

Sistema Integrado de Aquisição de Dados para Potenciometria

Rui Manuel Gonçalves, Sérgio Santiago e Santiago, António Rui Borges, Artur Carneiro Pereira
 DETUA
 Armando Duarte, Isabel Boal
 DQUA

Resumo - Este artigo descreve um sistema integrado de aquisição de dados para potenciometria desenvolvido sobre o sistema operativo Windows 3.1 da Microsoft. Este sistema é basicamente constituído por um computador ligado através das suas portas série a uma bureta e a um potenciómetro. A comunicação com estes periféricos é *interrupt-driven*.

Uma vez que se destina a utilizadores com poucos conhecimentos na área da computação o sistema foi concebido por forma a ser o mais intuitivo e o mais interactivo possível e, por forma a permitir um controlo o mais apertado possível sobre o processo de aquisição.

Abstract - This paper describes an integrated data acquisition system for potentiometry. It is intended to run under Microsoft Windows 3.1. The system assumes a hardware set up consisting of a PC connected (through its serial ports) to a potentiometer and to a burette. The communication with these two devices is interrupt driven.

Since it aims to be used by non-computer experts, it creates a user friendly environment which enables a tight control over the acquisition process.

1. INTRODUÇÃO*

A. Potenciometria

A utilização de técnicas potenciométricas na área da instrumentação química permite estimar a concentração de iões específicos em soluções. Um procedimento comum consiste em adicionar volumes sucessivos (constantes ou variáveis) de um determinado produto a uma solução e medir, após cada adição, o potencial existente num eléctrodo aí mergulhado. As curvas, assim obtidas, são depois processadas para a extracção de parâmetros característicos da solução e/ou da reacção que teve lugar.

Este projecto visa automatizar o processo de medida e construir um ambiente integrado que possibilite aos utilizadores um controlo tão apertado quanto possível da aquisição de dados, quer ao nível da variação de parâmetros específicos (volumes do produto a adicionar, tempos de estabelecimento, etc.), quer a nível de pré-processamento *on-line* dos valores já adquiridos.

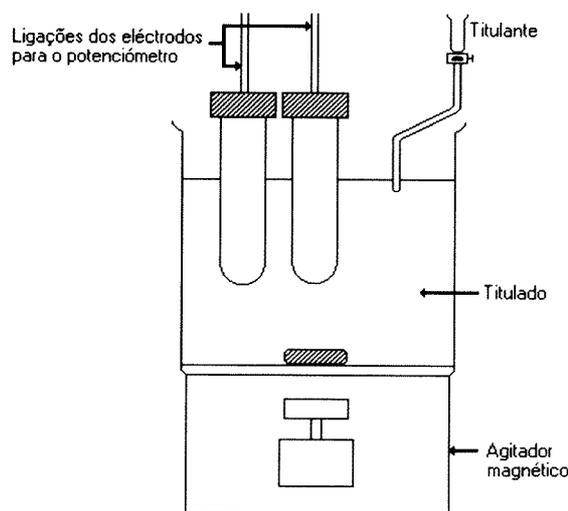


Fig. 1

Esquema de uma Titulação Potenciométrica.

B. Sistema Operativo Windows 3.1

O sistema operativo Windows 3.1 da Microsoft permite uma infínitá variedade de possibilidades no projecto e concepção de aplicações. Quase todo o projecto das aplicações é feito sem o conhecimento arquitectónico da máquina, sem preocupações quanto ao interface com o monitor, impressoras, rato, etc...

O Windows 3.1 oferece ao programador toda uma nova filosofia de programação. Com o evoluir do software e hardware dos computadores, começa a deixar de ser necessário o projecto de aplicações diferentes para máquinas diferentes. No Windows 3.1 existe todo um novo modelo de programação, todo um novo conceito de ambiente de programação e um infínitá número de novos conceitos e termos que se tornarão, de algum modo, de uso comum no futuro.

Em Windows 3.1 é permitido que diferentes aplicações corram concorrentemente, e é permitido às aplicações o partilhar de dispositivos de entrada/saída, memória, vídeo e do CPU. Há toda uma série de funções de utilização e acesso a estes dispositivos à disposição do programador. O uso destas funções, para além de (nalguns casos) facilitar o trabalho do programador, garante o não conflito entre aplicações no acesso a dispositivos comuns. O Windows 3.1 funciona como um árbitro sobre as aplicações, gerindo a ocupação de cada recurso e o tempo de ocupação por parte de cada aplicação. No caso da comunicação através das portas, o Windows 3.1 dispõe de

* Trabalho realizado no âmbito da disciplina de Projecto.

funções que permitem a utilização de *buffers* de tamanho variável para cada porta, e a comunicação é *interrupt-driven*.

Porém, apesar do Windows 3.1 ser um ambiente *multitasking*, é também *nonpreemptive* pelo que uma aplicação deve partilhar o CPU e não deve tentar tomar o seu controlo exclusivo. O Windows 3.1 não poderá retomar o controlo do CPU a não ser que a aplicação lho devolva.

O Windows 3.1 só pode receber ou enviar informação duma ou para uma aplicação através de eventos. O Windows 3.1, quando um evento é gerado, "lê-o" e envia uma mensagem à aplicação, à qual pertence o objecto sobre o qual ocorreu o evento. Todas estas mensagens vão para um *buffer* do tipo FIFO, denominado *message queue*.

Este tipo de *multitasking* é um método cooperativo que confia no facto das aplicações, elas próprias, partilharem alternadamente os recursos do sistema.

Em Windows 3.1 uma aplicação não deve aceder à memória ou a dispositivos entrada/saída directamente (sem o controlo do Windows).

No projecto de uma aplicação deve-se fazer com que o utilizador tenha a sensação que a controla, e não que é controlado por ela, ou seja, a aplicação deve ser moldada de modo a adaptar-se ao utilizador e não o contrário.

A aplicação deve ser o mais INTERACTIVA possível e deve ser facilmente ADAPTÁVEL às exigências do utilizador em aspectos como a sua aparência (cor) e funcionabilidade (menus).

O interface da aplicação deve ser totalmente TRANSPARENTE ao utilizador, de modo a permitir que este apenas faça uso dela sem que para isso seja necessário um conhecimento profundo do seu funcionamento.

O interface deve ser o mais INTUITIVO possível, e a aplicação deve actuar de modo a que o utilizador se aperceba de todas as suas acções e respectivas consequências sem ter que ler o manual da aplicação.

A CONSISTÊNCIA da aplicação deve ser vista de dois pontos: do ponto de vista do utilizador - CONSISTÊNCIA EXTERNA - e do ponto de vista do Windows - CONSISTÊNCIA INTERNA. A CONSISTÊNCIA EXTERNA da aplicação deve ser baseada na experiência do utilizador e no seu habitual ambiente de trabalho. A CONSISTÊNCIA INTERNA da aplicação refere-se a manter a filosofia de funcionamento e utilização das aplicações em Windows.

Todo o texto, elementos visuais e ideias devem ser organizados de modo a criar um écran visualmente agradável, sem por isso distrair o utilizador. As cores escolhidas devem contrastar entre si, mas não devem agredir visualmente o utilizador, e devem ser escolhidas em função do seu gosto e das condições de luz do ambiente de trabalho. Todos os objectos semelhantes devem ser mantidos juntos e sempre que possível devem criar-se comandos tridimensionais e de fácil utilização.

O utilizador deve ter *feedback* imediato das suas acções, ou seja, se seleccionar um objecto ou uma parte de texto (num editor), o écran deve revelar esta selecção. Embora

este *feedback* gráfico seja primordial, o *feedback* textual ou áudio podem também ser usados.

Erros do utilizador não devem ser fatais, pois o utilizador normalmente usa tentativas e erros para aprender a usar uma aplicação. A possibilidade de escolhas erróneas deve ser minimizada, e sempre que um erro ocorra a sua correcção deve ser facilmente efectuada pelo utilizador. As mensagens de erro, que normalmente são transmitidas ao utilizador pela aplicação, devem, para além da informação de erro, especificar resumidamente o modo de o ultrapassar, sempre que tal seja possível.

Deve utilizar-se, sempre que possível, as potencialidades do sistema e da aplicação, em vez das do utilizador.

Todo o projecto da aplicação deve ser feito em função do utilizador e da facilidade de utilização, e não só da funcionalidade. Os testes de utilização são primordiais e críticos para o sucesso e aceitação de uma aplicação. Uma boa apresentação gráfica, e até áudio, da aplicação pode impressionar o utilizador numa primeira análise superficial, porém, se a aplicação não for cómoda e facilmente inteligível e utilizável, fará com que o seu sucesso seja limitado.

II. DESCRIÇÃO DO AMBIENTE

O ambiente de trabalho foi concebido de modo a ser o mais interactivo e intuitivo possível e de modo a reduzir ao mínimo a intervenção do utilizador no processo de aquisição.

Os parâmetros iniciais da aquisição podem ser modificados durante o decorrer da mesma, possibilitando ao utilizador um controlo tão apertado quanto possível do decorrer da experiência. Os valores adquiridos são, além de visualizados no écran, armazenados (juntamente com outros valores relativos à experiência) num ficheiro. O utilizador tem também a possibilidade de armazenar, num outro ficheiro, toda a "história" do processo de aquisição efectuado.

Os valores de potencial dos eléctrodos são adquiridos por um potenciómetro em mV e são visualizados no écran quer em forma de texto, quer em forma gráfica, permitindo deste modo uma mais fácil monitorização da aquisição.

A. Filosofia da Aquisição

Todo o processo de aquisição de valores e cálculo de incrementos é transparente ao utilizador. Este apenas tem de escolher a estratégia a seguir durante a experiência, quanto a, nomeadamente critérios de estabilidade a aplicar aos valores lidos, modo de cálculo dos incrementos a adicionar à solução, decisão de fim de experiência e selecção dos eléctrodos a utilizar.

Existem dois tipos de critérios de decisão quanto à estabilidade dos valores lidos. Estes critérios são mutuamente exclusivos.

Pode-se, por um lado, considerar um valor final estável se a sequência de valores médios obtida num dado intervalo de tempo (Tempo de Estabilização) estiver toda contida no intervalo delta - critério de intervalo fixo. Por outro lado, e este será outro critério de decisão, pode-se considerar um valor final estável quando se obtém uma sequência de valores médios (Nº de Valores Médios) todos iguais - critério de intervalo variável.

Porém, pode dar-se o caso de o tempo de espera para obtenção de um valor estável, segundo um dos critérios atrás definido, seja demasiado grande ou mesmo infinito (o sistema é intrinsecamente instável). Há, pois, toda a vantagem em estabelecer-se um tempo máximo de decisão (Tempo Máximo de Medida). Assim, se se atingir este tempo máximo de decisão sem se obter a estabilidade, o valor final é calculado também a partir da sequência de valores médios, e é considerado instável.

Para uma maior fiabilidade dos valores finais, os valores a partir dos quais se toma a decisão, são eles próprios médias de um número (Nº de Medicões (Média)) de valores lidos directamente do potenciómetro.

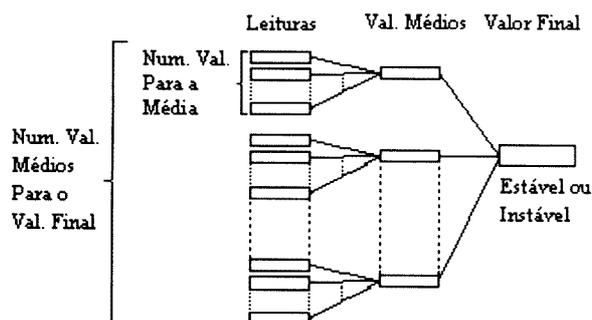


Fig. 2

Método para o Cálculo de Valores Finais.

O cálculo dos incrementos de volume de titulante a adicionar à solução pode seguir dois critérios, também mutuamente exclusivos.

O primeiro critério é o de impor um incremento constante. O outro critério é o de definir um valor máximo para a variação de potencial pretendido e um valor máximo para o incremento de volume a adicionar à solução, impondo assim os limites de definição da curva. Se a variação de potencial entre dois valores consecutivos não ultrapassar a definição imposta, o novo incremento a adicionar será o imposto como máximo. Se a variação de potencial ultrapassar a definição imposta, o novo incremento é calculado segundo a interpolação linear indicada na Fig. 3.

Estes valores de incremento de volume são adicionados à solução através de uma bureta ligada ao computador

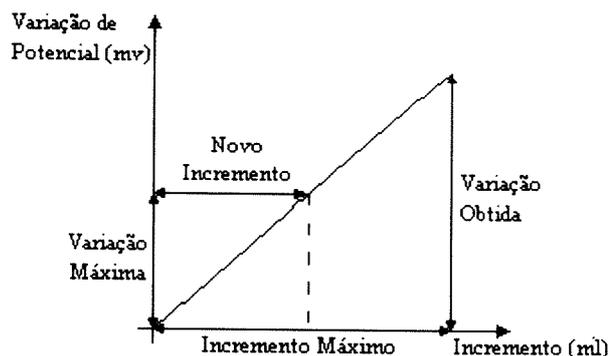


Fig. 3

Método de Cálculo de Incrementos Variáveis.

O fim da experiência é imposto quando pelo menos uma das três condições seguintes é verificada:

- ◆ Foi atingido o valor esperado como final da experiência.
- ◆ Foi atingido o número máximo de pontos de amostragem especificado.
- ◆ Não se obteve uma variação de potencial considerada mínima.

B. Reagentes e parâmetros da reacção

O utilizador introduz os dados referentes aos reagentes que vão intervir na reacção e os parâmetros que a controlam através de dois quadros normalmente designados por *Dialog Boxes*.

Dialog Boxes são janelas que são utilizadas para efectuar o "diálogo" entre uma aplicação e o utilizador. Os seus campos podem ser acedidos através da utilização do rato e/ou do teclado. Para além de campos de edição, as *Dialog Boxes* possuem dois botões: o botão Ok. e o botão Cancelar. O botão Ok faz a validação de todos os dados introduzidos. O botão Cancelar apenas fecha a *Dialog Box*, cancelando qualquer alteração nos dados que tenha sido efectuada pelo utilizador.

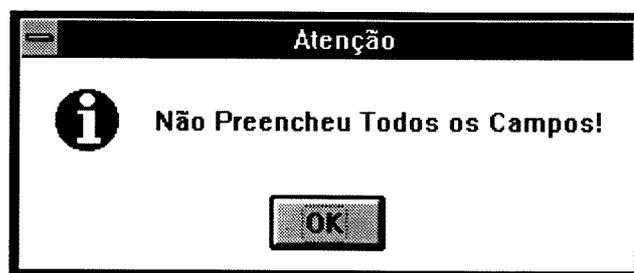


Fig. 4

Message Box Usada para Notificar o Utilizador da Falta de Preenchimento dum Campo de Edição.

Todos os campos de edição numa *Dialog Box* têm que estar preenchidos para que os dados introduzidos sejam considerados válidos. Isto é, se o utilizador tentar sair da *Dialog Box* premindo o botão Ok, é confrontado com uma mensagem de aviso sendo o foco colocado num campo

não preenchido. Esta mensagem de aviso ao utilizador é feita através de uma *Message Box* representada na Fig. 4.

A *Dialog Box* que permite ao utilizador a introdução dos dados relativos aos reagentes da experiência intitula-se "Reagentes - Inicialização" e o seu aspecto é representado na Fig.5. O acesso a esta *Dialog Box* só é possível no início de cada experiência.

Todos os valores introduzidos nos campos desta *Dialog Box* são apenas de interesse informativo para o utilizador, e são colocados à cabeça do ficheiro de armazenamento de dados.

A *Dialog Box* intitulada "Parâmetros de Inicialização e Estabilização" permite ao utilizador a introdução dos parâmetros de controlo da experiência, e está representada na Fig. 6.

Na caixa "Estabilidade", pode ser escolhido o critério de estabilidade, introduzidos os valores dos parâmetros a ele referentes e introduzido o valor do tempo necessário ao estabelecimento da reacção.

Na caixa "Eléctrodos", estão os campos de selecção dos eléctrodos que se pretende utilizar no decorrer da aquisição. Estes campos só se encontram disponíveis na definição inicial das condições da aquisição. Após o arranque desta, o seu estado não pode mais ser modificado.

Na caixa "Tipo de Incrementos", o utilizador pode introduzir o valor do incremento inicial de volume a adicionar à solução e escolher o tipo de incrementos a fazer posteriormente.

Fig. 5
Dialog Box "Reagentes - Inicialização".

Fig. 6
Dialog Box "Parâmetros de Inicialização e Estabilização".

Na caixa "Condição de Terminação", o utilizador define as condições de decisão do fim de experiência.

Os valores introduzidos nos campos desta *Dialog Box* não podem ser "absurdos". Ou seja, não podem ser valores que façam com que a aquisição se torne impraticável (Ex: O tempo de estabilização tem de ser tal que permita pelo menos uma medição e, por outro lado, não pode ser maior que o tempo máximo de medida). Se o utilizador tentar introduzir um valor "absurdo", esse valor é corrigido e o foco direccionado para o campo de edição onde ocorreu o erro e respectiva correcção, dando assim hipótese ao utilizador de alterar o valor inserido pela aplicação.

C. Armazenamento de dados

Para efectuar o armazenamento dos dados referentes à reacção, de maneira a que se possa mais tarde processá-los, é permitida ao utilizador a introdução do nome do ficheiro para o armazenamento. Este procedimento é feito na *Dialog Box* "Armazenamento de Dados", onde o utilizador poderá também escolher o armazenamento, ou não, de toda a "história" da reacção. Para além dos reagentes, parâmetros e valores finais que são armazenados no primeiro ficheiro, o ficheiro da "história" da reacção contém também todos os valores médios correspondentes a cada valor final.



Fig. 7
Dialog Box Armazenamento de Dados.

D. Visualização dos resultados

O utilizador possui dois modos diferentes de monitorização do processo de aquisição no écran, um em forma de texto, e o outro em forma de gráfico. Assim, poderá acompanhar o desenrolar da experiência, e poderá controlá-lo através da mudança dos parâmetros.

Na janela "Gráfico", o utilizador poderá visualizar a curva da variação do potencial medido no(s) eléctrodo(s) mergulhado(s) na solução em função dos incrementos de volume de titulante adicionados. Esta janela de visualização possui no canto inferior direito um esquema de cores dos pontos e segmentos de recta de modo a que o utilizador possa diferenciar a(s) curva(s) referente(s) ao(s) eléctrodo(s) usado(s) e os respectivos pontos finais estáveis e instáveis.

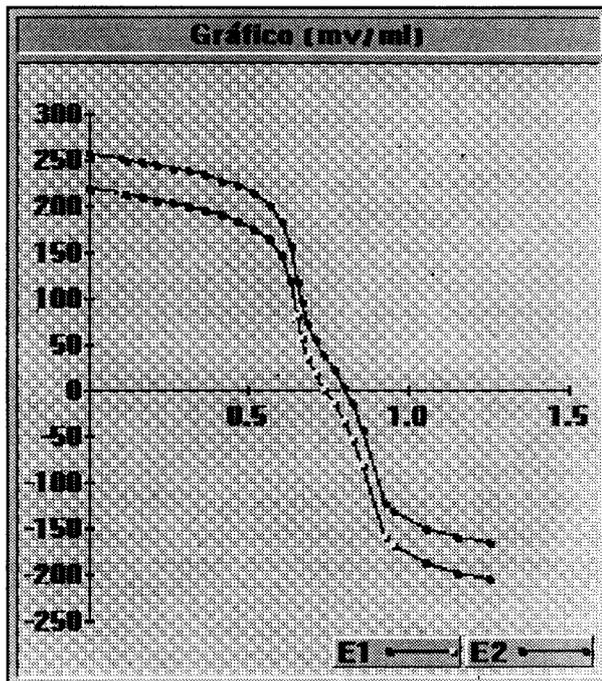


Fig. 8
Janela de Visualização "Gráfico".

Medições Efectuadas	
v= 0.948 ml	
E1= -156.6 mv dp1= 00.21	
E2= -128.3 mv dp2= 00.51	
E1= -186.0 mv E2= -145.1	
E1= -186.0 mv E2= -145.1	
v= 1.048 ml	
E1= -186.0 mv dp1= 00.00	
E2= -147.8 mv dp2= 00.00	
E1= -196.1 mv E2= -156.1	
E1= -196.0 mv E2= -156.1	
v= 1.148 ml	
E1= -196.1 mv dp1= 00.01	
E2= -157.9 mv dp2= 00.00	
E1= -202.5 mv E2= -160.3	
E1= -202.4 mv E2= -160.3	
v= 1.248 ml	
E1= -202.4 mv dp1= 00.01	
E2= -164.5 mv dp2= 00.00	

Fig. 9
Janela de Visualização "Medições Efectuadas".

Na janela "Medições Efectuadas" o utilizador poderá visualizar a listagem dos valores do potencial medido no(s) eléctrodo(s) mergulhado(s) na solução, para os correspondentes incrementos de volume de titulante adicionados. Para além dos valores finais, o utilizador poderá também visualizar os valores médios de potencial

correspondentes. Os valores são representados em cores diferentes de modo a se diferenciar valores médios de valores finais de potencial e valores finais estáveis de valores finais instáveis.

Esta janela possui uma barra de *scroll* vertical, permitindo ao utilizador um deslocamento dos valores na janela, de maneira a poder visualizar alguns valores passados.

III. FUNCIONAMENTO DO SISTEMA

Ao iniciar a aplicação é efectuada a abertura da comunicação com a bureta e o potenciômetro através das portas série do computador. Se ocorrer um erro na abertura da comunicação, o utilizador é avisado da sua ocorrência e a aplicação é abandonada e volta ao seu estado minimizado (*Icon*).

Depois de aberta a comunicação com a bureta e o potenciômetro, o utilizador tem à sua disposição quatro comandos na barra de menu da janela principal da aplicação: "Reagentes...", "Parâmetros...", "Começar" e "Sair".

Activando os comandos "Reagentes..." ou "Parâmetros...", o utilizador tem acesso às *Dialog Boxes* "Reagentes - Inicialização" ou "Parâmetros de Inicialização e Estabilização" respectivamente.

Ao activar o comando "Começar" é iniciado o processo de inicialização da aquisição. É testada a ligação do potenciômetro e da bureta, ao computador e à alimentação. Enquanto a aplicação detectar falhas nas ligações, avisa o utilizador para as confirmar.

Caso a bureta não esteja cheia quando do início da aplicação, o utilizador tem a possibilidade de reche-la pois a aplicação confirma a sua pretensão em fazê-lo ou não. Por outro lado se o utilizador não mudou os valores dos reagentes ou parâmetros antes de activar o comando "Começar", a aplicação permite-lhe ainda antes de começar a aquisição a possibilidade de o fazer, ou então se o desejar começa com os valores de defeito armazenados na última experiência.

Ainda antes do arranque propriamente dito da aquisição, é pedido o nome do ficheiro de armazenamento de dados. Dá-se então início à aquisição.

O potenciômetro é colocado no modo de leitura em mv e as janelas de visualização dos resultados são abertas.

Faz-se a leitura do potencial inicial da solução e se o utilizador optou por fazer um incremento inicial de volume de titulante na solução, procede-se ao seu adicionamento. É esperado um tempo de estabelecimento da solução (ou não, dependendo da opção do utilizador), é feita a leitura do potencial da solução, é decidida a estabilização dos valores e é calculado o novo incremento de volume a adicionar à solução, segundo os critério escolhidos. Repete-se todo este procedimento até se atingir uma das condições de decisão de fim de experiência.

Se o utilizador desejar fazer uma pausa na aquisição, activa o comando "Temporariamente" do sub-menu "Parar" do menu da janela da aplicação. Pode em seguida recomeçar no ponto em que tinha deixado a experiência, activando o comando "Recomeçar".

O utilizador pode forçar o fim de experiência em qualquer ponto da aquisição, activando o comando "Fim de Experiência" do sub-menu "Parar". Ao activar este comando a aquisição pára mas o utilizador continua a poder visualizar os dados da reacção agora terminada. Se desejar realizar uma nova experiência activa o comando "Nova Experiência" da barra de menu da janela da aplicação e o processo é reiniciado.

Se o utilizador desejar abandonar a aplicação em qualquer ponto da aquisição pode fazê-lo através do comando "Sair", e todos os valores adquiridos até esse momento ficam armazenados no ficheiro de armazenamento de dados.

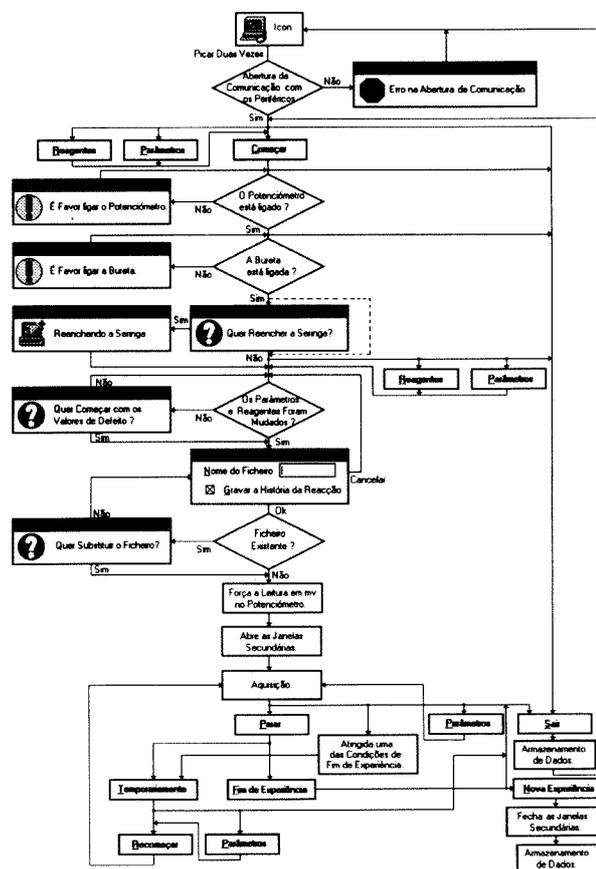


Fig. 10
Diagrama de Funcionamento.

IV. ESTRUTURA DO PROGRAMA

O programa está dividido basicamente em dez grandes módulos, de modo a permitir uma melhor estruturação, e um melhor acesso a zonas do programa onde se pretendam fazer alterações.

Assim, as rotinas de registo das janelas principal e secundárias da aplicação, de abertura e fecho da janela da aplicação, e de processamento das mensagens enviadas a estas janelas constituem o módulo principal. Entre as mensagens a gerir, a mais importante para esta aplicação é a do relógio do sistema, uma vez que as leituras e o seu processamento são feitos em ciclos de relógio definidos. A declaração das variáveis globais também é feita neste módulo.

As rotinas de processamento de mensagens das *Dialog Boxes* "Reagentes - Inicialização", "Parâmetros de Inicialização e Estabilização" e "Armazenamento de Dados" estão em três módulos separados. A leitura de todas as variáveis de entrada e o controlo sobre a introdução de valores absurdos é feito nestes módulos.

Todas as rotinas comuns às *Dialog Boxes*, como as de colocação destas no centro da janela da aplicação e as de controlo da inserção de caracteres não válidos nos campos de edição das mesmas foram colocadas num outro módulo independente.

Para cada janela de visualização de resultados ("Gráfico" e "Medições Efectuadas") existe um módulo com as rotinas de processamento das respectivas mensagens associadas. Nestes módulos estão também as rotinas de escrita nas respectivas janelas, que são usadas quando é enviado pela janela principal uma mensagem de aviso de escrita, depois do processamento dos valores estar feito.

Num outro módulo do programa, estão as rotinas de cálculo de incrementos a adicionar à solução, a sua correspondência em impulsos a transmitir à bureta e cálculo de médias e desvios padrão de potencial.

As rotinas de abertura e inicialização da comunicação com a bureta e com o potenciómetro e de leitura e escrita nas portas de comunicação e as rotinas que tem com função gerir o armazenamento dos valores lidos em disco e em memória constituem outros dois módulos.

V. COMENTÁRIOS E CONCLUSÕES

O sistema operativo Windows 3.1 da Microsoft, não é o mais indicado para sistemas de tempo real, devido às características do seu *multitasking nonpreemptive*. A mensagem do despertador do sistema (*timer*) é a mensagem de menor prioridade, e uma vez chegada ao *message queue* o seu envio pode ser atrasado pela chegada de outras mensagens. Assim um sistema de aquisição de dados que se queria periódico deixa de o ser. E, com o acumular de mensagens do despertador no *message queue* fazem-se por vezes medidas consecutivas, quando se desejava que houvesse um tempo de espera entre elas. Este problema de acumulação de mensagens no *message queue* pode ser evitado, desligando-se o despertador sempre que a mensagem por ele enviada é recebida pela função que faz o seu processamento e tornando a ligá-lo no fim desse mesmo processamento. Porém este método não resolve o problema da não periodicidade das medidas, podendo os atrasos no envio das mensagens do

despertador serem minimizados pelo utilizador ao não fazer correr outra aplicação enquanto utiliza o Sistema Integrado de Aquisição de Dados para Potenciometria, no entanto este método vai contra a filosofia do sistema operativo em questão. O método de resolver estes problemas seria o de fazer com que o despertador envia-se a mensagem directamente para uma função encarregada de a processar, sem passar pelo *message queue*.

REFERÊNCIAS

- [1] Tim Farrell, Runnoe Connally, "Programming in Windows 3.1", Second Edition, Programming Series, QUE.
- [2] Rogerio Saran, "Programando em Windows 3.1", Livros Érica Editora Ltda.
- [3] Brian W, Kernighan, Dennis M. Ritchie, "The C Programming Language", Second Edition, Prentice Hall Software Series.
- [4] Peter A. Dornell, Philip E. Margolis, "C: A Software Engineering Approach", Springer - Verlag.
- [5] Mark Szamrej, "Custom Toolbars", Windows/DOS Developer's Journal, February 1993, vol. 4, Nº 2, pp. 47-57.
- [6] John Caufield, "Dialog Boxes on the Fly", Windows/DOS Developer's Journal, February 1993, vol. 4, Nº 2, pp. 25-38.
- [7] John Reynolds, "Real-World Device Control", Windows/DOS Developer's Journal, March 1993, vol. 4, Nº 3, pp. 6-22.
- [8] Guy Eddon, "Monitoring Hardware Interrupts in Real Time", Windows/DOS Developer's Journal, February 1993, vol. 4, Nº 3, pp. 23-28.
- [9] Microsoft Corporation, "Run-Time Library Reference", Microsoft C/C++ Version 7.0.
- [10] Orion Research Incorporated, "Laboratory Products Group Bench Top pH/ISE Meter Instruction Manual".