ao protocolo pré-estabelecido.

4.3.2.1 Configuração da camada de captura

No ciclo de configuração do sistema, o computador hospedeiro envia um pacote com o CODE, parte do HEADER, como a constante CONFIG_GAIN e na parte DATA do pacote a configuração, mascarada nos *bits*, que o ganho que AD8555[17] deve assumir para o microprocessador. Após a recepção dos dados iniciais, o microprocessador vai para o estado CONFIG. Como resposta o microprocessador deve enviar um ACK_CONFIG, caso a configuração tenha sido executada com sucesso, ou ERRO se não conseguir se configurar corretamente, O mesmo processo se repete para CONFIG_OFFSET, CONFIG_TIME e CONFIG_CHANNEL. Após isso, o microprocessador retorna para o estado WAIT. Caso algum erro ocorra durante a troca de mensagens do protocolo, um código ERRO é enviado para o software e o microprocessador volta para o estado de WAIT.

Na Figura 16 está esquematizada a troca de mensagens para que o microprocessador se configure corretamente para a aquisição dos dados.

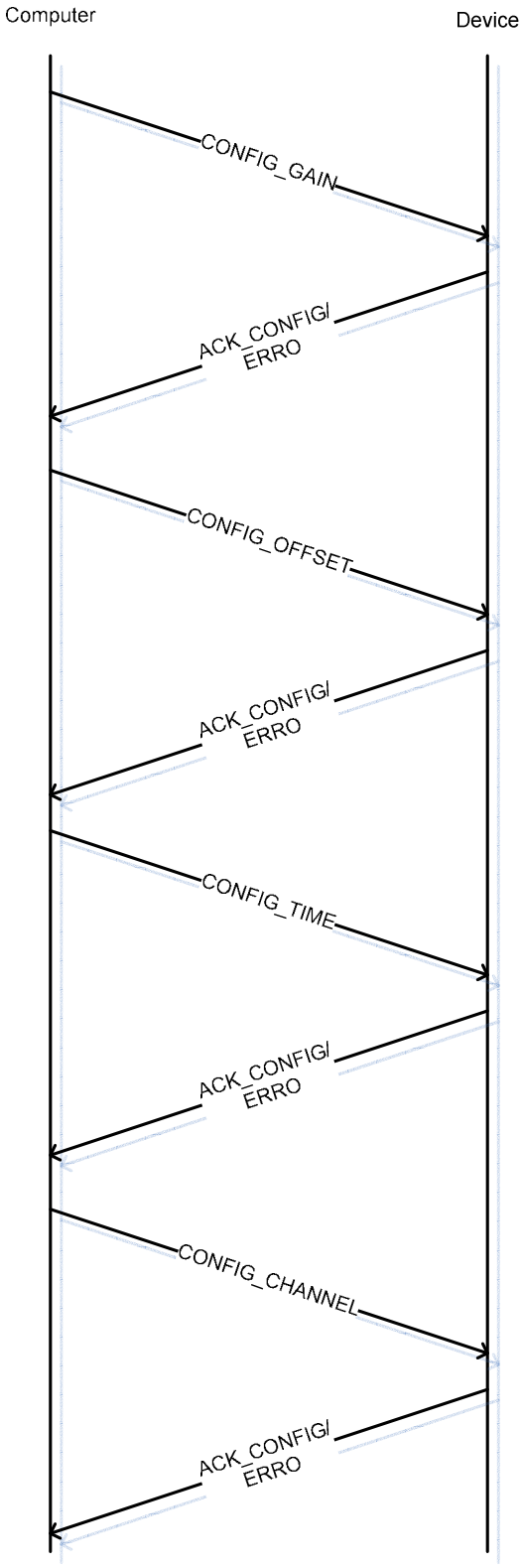

Figura 16 – Fluxo de mensagens para configuração

4.3.2.2 Envio de dados para camada de software

O processo para enviar dados, Figura 17, é semelhante ao de configuração, porém com algumas mudanças, como mostrado na figura abaixo.

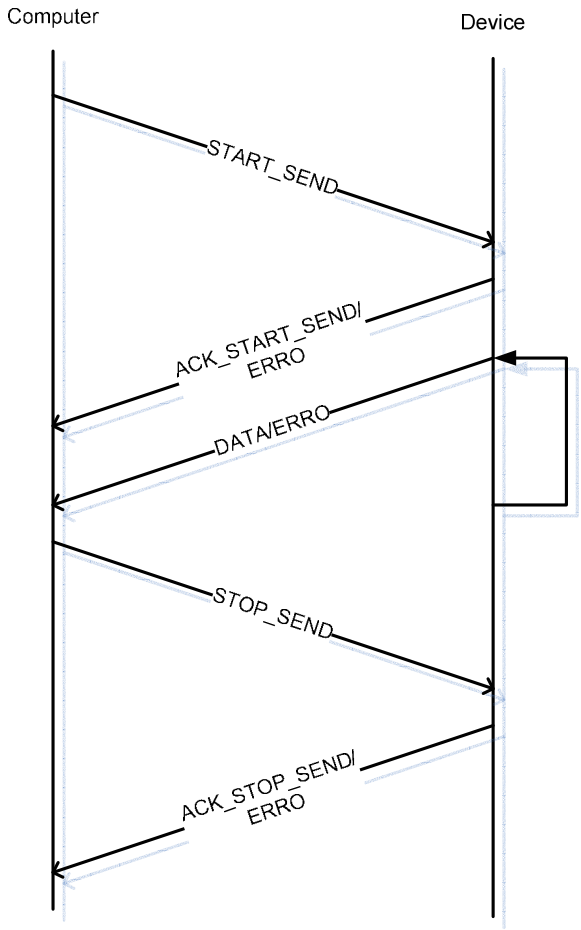

Figura 17 – Fluxo de mensagens para envio de dados

Na captura de dados, o computador envia o CODE com a constante START_SEND para o microprocessador. O microprocessador deve enviar o sinal ACK_START_SEND confirmando que recebeu o pacote corretamente ou ERRO, caso não. Após isso o microprocessador vai para o estado SEND_DATA e começa a envia os dados capturados para o computador. Os pacotes enviados

possuem o CODE com a constante DATA e o CHART com a constante do gráfico que aquele dado pertence ou ERRO caso alguma anomalia ocorra durante o processo. Quando o computador não quiser coletar mais dados manda para o microprocessador um STOP_SEND e o microprocessador envia um ACK_STOP_SEND em caso de sucesso ou um ERRO. Caso algum erro ocorra durante a troca de mensagens do protocolo, um código ERRO é enviado para o software e o microprocessador volta para o estado de WAIT. Após isso, o microprocessador retorna para o estado WAIT.

Com apenas esses dois fluxos de troca de mensagens, o sistema realiza a ponte de comunicação entre o hardware e o software. Os valores constantes utilizados para implementação do protocolo podem ser vistos da Tabela 1.

Tabela 1 - Constantes do protocolo de comunicação

Nome	Valor em hexadecimal
CONFIG_GAIN	0x02
ACK_CONFIG	0x03
START_SEND	0x04
ACK_START_SEND	0x05
STOP_SEND	0x06
ACK_STOP_SEND	0x07
DATA	0x08
CONFIG_OFFSET	0x09
CONFIG_TIME	0x0A
CONFIG_OFFSET	0x0B
ERRO	0x3F

4.4 Camada de software

O SDView, abreviação de S*preading Depression View*, é um software específico para armazenamento e visualização para experimentos de *spreading depression*. Este sistema foi desenvolvido utilizando as ferramentas profissionais que estão disponíveis no Centro de Informática[23], para maior robustez e agilidade de desenvolvimento.

4.4.1 Tecnologias usadas

Lista de todas as ferramentas utilizadas durante todo desenvolvimento do software:

- Microsoft Visual Studio 2005 [13]: IDE (integrated development environment) desenvolvido pela Microsoft para desenvolvimento de aplicações .NET 2.0. Ferramenta que ajuda muito durante o desenvolvimento de projetos de um porte mais elevado.
- Visual C# [14]: linguagem orientada a objetos, parte da plataforma .NET 2.0, utilizada para desenvolvimento do software. Linguagem bastante poderosa, com suporte as melhores praticas de programação.
- Componente ZedGraph [15]: componente *open-source* utilizado facilitar o desenho de gráficos 2D utilizando a plataforma .NET 2.0. Bastante versátil esse componente permite o desenho de curvas complicadas e com muitos pontos, com muita facilidade e rapidez. Além de ter recursos de exporta os gráficos como figura, impressão e zoom. Exemplo de um gráfico gerado pelo componente na Figura 18.

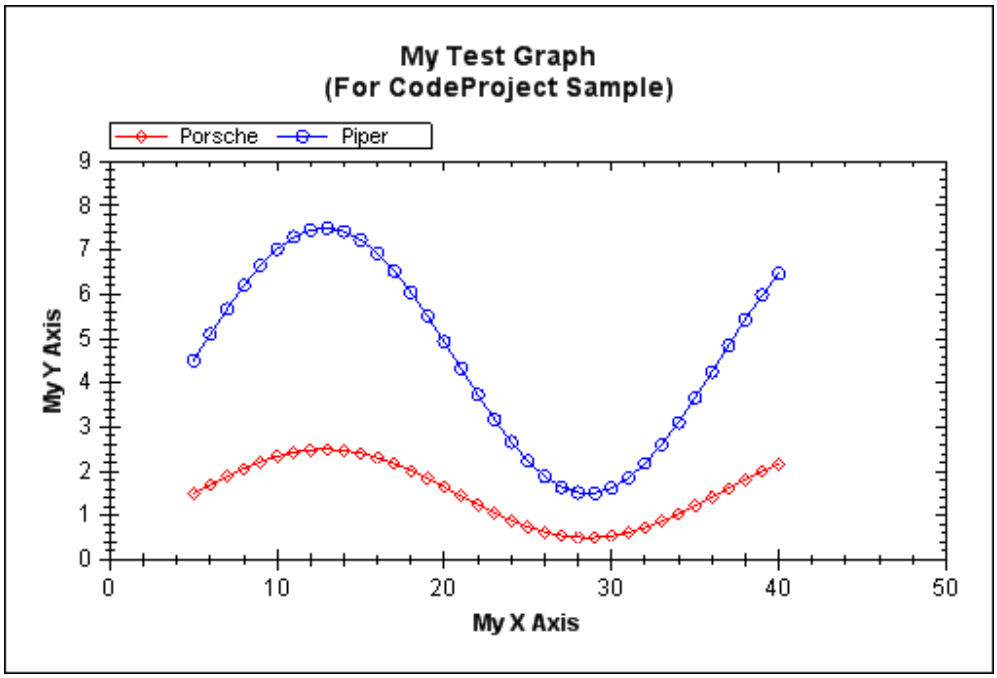

Figura 18 - Exemplo de gráfico usando o ZedGraph

4.4.2 Casos de uso

Como o software é relativamente simples do ponto de vista de interação com o usuário e outros subsistemas, os casos de uso recolhidos junto ao Departamento de Nutrição da UFPE estão representados na figura 19, abaixo.

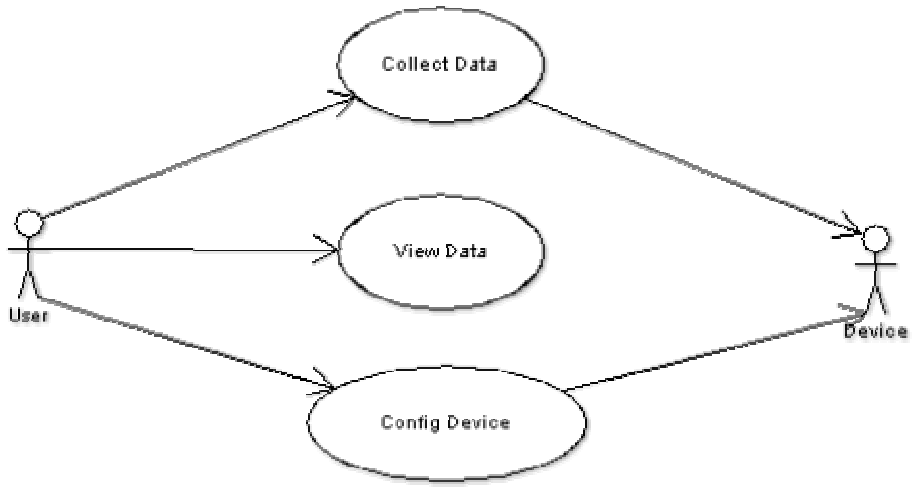

Figura 19 - Casos de uso

Descrição dos casos de uso:

- Collect Data: coleta os dados oriundos do microprocessador e armazena no computador.
- View Data: visualiza os dados dos sinais capturados pelo computador.
- Config Device: configura o microprocessador para a captura correta dos dados.

4.4.3 Descrição da estrutura do software

O padrão de projeto escolhido para o desenvolvimento do projeto foi o s*ingleton*[16]. Este padrão garante a existência de apenas uma instância de uma classe, mantendo um ponto global de acesso ao seu objeto. Como a natureza do sistema não necessitava da instância de vários tipos de objetos com muita freqüência adotei esse padrão, além dele ser mais simples para controle de erros.

Como todo projeto com um tamanho intermediário orientado a objetos, vários pacotes foram desenvolvidos para atender os requisitos que os casos de uso, mostrados na seção acima geraram. Todos os pacotes têm uma interface única que se juntam numa fachada, e essa faz a comunicação entre as camadas mais inferiores com a interface gráfica e com o usuário. Essa estrutura é muito utilizada em projetos orientados a objetos.

Uma descrição mais detalhada de cada pacote do sistema será apresentada a seguir, para um melhor entendimento do funcionamento do software.

Pacote de Comunicação:

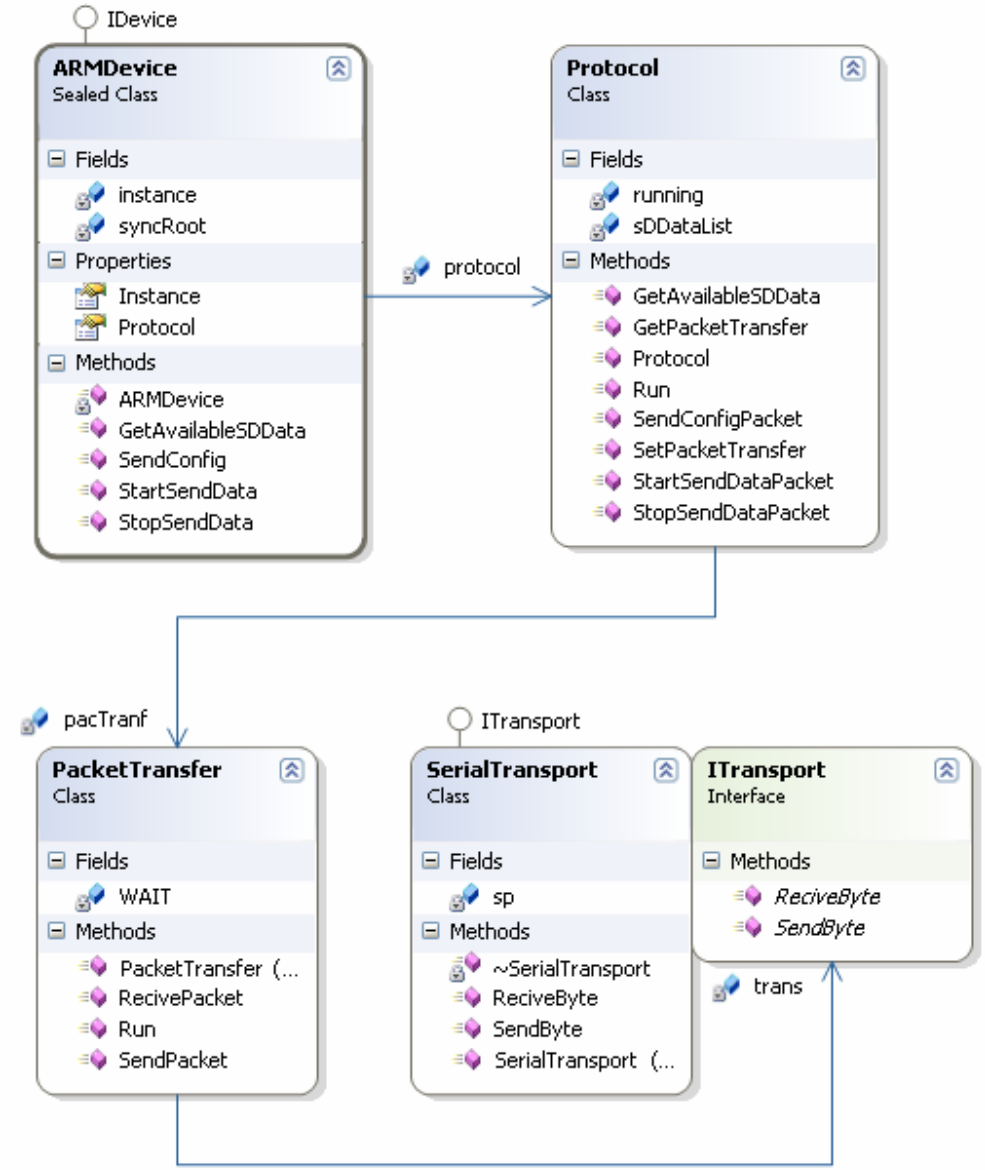

Figura 20 - Pacote de comunicação

A figura 20 apresenta a estrutura do software de comunicação, que é composto pelos módulos:

- **Classe SerialTransport:** responsável pelo transferência e recepção dos bytes através da porta serial do computador. Nela é configurada a conexão

serial com o microprocessador, como taxa de transferência de dados, nome da porta e tempo de *timeout* da transferência de dados.

- **Classe PacketTransfer:** responsável pelo recebimento e envio de pacotes pela camada de transporte. Ele quebra os pacotes recebidos em dois pedaços de oito *bits* e os enviam para a camada de transporte. Tem como atributo um ITransport tornando transparente o tipo de conexão com a camada de captura.
- **Classe Protocol:** implementa o protocolo descrito na seção 4.3.2 desse capitulo. Responsável pela troca de mensagens através dos pacotes entre o software e o hardware. Também possui um *buffer* com os dados enviados oriundos da captura de sinais.
- **Classe ARMDevice:** classe que oferece os serviços da camada de captura para a camada de software. Como configura o microprocessador para captura e começar e terminar o envio de dados.

Pacote de armazenamento de dados:

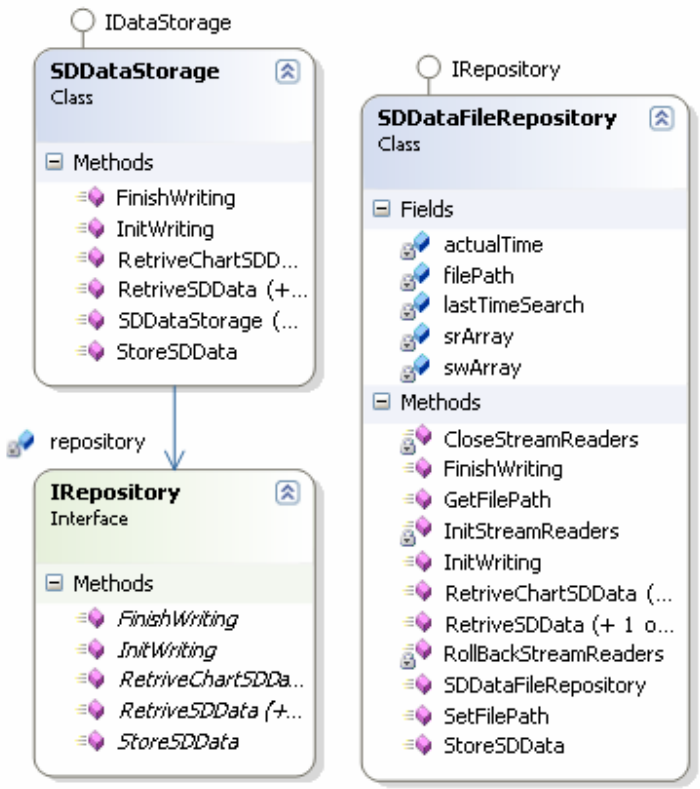

Figura 21 - Pacote de armazenamento

O pacote responsável pelo armazenamento de dados, ilustrado da Figura 21, é composto pelos módulos:

- **Classe SDDataFileRepository:** responsável por armazenar e recuperar os dados capturados pelo microprocessador. Grava os dados durante a captura e os recupera de arquivos para visualização. Implementa a interface IRepository.
- **Classe SDDataStorage:** classe que está no topo da camada de armazenamento. Deixa transparente para o software a maneira como os dados estão armazenados no computador.

Pacote de gráficos:

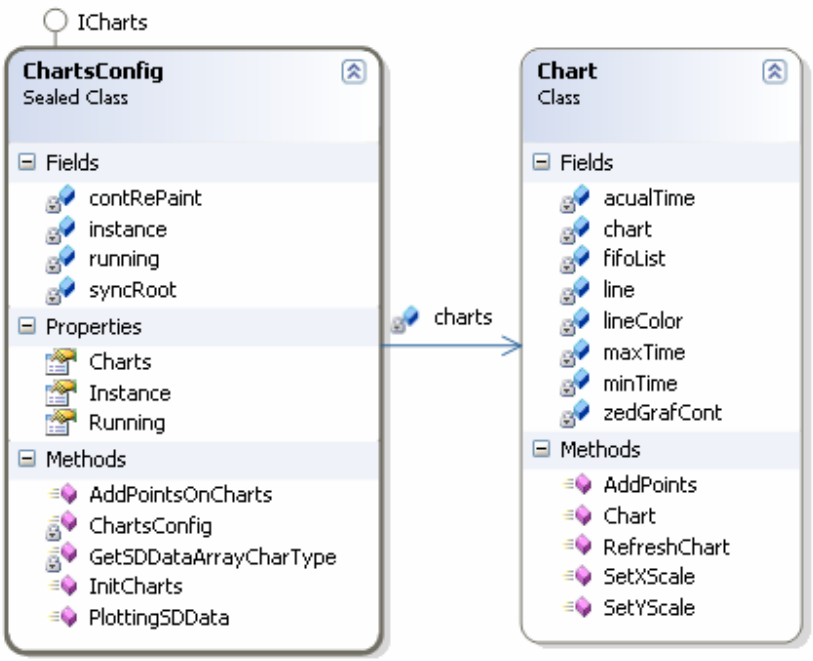

Figura 22 - Pacote de gráficos

A confecção dos gráficos resultantes das análises dos processos biológicos é desenvolvida a partir de uma série de classes apresentadas na Figura 22, descritas em detalhes a seguir;

- **Classe Chart:** classe básica que representa o objeto gráfico do componente ZedGraph[15]. Ele que faz atualizações e configura como os dados serão desenhados sobre o gráfico.
- **Classe ChartsConfig:** Coleção de charts, manipula os dados antes que eles cheguem na classe base. Distribui os dados entre os gráficos para que não haja inconsistência nos gráficos desenhados.

Fachada SDView:

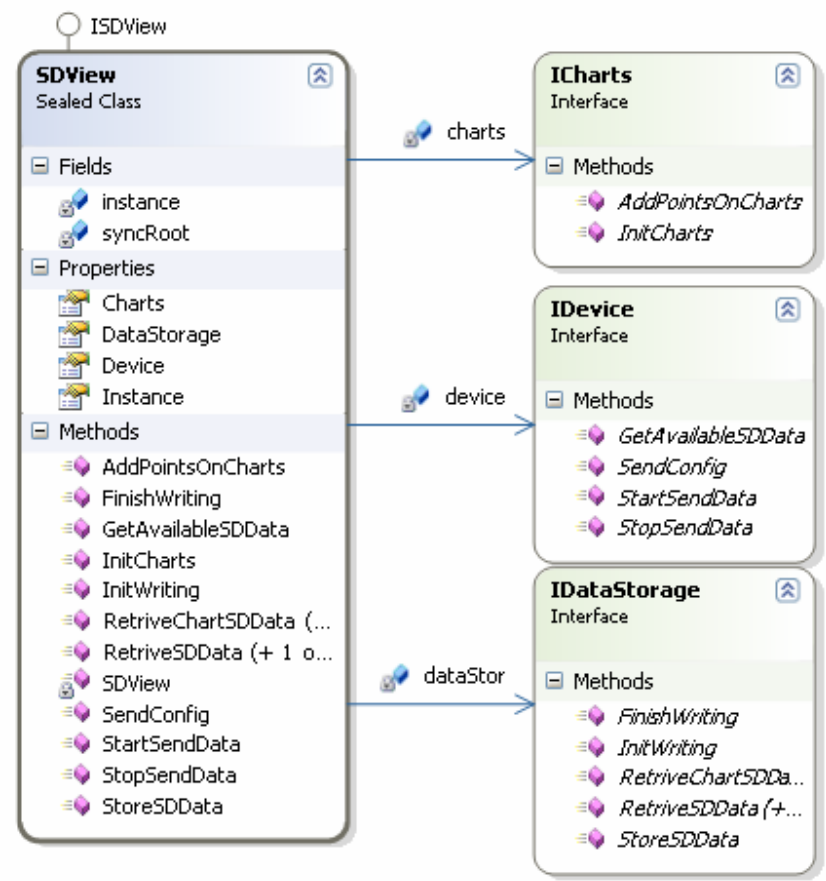

Figura 23 - Fachada do software

Classe principal do software que reúne todos os métodos que a interface gráfica deve chamar para o funcionamento do software, mostrada na Figura 23. Nela estão contidas as principais instâncias dos pacotes, que provêm todos os serviços do sistema.

Com esse sistema que envolve camadas de hardware, software e comunicação, procuramos dar uma solução para a automatização das experiências de *spreading depression* que são realizadas no Departamento de Nutrição da UFPE.

5. Resultados obtidos

Este capítulo apresentará os resultados obtidos a partir da implementação do sistema proposto no capítulo 4. Semelhante à estrutura do capítulo 4, os resultados serão apresentados de maneira similar à estrutura da arquitetura.

5.1 Captura

Da camada de captura foi montado um protótipo da estrutura apresentada com apenas um AD8555[17], ao invés dos dois como demandava a captura, como ilustrado na Figura 24. Foi montado primeiramente dessa maneira, para haver uma validação do sistema proposto, antes de haver uma montagem do sistema final.

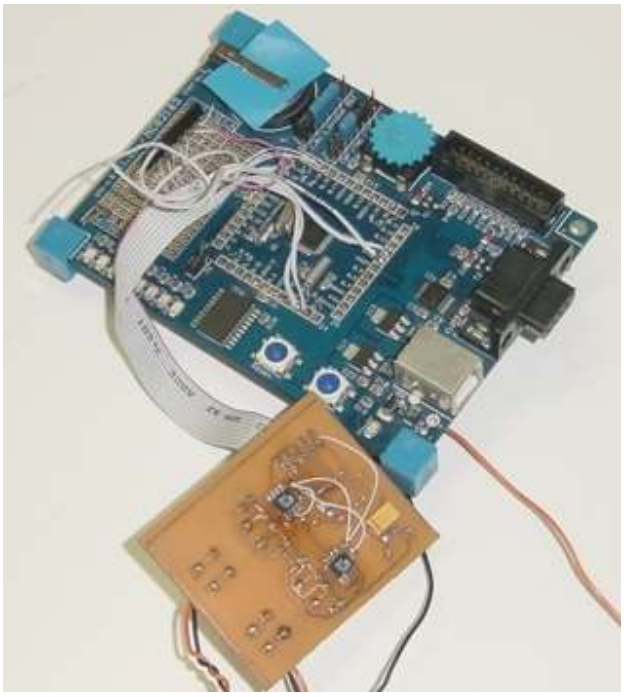

Figura 24 - Protótipo de captura

Acima da Figura 24, de azul é o microprocessador ARM[12] e abaixo o amplificador digital AD8555[17]. Os fios interligam os módulos são as conexões necessárias para a troca de informações entre o microprocessador e o amplificador, assim eles poderem trabalhar em conjunto para a captura.

5.2 Comunicação

A união entre o microprocessador e o software ainda não foi realizada, mas os dois sistemas estão implementando o mesmo protocolo e ambos foram testados isoladamente e validados, neste aspecto.

O protocolo da parte do microprocessador foi validado utilizando o µVision IDE[27], como fonte de dados para o microprocessador, que respondeu corretamente aos comandos.

O protocolo da parte software foi validado por uma emulação do microprocessador por uma máquina de estado semelhante à apresentada na seção 4.2.2.1, com a transferência dos dados via serial realizada via *loopback*[28], como ilustrado na Figura 25.

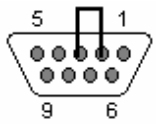

Figura 25 - *Loopback* na serial

5.3 Software

Foram desenvolvidas todas as classes que estruturam a arquitetura proposta do sistema, mas a interface gráfica apenas foi implementado um protótipo para a validação no Departamento de Nutrição, como ilustrado na Figura 26.

Após a realização de uma experiência com ela e sua validação em desenhar os gráficos a partir dos dados capturados, então melhorias na GUI serão implementadas para melhorar a visualização de acordo com as necessidades do Departamento de Nutrição.

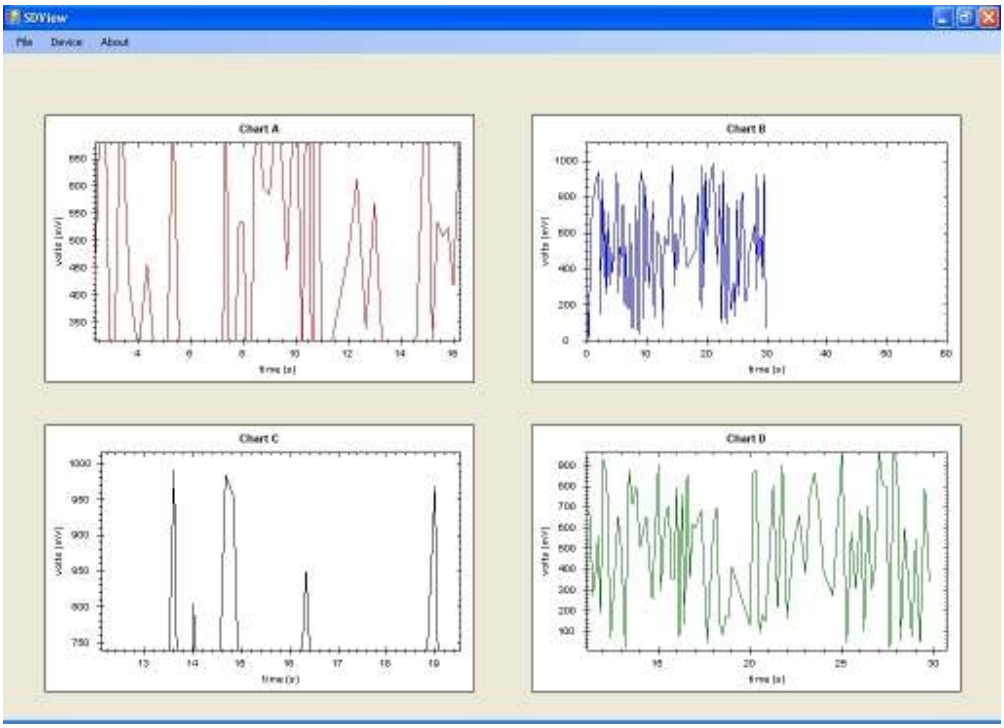

Figura 26 - Software SDView

Utilizando um *add-in* do Microsoft Visual Studio 2005[13], o Line Counter[26] que é capaz de calcular algumas métricas sobre o projeto, foram levantados os seguintes dados sobre o projeto de software SDView:

- Linhas de código: 2188
- Linhas de comentário: 594
- Número de classes: 31

6. Conclusão e trabalhos futuros

Durante esse trabalho de graduação foi possível trabalhar por várias áreas da engenharia da computação que apreendi durante todo curso, pois nele foi desenvolvido em vários módulos de diferentes áreas como mostrado no capitulo anterior. Este trabalho proveu subsídios para a implementação de uma plataforma para leitura, processamento e visualização de dados oriundos de experimentos biológicos em cobaias, voltados para experimentos de "*Spreading Depresion".*

Dessa maneira pude utilizar um pouco dos conhecimentos adquiridos durante todo curso, tanto da parte de informática como da área de engenharia eletrônica, abrindo um novo mundo de possibilidades de trabalhos que posso realizar com sucesso, a partir de agora.

Este projeto permitiu também uma maior integração entre os trabalhos em desenvolvimento no Departamento de Nutrição e o no Centro de Informática, em sistemas embarcados, desenvolvendo trabalhos de origem interdisciplinar, de grande proveito para ambas às partes em processamento digital de sinal e interfaces.

Como próximos passos para o desenvolvimento do sistema podem citar:

- Desenvolver um estudo de caso completo, com captura de dados reais;
- Melhora na interface gráfica;
- Maior robustez do sistema;
- Mudança no armazenamento dos dados para um banco de dados;
- Porte dos dados para outros programas de visualização;
- Mudança da transmissão de dados pela serial para USB;
- A partir dos dados capturados fazer análises;
- Realizar/implementar filtragens dos sinais para retirar possíveis ruídos de leitura do sistema.

Referências

[1] Processamento dos Sinais Biológicos, http://www.virtual.epm.br/material/tis/curr-med/temas/med5/med5t12000/tele/sinais_digitais.html - Último acesso em 24/03/07.

[2] Processamento de Sinais Biológicos, http://www.leb.usp.br/psignal.html - Último acesso em 24/03/07.

[3] Nutrição adequada: A base do funcionamento cerebral, http://cienciaecultura.bvs.br/scielo.php?pid=S0009-67252004000100024&script=sci_arttext - Último acesso em 24/03/07.

[4] Conversores Digital/Analógico (D/A) e Analógico/Digital (A/D), www.eletrica.ufpr.br/marlio/medidas/apostila/apostila3a.pdf - Último acesso em 24/03/07.

[5] Understanding analog to digital converter, http://www.embedded.com/showArticle.jhtml?articleID=60403334 - Último acesso em 24/03/07.

[6] Conversão Analógica Digital, www.universiabrasil.net/mit/6/6071/PDF/f02-lec20a_val.pdf - Último acesso em 24/03/07.

[7] Shiro Usui e Itzhak Amidror , Digital Low-Pass Differentiation for Biological Signal Processing, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?isnumber=4121322&arnumber=4121327&count=9&index=3 - Último acesso em 26/03/07.

[8] Conceitos básicos de filtragem digital multitaxa, http://www.maxwell.lambda.ele.puc-rio.br/cgi-bin/PRG_0599.EXE/3731_4.PDF?NrOcoSis=7083&CdLinPrg=pt - Último acesso em 26/03/07.

[9] Filtro FIR, http://pt.wikipedia.org/wiki/Filtro_FIR - Último acesso em 26/03/07.

[10] IIR, http://es.wikipedia.org/wiki/IIR - Último acesso em 26/03/07.

[11] AxoScope 9, http://www.moleculardevices.com/ - Último acesso em 27/03/07.

[12] ARM, http://www.arm.com/ - Último acesso em 27/03/07.

[12] Protocolo, http://pt.wikipedia.org/wiki/Protocolo - Último acesso em 28/03/07.

[13] Microsoft Visual Studio Official Website, http://msdn.microsoft.com/vstudio/ - Último acesso em 29/03/07.

[14] Visual C# Home, http://msdn2.microsoft.com/en-us/vcsharp/default.aspx - - Último acesso em 29/03/07.

[15] ZedGraph Wiki, http://zedgraph.org/wiki/index.php?title=Main_Page - Último acesso em 29/03/07.

[16] Singleton, http://pt.wikipedia.org/wiki/Singleton - Último acesso em 29/03/07.

[17] AD8555, Analog Devices, http://www.analog.com/en/prod/0,,759_782_AD8555,00.html - Último acesso em 29/03/07.

[18] ARM LPC2101, http://www.mct.net/download/philips/lpc2103.pdf - Último acesso em 29/03/07.

[19] Spreading Depression of Activity in the Cerebral Cortex, http://www.garfield.library.upenn.edu/classics1992/A1992JL32200001.pdf - Ultimo acesso em 29/03/07.

[20] Aristides Leão, http://cienciahoje.uol.com.br/controlPanel/materia/view/730 - Ultimo acesso em 29/03/07.

[21] Filtro passa-faixa, http://pt.wikipedia.org/wiki/Filtro_passa-faixa - Ultimo acesso em 29/03/07.

[22] Filtro passa-alta, http://pt.wikipedia.org/wiki/Filtro_passa-altas - Ultimo acesso em 29/03/07.

[23] Centro de Informática da UFPE, http://www.cin.ufpe.br - Ultimo acesso em 29/03/07.

[24] Amplificador Diferencial, http://pt.wikipedia.org/wiki/Amplificador_diferencial - Ultimo acesso em 29/03/07.

[25] Hamer CF, Ifeachor EC, Jervis BW, Digital filtering of physiological signals with minimal distortion, www.springerlink.com/index/53258L294316V143.pdf - Ultimo acesso em 29/03/07.

[26] Line Counter, a Visual Studio 2005 Add-In, http://www.codeproject.com/useritems/LineCounterAddin.asp - Ultimo acesso em 29/03/07.

[27] µVision IDE, http://www.keil.com/uvision/ - Ultimo acesso em 29/03/07.

[28] How to Do a Serial Loopback Test, http://zone.ni.com/devzone/cda/tut/p/id/3450 - Ultimo acesso em 29/03/07.