

Sistema de Aquisição de Sinais Biológicos

Rui Paulo Ferreira, Rui Pedro Alves, Ana Maria Tomé, Osvaldo Rocha Pacheco

Resumo – Neste artigo apresentamos um sistema para aquisição de sinais biológicos. O sistema desenvolvido é aberto, modular e versátil permitindo fácil adaptação aos diferentes tipos de sinais e ambientes de utilização. Por outro lado, o sistema permite também, que se integre à posteriori procedimentos de processamento de sinal, tornando-se assim, numa útil plataforma para teste de algoritmos de processamento de sinal.

Abstract - In this paper we describes an acquisition system to deal with biological signals. The system modularity and versatility allows its configuration to different environment (such as a neurophysiology laboratory). The system is also suited to test signal processing algorithms on the acquired or stored biological signals.

I - INTRODUÇÃO*

No princípio da década de 90, foi desenvolvido no DETUA um sistema para aquisição, visualização e armazenamento de sinais biológicos (*Record*). Este sistema foi desenvolvido com base num PC – compatível, equipado com uma placa de aquisição da *Data Translation*, correndo sobre o MS-DOS. O sistema tem sido usado em inúmeros estudos clínicos, nomeadamente, quando se pretende em rotina clínica o armazenamento dos sinais para estudos posteriores, e serviu de base para um conjunto de aplicações na área da telemedicina [1]. Apesar da grande utilização do sistema, este apresenta algumas limitações, em dois planos distintos. Um dos planos, tem a ver com o sistema operativo de suporte e com o *hardware* usado para a aquisição. O segundo determina um conjunto de limitações que têm a ver com opções tomadas durante o desenvolvimento, que em rotina clínica se manifestaram pouco funcionais.

As limitações derivadas do sistema operativo e da placa de aquisição, tornam o sistema pouco flexível, sobretudo no que diz respeito à aquisição, porque a utilização de novas placas implicaria a reconversão de toda a biblioteca de aquisição. Por outro lado, atendendo a que toda a programação foi desenvolvida para o MS-DOS e para plataformas computacionais pouco potentes, as bibliotecas de visualização e de aquisição foram desenvolvidas em *assembler*. A reutilização do código, a substituição da placa de aquisição, ou mesmo a introdução de novas funcionalidades, por exemplo, alteração das características da visualização, ou a introdução de algoritmos de

processamento digital de sinal, é assim para este sistema, uma tarefa algo complicada.

O segundo grupo de limitações está relacionado com a interface com o utilizador, nomeadamente, com a visualização dos sinais adquiridos ou a visualização do resultado de um qualquer processamento realizado. Para obviar este conjunto de limitações, desenvolveu-se uma aplicação que corre sobre o *Windows 95*, (*WinSigid*) [2] vocacionada apenas para a visualização, edição e arquivo de sinais biológicos. Assim, foram introduzidas novas funcionalidades para a visualização e procedimentos que permitem a selecção e armazenamento de segmentos de sinal previamente gravado. Esta aplicação é usada em rotina clínica no Hospital Geral de Sto. António em estudos de epilepsia, para segmentação e arquivo de sinais previamente adquiridos com o *Record* e tem integrado outras aplicações, onde se torna necessário a edição de sinal [3].

II – ESPECIFICAÇÃO DO SISTEMA

Com base nos sistemas referidos anteriormente, o propósito deste projecto consistiu em desenvolver um sistema que integrasse as facilidades de aquisição do *Record*, com as funcionalidades de visualização do *WinSigid*, aumentando a versatilidade, eliminando a dependência de um qualquer *hardware* específico, e que permitisse a fácil inclusão de primitivas para processamento digital de sinal, garantindo a manipulação de ficheiros de sinal, nomeadamente a selecção de segmentos com interesse clínico, para gravação no modo multi-segmento. Assim, poderemos resumir da seguinte forma, os objectivos genéricos deste trabalho:

- Desenvolvimento de uma aplicação para aquisição de sinais biológicos, que funcione sobre o sistema operativo Microsoft Windows 95.
- Estudar a utilização de linguagens orientadas à programação gráfica, nomeadamente o LabVIEW, quanto à possibilidade de suportarem o desenvolvimento de uma aplicação que permita a aquisição de um número variável de canais (até um máximo de 16), com possibilidade de termos frequências de amostragem diferentes em cada canal. Estudo de metodologias que nos garantam a independência do *hardware* de aquisição.
- Desenvolvimento de uma interface de visualização, com o mesmo tipo de funcionalidades disponibilizadas pelo *WinSigid*.

* Trabalho realizado no âmbito da disciplina de projecto.

- Desenvolvimento de módulos para armazenamento dos sinais em disco, usando o formato SIGIF [4], garantindo-se a compatibilidade com outras aplicações já desenvolvidas.

- Desenvolvimento de mecanismos, que permitam a fácil inclusão de algoritmos de compressão em tempo real e primitivas de processamento, tais como inserção de filtros digitais, cálculo de transformadas ou algoritmos de detecção automática de eventos.

As funcionalidades que se apresentam de seguida foram definidas como prioritárias.

Visualização e edição de sinal:

- Aspecto visual semelhante ao do papel.
- Amplitude em valor absoluto.
- Configuração do tempo por página.
- Selecção e gravação de multi-segmentos.
- Escalas diferenciadas.

Aquisição:

- Selecção dos canais a adquirir.
- Calibração.
- Configuração da frequência de amostragem.
- Gravação de multi-segmentos.
- Indicação do espaço em disco disponível.
- Indicação da carga computacional.

Arquivo:

- Possibilidade de integrar algoritmos de compressão de sinal.
- Gravação no formato SIGIF.

Processamento:

- Possibilidade de se integrar módulos para teste de algoritmos de processamento digital de sinal.

III - METODOLOGIAS

O LabView é uma linguagem de desenvolvimento de aplicações, semelhante a outras linguagens mais conhecidas, tais como o C ou o BASIC [5, 6, 7, 8]. A diferença fundamental entre as linguagens clássicas e o LabView resulta do facto desta última ser orientada à programação gráfica, conhecida como G, em que os programas são gerados na forma de diagramas de blocos. Cada bloco tem um ícone que indica a operação a que está associada. O diagrama de blocos é uma representação gráfica do programa que se pretende realizar, e nele se encontram presentes os ícones usados e as respectivas entradas e saídas de dados. Fazendo a ligação dos ícones de acordo com o algoritmo do problema, obtém-se o código final.

Os programas do LabVIEW são conhecidos por *virtual instruments* (VI's), porque a sua apresentação e operação imitam instrumentos reais. Os VI's têm simultaneamente uma interface visual interactiva e o diagrama de blocos correspondente ao código fonte. A interface interactiva de

um VI é conhecida por *front panel*, porque simula o painel de um instrumento físico. O *front panel* pode conter selectores, botões de pressão, gráficos e outros controlos e indicadores que concretizem as entradas e as saídas do VI.

Como exemplo, apresentamos nas figuras 1 e 2, o *front panel* e o diagrama de blocos de um VI. Este VI executa um ciclo *for* de 0 a 9 indicando através de dois indicadores se o número é par ou ímpar. Como se pode verificar, os indicadores presentes no *front panel* encontram-se representados no diagrama de blocos.



Figura 1 - Front Panel.

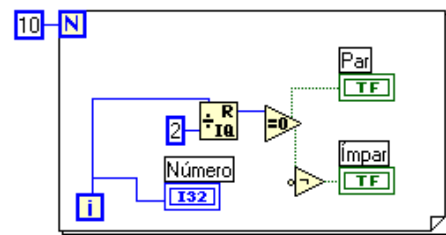


Figura 2 - Diagrama de blocos.

O LabView tem extensas bibliotecas de funções e subrotinas que cobrem a maior parte das tarefas de programação para além de incluir as clássicas instruções do tipo *for*, *while* e *case*, entre outras. Contém também bibliotecas específicas para aquisição de dados, VXI, GPIB e controlo série de instrumentos, análise de dados, representação de dados e armazenamento. Inclui ainda ferramentas de desenvolvimento convencionais, por forma a que se possam criar *breakpoints*, fazer a animação da execução, para ver a forma como os dados fluem ao longo do programa, ou ainda para fazer o *debug* do programa.

Os VI's são hierárquicos e modulares, podendo portanto, ser usados como programas ou como subprogramas dentro de outros programas. Um VI dentro de um outro VI é conhecido por Sub-VI. O ícone e o conector de um VI trabalham como uma lista de parâmetros gráficos que permitem que outros VI's possam passar ou receber dados do SubVI. Assim, a programação em LabVIEW terá forçosamente que ser modular, em que um problema complicado se pode dividir em pequenas tarefas que correspondem a um Sub-VI. Pelo facto de se poder executar cada Sub-VI separadamente, o *debug* é relativamente fácil de realizar. Muitas vezes um Sub-VI pode executar subtarefas comuns a várias aplicações,

podendo-se, desta forma, criar bibliotecas de Sub-VI's para as aplicações que se pretendam construir.

LabVIEW e as DLL's

Dynamic linking é um mecanismo de ligação entre aplicações e livrarias na fase de execução (*run time*). Desta forma as livrarias permanecem em ficheiros separados, não sendo copiadas para os ficheiros executáveis das aplicações.

O LabVIEW para o Windows 95 tem a função *Call Library Function* que pode ser usada para aceder a *DLL's* de 32-bits, permitindo, assim, o desenvolvimento de funções noutras linguagens. Uma das vantagens do uso de *DLL's* é a de se poder programar a um nível mais baixo, retirando daí uma maior rapidez na execução de funções e permitindo ainda, que se possam utilizar módulos já desenvolvidos e testados noutras linguagens.

A função *Call Library* suporta uma grande variedade de tipos de dados e de protocolos de comunicação entre aplicações e livrarias, possibilitando o uso da maioria das livrarias *standard* do Windows 95.

Utilização de placas de aquisição com o LabVIEW - Driver NI-DAQ

O *driver NI-DAQ* é uma *DLL* que fornece ao LabVIEW uma interface de alto nível para comunicação com um vasto leque de placas de aquisição ou com hardware de condicionamento de sinal. Este *driver* pode ser configurado para o tipo de placa utilizada (usando um software específico (*wdaqcf32.exe*)), tornando desta forma o software desenvolvido independente da placa usada.

Descrição da placa de aquisição AT-MIO-16E-10

A série *AT-MIO-16E-10* é a primeira da National Instrument inteiramente *junper free*, sendo a configuração e a calibração realizadas através do software de configuração referido. As principais características desta placa são:

- 16 canais de entrada normais, que se podem configurar como 16 entradas pseudo-diferenciais ou 8 diferenciais e em que cada um dos canais pode ser seleccionado através de software;
- entradas unipolares (0 a +10 V) ou bipolares (-5 a +5 V);
- a ADC é de aproximações sucessivas e de 12 bits;
- possibilidade de auto-calibração após a variação da temperatura;
- frequência máxima de amostragem é de 100K amostras/segundo;
- configuração independente (por software) do ganho de cada canal, permitindo assim, para os diferentes canais, a utilização da resolução máxima da ADC.

Desenvolvimento de Sub-VI's para gerar e ler o formato SIGIF

Para que o software desenvolvido fosse compatível quer com o *WinSigid*, quer com as restantes aplicações desenvolvidas no DETUA para processamento de sinais biológicos, foi necessário criar Sub-VI's que permitissem a leitura de ficheiros já existentes no formato SIGIF e a criação de novos ficheiros neste mesmo formato. Assim, foram usadas duas técnicas em conjunto:

- criação de *DLL's* a partir da livreria existente para a manipulação de ficheiros no formato SIGIF, o que permite a reutilização de livrarias já existentes;
- criação de Sub-VI's à base de estruturas próprias do LabVIEW, que implementem funções semelhantes.

Para a criação de ficheiros SIGIF optámos por usar a segunda técnica, uma vez que o cabeçalho é constituído somente por *strings*. Para a leitura de ficheiros SIGIF, optámos por usar *DLL's*, pois a interpretação dos dados é um pouco complexa e o código necessário estava disponível. Os Sub-VI's desenvolvidos, poderão ser usados em trabalhos futuros, nesta área, em que a linguagem de desenvolvimento seja o LabVIEW.

IV – DESCRIÇÃO DAS FUNCIONALIDADES

Nesta secção apresenta-se, de uma forma sucinta algumas das funcionalidades da aplicação *WinSigid - Record Utility*, desenvolvido no âmbito deste projecto, para o sistema operativo Microsoft Windows 95.

Menu Principal

O menu principal disponibiliza quatro opções que permitem a selecção da aplicação pretendida (figura 3). As aplicações disponíveis no presente são:

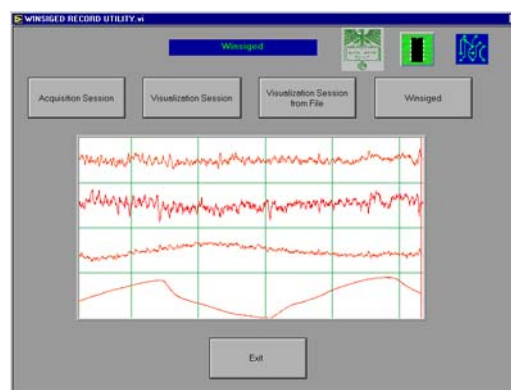


Figura 3 - Menu Principal.

- **Aquisição:** esta aplicação destina-se à aquisição e gravação de sinais biológicos em disco.
- **Visualização:** permite visualizar os sinais após a aquisição sem efectuar a gravação.

- Visualização de Ficheiros: permite a visualização dos sinais a partir de um ficheiro em disco, simulando uma aquisição em tempo real.
- *Winsiged*: executa a aplicação *Winsiged*.

Sessão de Aquisição

Neste modo de operação, o utilizador terá que seguir uma série de etapas, destinadas a configurar a aquisição pretendida. Essas etapas são as seguintes:

- Calibração;
- Selecção dos canais a visualizar e a enviar para ficheiro;
- Introdução do nome do ficheiro;
- Visualização e gravação dos segmentos;

Após a tomada de opções referida, a visualização decorre no painel apresentado na figura 4.



Figura 4 – Painel de aquisição.

Sessão de Visualização

A sessão de aquisição apresentada no ponto anterior, permite a visualização sem gravação em disco, sendo no entanto necessário proceder à calibração, o que não é muito prático quando apenas se pretende visualizar os diferentes sinais. Para simplificar este procedimento, foi criada a sessão de visualização. Nesta sessão o utilizador pode escolher um conjunto específico de canais para a visualização; o aspecto gráfico é idêntico ao apresentado na figura 4.

Sessão de Visualização de Ficheiros

Na sessão de visualização de ficheiros previamente adquirido, o utilizador terá que realizar um conjunto de opções como sejam: indicação do nome do ficheiro, selecção dos canais guardados no ficheiro e indicação dos segmentos a visualizar. O painel principal da sessão de visualização é apresentado na figura 5.

WinSinged

A sessão de visualização apenas integra um subconjunto das funcionalidades disponibilizadas pelo *WinSinged*.

Assim, para que se possa usar de uma forma integrada as diferentes aplicações, achou-se conveniente introduzir uma opção que permita a execução do *WinSinged*.

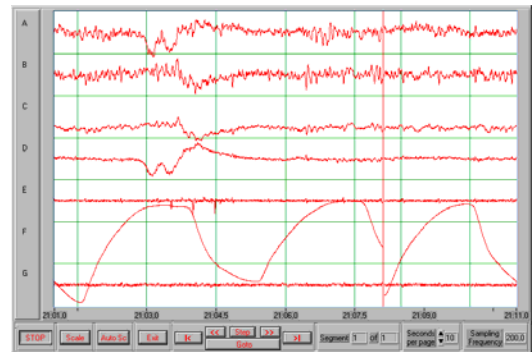


Figura 5 – Painel principal da sessão de visualização a partir de ficheiros.

VI - CONCLUSÕES E TRABALHO FUTURO

Dos requisitos inicialmente propostos, apenas não foi possível concretizar a aquisição de canais com diferentes frequências de amostragem. A placa de aquisição escolhida apenas tem uma ADC, que é partilhada no tempo por todos os canais, impedindo assim, a aquisição a multifrequência.

No desenvolvimento deste trabalho, foram mantidos os requisitos do sistema tomado como base, tais como a aparência visual e o formato de gravação dos sinais. Foi testada a inclusão de alguns algoritmos para processamento de sinal, quer através da introdução de VI's desenvolvidos especificamente para o efeito, quer através da reutilização de algoritmos através da criação de DLL's. Verificou-se, como seria de esperar, que com as DLL's a execução é mais rápida. Desta forma, parece lícito concluir, que o sistema constitui uma boa plataforma para testar o funcionamento em tempo real de algoritmos de processamento digital de sinal.

O desenvolvimento futuro deste sistema, passará por uma primeira fase que consistirá na validação clínica do sistema e uma segunda pelo desenvolvimento em várias vertentes, tais como:

- Introduzir procedimentos de comunicação, permitindo em ambiente hospitalar, que a aquisição e a análise dos sinais se realize em diferentes salas. Esta implementação poderá também permitir a realização de exames em regime ambulatorio. Com este objectivo, foram realizados alguns testes, usando a livreria de TCP / IP fornecida pelo LabView, tendo os resultados sido satisfatórios.
- Introdução de técnicas de compressão de sinal.
- Introdução de algumas funcionalidades, para a adaptação da aplicação ao tipo de sinal biológico a adquirir.
- Desenvolvimento de um *driver* para a *DT2821*; com a utilização do sistema *Record* foi adquirido um elevado número de placas de aquisição *DT2821* da *Data Translation* (a única compatível com o sistema), assim, seria interessante desenvolver um *driver* que

permitisse a utilização destas placas no novo sistema desenvolvido, rentabilizando-se o conjunto de placas existente.

REFERÊNCIAS

- [1] Francisco Vaz, Osvaldo Rocha Pacheco, António Martins da Silva - "A Telemedicine application for EEG signal transmission", 13th Annual International Conference of the IEEE/ EMB SOCIETY, 1991, Orlando, USA.
- [2] Osvaldo Pacheco, Ricardo Conceição, Filipe Tranco, Ana Maria Tomé - "Visualization and Storage of Biological Signals in Neurophysiology Laboratory ", 19th Annual International Conference IEEE/EMB SOCIETY, 1997, Chicago, USA
- [3] José Miguel Ramos, Nuno Filipe Santos - "Telemedicina Aplicada à Neurofisiologia Clínica", Relatório de Projecto. DETUA. 1997.
- [4] M. B. Cunha, J. P. Cunha, T. O. Silva, "Sigif: A Digital Signal Interchange Format for Biological Signals", 15th Annual International Conference IEEE Engineering in Medicine and Biological Society, San Diego, USA, 1993.
- [5] LabVIEW for Windows Tutorial. NATIONAL INSTRUMENTS. 1996.
- [6] LabVIEW for Windows. User Manual. NATIONAL INSTRUMENTS. 1996.
- [7] LabVIEW for Windows. GPIB and Serial Port VI Reference Manual. NATIONAL INSTRUMENTS. 1996.
- [8] LabVIEW for Windows. Analysis VI Reference Manual. NATIONAL INSTRUMENTS. 1996.