

Electrónica e Telecomunicações

universidade de aveiro



AVEIRO • SET. • VOL. 1 • Nº 2

Revista do Departamento de Electrónica e Telecomunicações da Universidade de Aveiro

Electrónica e Telecomunicações

Revista do Departamento de
Electrónica e Telecomunicações
da Universidade de Aveiro

Editorial

Editores:

Francisco Vaz
José Luis Oliveira

Comissão editorial:

Alexandre Mota
Ana Maria Tomé
Aníbal de Oliveira Duarte
António Ferrari de Almeida
António Nunes da Cruz
António Sousa Pereira
Atílio Gameiro
Dinis Magalhães dos Santos
Fernando Ramos
Joaquim Arnaldo Martins
João Pedro Estima de Oliveira
José Alberto Fonseca
José Alberto Rafael
José Carlos Neves
José Carlos Pedro
José Ferreira da Rocha
José Rocha Pereira
Nelson Pacheco da Rocha
Maria Beatriz de Sousa Santos
Paulo Jorge Ferreira

Morada e Secretariado:

Departamento de Electrónica e
Telecomunicações
Universidade de Aveiro
Campo Universitário
3800 Aveiro
Portugal

Artes Gráficas: Sérgio Cabaço

Impressão: Gráfica Aveirense

Tiragem: 600 exemplares

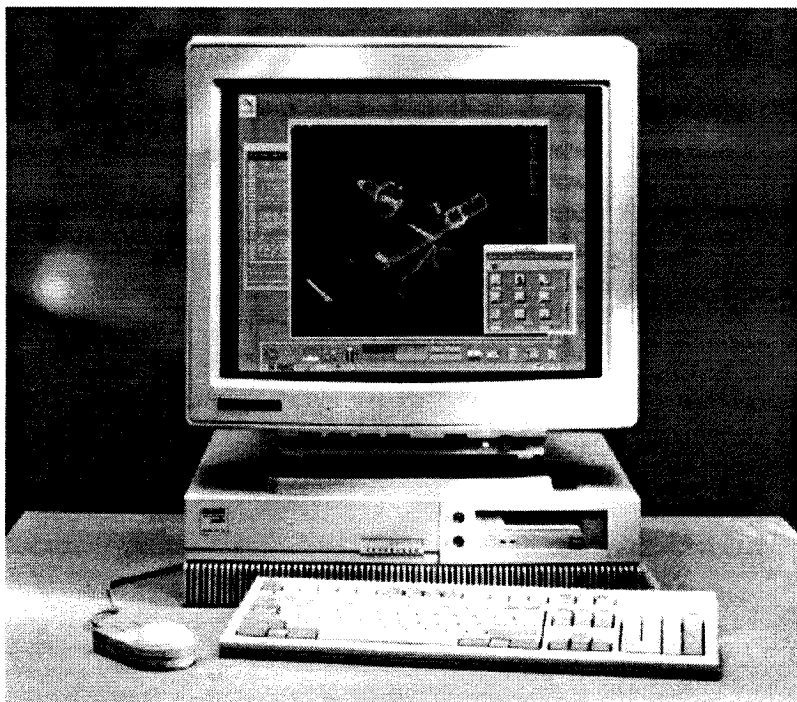
A publicação do primeiro número de Electrónica e Telecomunicações conseguiu atingir alguns dos objectivos referidos no editorial de abertura. Com efeito passámos a ter um instrumento para divulgar as actividades do DETUA e deu-se um passo importante para que os seus membros passem a relatar os seus trabalhos publicamente e de uma forma sistemática. Não é de admirar pois, que num muito menor período de tempo, tenham sido apresentados trabalhos em número suficiente para permitir esta edição e, tudo indica, um terceiro número no início do próximo ano. O objectivo de se ter uma publicação com periodicidade semestral parece pois estar ao nosso alcance.

A difusão da revista foi feita internamente e um número significativo de alunos adquiriu o primeiro número, embora não se atingisse o que desejávamos. Gostaríamos de vir a criar condições para uma maior colaboração do corpo discente nos próximos números, para que possa ser considerada uma obra de "todos" e se torne num elemento importante na vida académica dos estudantes de Engenharia Electrónica e Telecomunicações.

Neste número continuam a ser maioritários os artigos onde se descrevem os resultados obtidos em teses de doutoramento e mestrado e em projectos do 5º ano da licenciatura. Apresentamos também alguns artigos que relatam a investigação levada a cabo em projectos de âmbito nacional e internacional em que participaram docentes e alunos do DETUA.

Agradecemos todas as críticas e chamadas de atenção para gralhas, erros de composição e defeitos de impressão que o primeiro número apresentou. Esperamos que desse ponto de vista tenhamos progredido no bom sentido e continuamos a contar com os comentários dos leitores para melhorar a qualidade desta revista.

Workstations HP - A Potência ao seu alcance



Se necessita de potência para executar as suas tarefas a um preço acessível, as Workstations Hewlett-Packard são a sua solução.

Nos modelos da série 700, com várias opções disponíveis, diferentes performances e capacidades, poderá certamente encontrar a melhor configuração para o seu problema.

Contacte-nos para mais informações.



	712/60	712/80	715/64	715/80	715/100	725/75	735/125
Packaging	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop
Clock	60Mhz	80Mhz	64Mhz	80Mhz	100Mhz	75Mhz	125Mhz
SPECint92	58.1	84.3	66.6	83.5	100.1	80.3	136
SPECfp92	85.5	122.3	96.5	120.9	137	126.8	201
MIPS	73	92	78	97	122	92	154
MFLOPS	13	30.6	24.3	30.5	37.8	31	57.4
Cache	64KB shared	256KB shared	256KB shared	256KB shared	256KB shared	256KB shared	256KB shared
RAM	16 - 128MB	16 - 128MB	32 - 256MB	32 - 256MB	32 - 256MB	32 - 256MB	32 - 400MB

COMPANHIA PORTUGUESA DE COMPUTADORES INFORMÁTICA E SISTEMAS, LDA

Sede Rua Costa Cabral, 575 - 4200 PORTO Telf.- 02/5570010 Fax - 02/5504540

Delegação Lisboa Largo da Lagoa, N° 8 E 2795 LINDA-A-VELHA Tel. 01/4141800 Fax /4040801

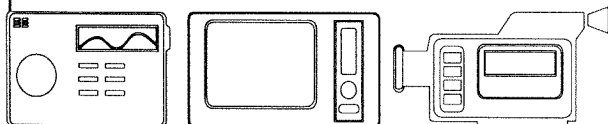
Índice

Portuguese Participation in the RACE TeleCommunity Project <i>Nelson Pacheco da Rocha, Ernesto Afonso, Pedro Breda, Manuel Machado, Leonor Moniz Pereira, Clara Cidade</i>	97
Ethernet Traffic Measurements: A Case Study <i>José Luís Oliveira, Joaquim Arnaldo Martins, Joaquin Sousa Pinto</i>	103
Linearização de Amplificadores de Potência por Realimentação Activa <i>José Carlos Pedro, Jorge Perez, José Rocha Pereira</i>	109
Determinação de Mapas de Actividade Cerebral numa Banda de Frequências <i>Carlos Loff Barreto, Francisco Vaz</i>	119
Sistema de Processamento Digital de Imagem para WINDOWS <i>Anabela Godinho, Osvaldo Santos, A. Sousa Pereira</i>	123
Receptor de Baixo Custo e Elevada Linearidade para Sistemas de Televisão por Cabo Óptico até ao Assinante (FTTH) <i>Paulo Grilo Domingues, José Carlos Pedro</i>	129
Impact of Non-zero Extinction Ratio on Optically Pre-amplified Receivers <i>L.F. Botelho Ribeiro, J.R. Ferreira da Rocha, J. Lemos Pinto</i>	135
Amplificador de Instrumentação de Sinais Biológicos <i>Rui Martins, Francisco Vaz</i>	139
A Simulator Function Library for the SWIFT LAN Manager Prototype <i>Armando J. Pinho, Fernando M. S. Ramos</i>	149
Uniformização de Interface de Programação para Aplicações Suportadas em Ambiente RDIS ou TCP/IP <i>Ilídio Ramalho, Fernando M.S. Ramos, Joaquim Arnaldo Martins</i>	153
Sistema Integrado de Tele-imagiologia <i>Luís F.C. Figueiredo, Fernando M.S. Ramos, A. Sousa Pereira</i>	161
Sistema de Televigilância Suportado na RDIS <i>José P.O. Santos, Fernando M.S. Ramos, Osvaldo A. Santos</i>	169
Voice Integration in a TCP/IP/Ethernet Based LAN <i>Joaquim Azevedo, Joaquim Arnaldo Martins, Pedro Guedes de Oliveira</i>	181

TV LAR

DE

João Sousa Ferreira



TUDO PARA ELECTRONICA EM GERAL

Acessórios de Rádio,
TV e Video

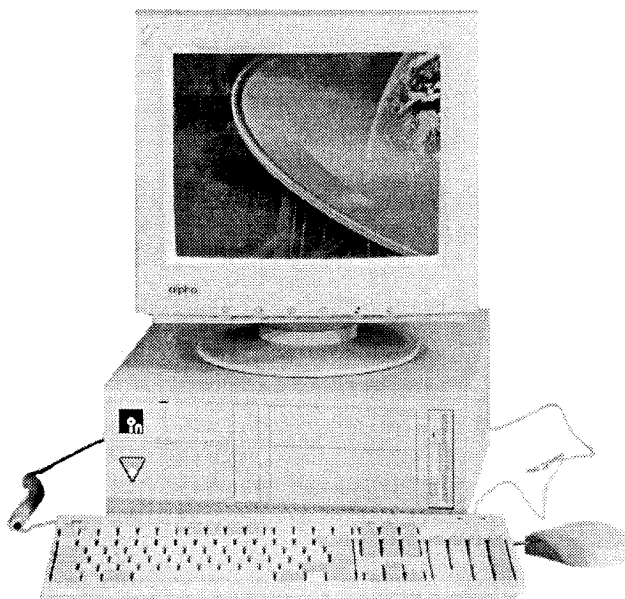
Rua Luís Gomes de Carvalho, nº 35

Tel. 21012 3800 - AVEIRO

Computadores Evolutivos

Arquitectura VESA Local-BUS
Placa aceleradora com 1 Mb em Local-BUS
Monitor SVGA 0.28 Low Radiation
Teclado KeyTronic 102 teclas Português
Rato KeyTronic Ergonómico

inSystems



inforlandia

Sistemas e Serviços de Informática, Lda.

Sede: Av. Dr. Lourenço Peixinho, 105 - 3800 AVEIRO
Tel: 034.29409 - Fax: 034.382412 - BBS: 034.382467

AG

Apple
Macintosh
Prosa, Programação
Informática
S.A.
Prosa Concessionário de Ano

1994

PROSA E **PowerPC**

Uma Dupla Imbatível

Quem é o primeiro
não perde um segundo



Em Aveiro Ligue 383 002/3

No Porto Marque 200 3242/54

Em Lisboa Contacte 386 24 04/36/91

Portuguese Participation in the RACE TeleCommunity Project

Nelson Pacheco da Rocha, Ernesto Afonso, Pedro Breda, Manuel Machado

DETUA

Leonor Moniz Pereira

FMH

Clara Cidade

TLP

Resumo - O projecto TeleCommunity, no contexto do programa RACE, é um conjunto de experiências de comunicação avançada que tem por objectivo o estudo da implementação de novos serviços de comunicação para pessoas deficientes. O artigo apresenta os serviços de apoio social e o equipamento terminal desenvolvido para a experiência portuguesa no âmbito do TeleCommunity.

Os serviços (Aprendizagem e Treino, Supervisão e Acompanhamento de Casos, Colaboração na Tomada de Decisões e Comunicação Interpessoal) foram projectados com a finalidade de explorar o potencial da RDIS para melhorar o apoio social a deficientes, nomeadamente idosos, deficientes visuais e deficientes mentais.

Os equipamentos terminais são terminais multimedia modulares, baseados em computadores pessoais com codecs, os quais apresentam a capacidade de providenciar, em simultâneo, videofonia ponto-a-ponto e ligações de dados. Diversos terminais foram distribuídos por diferentes Instituições portuguesas.

Abstract - The TeleCommunity project in the context of RACE program is a set of Advanced Communications Experiments that aim to study the implementation of advanced communications services for disabled people. The paper presents the support services and the terminal equipment developed for the Portuguese experiment within the TeleCommunity Project.

The services (Learning and Training, Remote Delivery of Expertise, Collaborative Decision Making and Interpersonal Communication) have been designed to exploit the potential of ISDN to support and improve the professional care of disabled people, namely elderly, visual and mental impaired.

The terminal equipment are modular multimedia terminals, based in personal computers with video codecs, that present an integrated capability to set-up, simultaneously, point-to-point videophony and data connections. Various terminals have been distributed among several Institutions, which are major Portuguese social care Institutions.

I. INTRODUCTION

The technological evolution of multimedia systems will help to shape the human environment in the near future. A

multimedia system is characterised by putting together images, video, audio, text and numeric processing in a desktop computer. The ideal multimedia system should be capable of handling both motion video and synchronised audio. For the full-motion video presentation, systems should be able to process and display several frames per second, together with the sound. Those systems should also record (live video from a camera and sound from a microphone, for instance), compress the data and store it on disk. Finally, they may be capable of transmitting data, real-time video and audio across a digital communication network. This digital communication network could be the Integrated Services Digital Network (ISDN).

ISDN is a full digital communication network designed to provide access to all telecommunication services, using a set of standardised access interfaces, and whose cable infra-structure is compatible with present telephone subscriber lines.

Those technologies present big potential of changing the environment and can contribute in a very positive way for a more independent life and a better social integration of disabled people.

The intervention for disabled people should comprise aspects such as restoring the loss of functions, compensating for the disability by enhancing other skills, compensating for disability through the use of technical and not-technical aids and changing the environment to adapt it to the skills of the person. The transmission of life and still pictures, sound and computer graphics have a great potential in developing systems that enable better co-ordination between the needs of the population and the intervention that is provided to them. The TeleCommunity project in the context of RACE II (Research and technology development in Advanced Communications technologies in Europe) is a set of Advanced Communications Experiments (ACEs) [1] which aim to carry out pre-competitive and pre-standardisation work in advanced communications and is concerned with how the transition to Integrated Broadband Communication (IBC) affects the implementation of advanced services for disabled people.

These experiments comprise field trials conducted in real-life scenarios and operate in Portugal, Germany,

Finland, United Kingdom, Ireland, Sweden, Norway, Belgium and the Netherlands.

The Portuguese ACE exploits the potential of ISDN in the support of the care of disabled people, namely visual, mental impaired and elderly. It provides different services that allow the target users the access to specialised support.

The terminal equipment for those services are modular multimedia terminals, based in personal computers with video codecs. Each terminal offers an integrated capability to set-up, simultaneously, point-to-point videophony and data connections, typically between a service provider and a client, providing the means to transfer and process image, voice and data, which is adequate to interactive remote care services.

II. END USERS

In certain countries there is a trend to integrate mental impaired people into the work place and home environment. This puts pressure on the rehabilitation programmes and there is a strong need for systems to support this process as most of these people require individual training. The relatively cheap personal computers with good graphics, high resolution, colour and sound facilities together with usable and cheaper software and telecommunication facilities are able to offer remote learning and training aids such as identification, repetition, monitoring of their learning progress and better visual and hearing possibilities, which can be used for memory reinforcement, task learning and sequencing, identification of new concepts and application to problem solving.

The visual impaired people can have, through telecommunication, access to specialised remote care. The remote care can be the support in social interaction skills and community living, occupational therapy and training, support and advice concerning domestic management and everyday tasks, assistance in maintaining education, improved access to other services and their acquisition and management.

The world geriatric population is increasing, mainly in industry countries. The increase of life expectancy and the reduction of birth rate caused modifications in population distribution according to age. Dealing with all the varied and additive problems acquired with age, means that elderly people require increasing assistance with everyday matters. A general trend amongst support organisations is to assist in the continuance of independent living since it is a common aim of the elderly: it is more resource intensive to move people into residential care, and institutionalisation often accelerates the process and consequence of ageing. New technologies can be used to improve the provision of the professional care to the elderly people in their resident context.

III. SERVICE SPECIFICATION

According to the literature and characteristics of the target groups, the following services had been considered relevant for the Portuguese ACE [2]:

- Remote Delivery of Expertise;
- Learning and Training;
- Interpersonal Communication;
- Collaborative Decision Making.

The Remote Delivery of Expertise aims to provide cooperation between experts (Team Work and Collaboration) in what concerns for instance the discussion of the better teaching methods for children with special needs in order to improve the assessment process, for a small group with a particular disability. Another objective of this service is to guide the teacher or the family (Supervision and Follow Up) to deal with difficulties concerning the intervention with the individual, contributing, thus, for the correct prosecution of the educational program.

The Learning and Training services can include sessions on a one-to-one basis, either providing direct information or transmitting stored information. Those options depend upon the nature of the training, the level of privacy and the homogeneity of the users regarding their needs. Different programs can be provided.

Mental impaired people usually have poor speech associated to language disorders. A large number of them depend on communication aids with non-orthographic written systems. The use of graphics in telecommunications (Interpersonnal Communication) may improve the social integration allowing them to communicate at distance to other persons, developing their communication abilities.

The Collaborative Decision Making service includes Counselling service, aiming advising family or careers, and Information service aiming helping the users to overcome problems. The time of this scheduled service is settled beforehand, concerning the period during which general guidance is available upon request.

IV. SYSTEM OVERVIEW

In order to accomplish the aforementioned services, modular terminals, based in personal computers with video codecs, have been developed. Different workplaces can be linked by point-to-point ISDN connections.

When two workplaces are connected, the all system works under a master slave relation: the control facilities (connection management, image and communication commands) are provided by the service centre terminal (service provider). A particular service centre terminal supervises the service centre and also provides the functions needed by the servicers to operate the local system and control the remote operation of the client equipment. The client terminal runs the local activities in the client Institution, or, eventually, in the client home.

Both the service centre terminal and the user terminal include telecommunication and control facilities, and easy-to-use multimedia user interfaces with text, graphics and images, and are constructed of a number of different modules, which are environment dependent, and may be mandatory or optional. The modular design is necessary either to adapt the terminals to specific users and to enable the re-use of the modules in different services and by various groups. Optional modules may be added inside the terminals (e.g., software packages and expansion cards) or may be connected to their output ports.

A. System Hardware

The basic configuration of each terminal includes [3] (fig. 1) a personal computer, a codec (a Tandberg Vision Model 15), an ISDN network controller (PCbit), a touch screen, a frame grabber (Video Blaster), a video camera, a document camera, a video switch, a hand-free telephone unit, and a lightning unit. Optionally, additional video cameras and/or document cameras may be included.

Each terminal is able to use two basic ISDN accesses. One of the basic accesses is reserved for the continuous real-time image transmission (codec operation), and the other one is used either for computer to computer (one B channel) and voice communication (the remainder B channel). This configuration is the most efficient, since three B channels are reserved for videophony and one for data communication, but it is also an expensive solution because it requires two basic accesses.

B. System Software

The system software manages the various local applications and, through telecommunication facilities, dialogues with remote terminals. The system software also manages, through interfaces, the hardware and software input and output devices, in order to provide the ability of special configuration mechanisms to recognise relevant user needs (e.g., sound and brightness control).

For the user interface, both the service centre and client terminal user interface, a consistent metaphor for user interaction has been established: the Virtual Resources

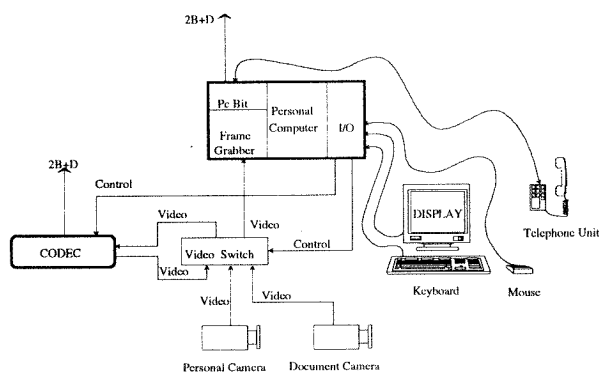


Fig. 1 - System hardware

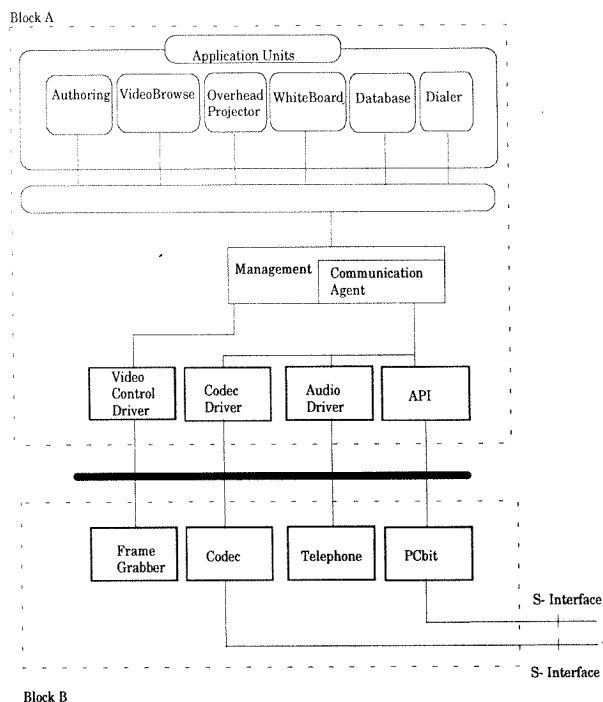


Fig. 2 - Software Architecture

Centre. This metaphor is based on the notion of a three dimensional room with different pieces of furniture and equipment, each one representing a different application (tool) with functions to establish connections over ISDN and with facilities for preparing, presenting, accessing and discussing information.

Shown in figure 2 is the Software Architecture, which has been designed with the objective of keeping the management functions isolated from the applications, the network and the hardware support.

The modules within the block A are the software units located on the personal computer, while the modules within the block B reside in the respective add-on boards or are stand-alone modules (video codec and hand-free telephone unit). The upper part of figure contains the environment dependent modules (Applications Units) with adequate user interfaces. The Application Units operate to control the different media in order to provide the different services types, upon the user requests.

The module Management provides control functions, and comprises the Communication Agent. The remainder units of the block A are software drivers for the dialogue with the special purpose hardware.

C. Communication Agent

The user can issue an invocation primitive to any of the available Application Units that will establish a permanent dialogue with the Management. A connection request primitive implies the participation of the Communication Agent that will issue an invocation request to remote site to ask its counterpart to invoke relevant process.

The Communication Agent controls the set-up, the dynamic reconfiguration and the progress of the communication, with inputs from the Application Units.

On the other hand, the Communication Agent, as a daemon running on each station, is also able to accept requests for communication establishment.

A flexible and robust module is necessary to make the application easy to use and much more acceptable. Furthermore, the Communication Agent must provide an environment that can support different types of participants. Hence, it supports asymmetrical communications in which the participation of users differs with respect to different media types, and control resources that are either local or remote.

Due the above mentioned reasons, communication management functions can be divided into two categories: establishment, and control of the communication. For the communication establishment, we have developed a procedure that consists of three phases: invitation, negotiation and set-up. In the invitation phase, the initiator of the communication sends out requests to the potential participant and accumulates their responses. The negotiation phase is important in accommodating heterogeneity among the communication participants. In this phase, the participating entities exchange information about their capabilities, their requested mode of participation. The connections between participants are set-up based on these exchanges.

When a video transmission is required, the set-up is finished after the establishment of a second ISDN connection reserved exclusively for the inter-codec communication. Contrary to the first connection, that is established through the ISDN controller board, this second connection is established through the codec.

After the set-up phase the Communication Agent has the responsibility to monitor the state of both connections and it is also involved during the disconnection phase.

V. APPLICATION UNITS

The Application Units provide means either for documents preparation and distance communication between the service provider and the client, which should comprise the integration of voice, video, screen-based communication tools, and presentation facilities.

The following Application Units have been developed for the Microsoft Windows environment: Authoring, Dialer, OverheadProjector, WhiteBoard, VideoBrowse and DataBase.

A. Authoring

The documents in the ACE environment might contain:

- A lesson that can be downloaded on the client terminal during the remote Learning and Training services.

- A general purpose material that can be used in Remote Delivery of Expertise and Collaborative Decision Making services.
- Pictograms suitable for the Interpersonal Communication service.

Documents can be loaded, prepared and annotated by means of the Authoring Application Unit. Authoring presents a balanced solution providing tools for browsing the information contained in a specific document, together with other tools to create and change multimedia compositions. These compositions can be entirely created by the user, supported by other applications, and improved from raw material from a library. The added information can be any type of media, such as text, graphics, images and sounds, and there are some composition mechanisms that allow groups of objects to be represented as unique entities and functions to create virtual structures. In order to be possible the communication between different applications a client-server scheme was used. The Authoring application is the client and it is able to import information from different servers. Since this server-client scheme has been implemented with Object Linking and Embedding (OLE) technology, Authoring acts as a client that is able to receive data from commercial OLE servers.

B. Dialer

The Dialer presents facilities to invoke a communication connection and to retrieve, in an association with a database, the destination number.

C. OverheadProjector

The OverheadProjector is an application without a balanced amount of browsing and authoring support, since it is polarised for the navigation through the document (browsing), which allows the selection of a pre-prepared lesson and the navigation through its different pages.

The information contained in the OverheadProjector window is private information, which means that the service provider is the only one who has access to it. When the service provider wants this information to be common to both terminals, he must "project" the related page in a public screen (WhiteBoard).

D. WhiteBoard

The "projection" of a specific page in the WhiteBoard corresponds to the download of information from the terminal of the service centre to the client terminal. After the download of the information, both the service centre and client terminal have the WhiteBoard windows with the same common information.

During such presentation the participants, both the service provider and the client, are allowed to perform the following actions on the screen combined with spoken

comments: draw with a virtual pencil, write with a virtual text, erase with a virtual erase and move with a virtual hand. Furthermore, information from a document camera can be "projected" in the WhiteBoard.

The use of these functions is permitted to both participants, but is always the service provider who has the control over them. Because of that, the WhiteBoard of the service provider terminal presents a Tool Box with the referred functions, which does not appear in the client terminal. This means that when the service provider requires a specific action of the client, such as drawing with a virtual pencil, he must choose the drawing icon of the Tool Box.

E. VideoBrowse

The VideoBrowse is a window in the upper left corner of the screen showing the communication partner or the self-view of the user and that presents several commands (Connect/Disconnect, Zoom/In, Freeze/Unfreeze, Local and Remote):

- The Connect button activates the Dialer, for the connection establishment. After a connection establishment, the name of the button is changed for Disconnect, which finishes the connection;
- The Zoom In/Zoom Out and Freeze/Unfreeze are also toggled buttons and they are, obviously, related with the zoom and freeze commands;
- The Local and Remote commands allow, respectively the selection of the cameras both of the remote and local sites, and enables the self-view.

F. DataBase

In intervention with disabled people it is a common procedure to collect information for the characterisation of the person itself and of its problem in order to get a better overview of the situation. A database that can contain all the information that can affect performance or illustrated special handicap situation, for instance on the use of the system, can be of a great help to understand common problems and find common solutions between different types of impairments. With that objective we have developed a database in order to allow afterwards data treatment permitting to get a better description of the population and a better understating of the problem.

Furthermore, additional information may be retrieved from the normal ACE operation, related to service acceptance, call patterns, call duration, destinations, service usage, user's difficulties and interviews. A software tool has been specially prepared to collect, process and analyse such data, which is invaluable in facilitating the evaluation of the services, in order to ensure that they are user-driven, rather than technology-driven services.

VI. DISABLED USER TERMINAL

The software architecture already described is the same for the service centre terminal and client terminal. However, concerning the client terminal special efforts have been made to achieve a user-friendly man-machine interface.

When a demanding call from a remote terminal appears, the client is signalled by an audio signal and symbolic picture indicating that a transmission is being established. If the client accepts the call the VideoBrowse and WhiteBoard applications will be activated after the set-up procedures. These applications have the same graphical aspect of the related applications of the service centre terminal, but they do not present control functions.

The Dialer application already mentioned during the description of the Applications Units is, in this case replaced by a special Dialer: when the application is started up, photos of the persons, that the user might want to call, appear on the screen. By pointing one of the photos a call is initiated. Special symbols are presented for signalling if the connection call has or has not been successful.

In addition to a face-to-face interaction, allowed by the videophony, and the possibility of sharing information, it is interesting to have skills for pictogram transmission. For this reason an additional application has been considered, SymbolPad, that includes a user-interface for pictogram's acquisition, transmission and reception. The client must select, from an available collection, the pictogram he or she wants to transmit. After this selection the client is able to put the pictogram on an adapted document camera [3]. The pictogram will be transmitted to the remote place and it is presented on the screen of both terminals.

VII. SITE NETWORK DEFINITION

Considering the dissemination of the ISDN in Lisbon area, the services viability and availability, the target groups, and the available terminals, several social care Institutions were selected for the Portuguese ACE. The first phase of the experiment was conducted in the first semester of 1994 with the participation of FMH (the service centre), Cercizimbra (mental impaired), Cerci Mira-Sintra (mental and visual impaired) and Cerci Lisboa (mental and visual impaired). During the second phase (second semester of 1994, additional service centres will be considered and two more Institutions will join the experiment: Centro Feliciano de Castilho (visual impaired) and Centro de Dia de Telheiras (elderly).

VIII. CONCLUSIONS

We have referred a experience concerning the development of remote care services for disabled people, in which ISDN is used as communication infra-structure.

The paper presents aspects related with service requirements and focuses the technical description of the terminal equipment that has been developed. During the field trials that are being conducted, most of the users had demonstrated satisfaction and present good performance in the use of the interface facilities.

Considering the staff opinion they considered easy to use the equipment and when asked about the viability of the services they referred that the disabled users could benefit very much from these types of services. Furthermore, they considered essential the use of the videophony, otherwise they could not see the users that could cause a lack of "human contact".

ACKNOWLEDGEMENT

The work presented in this paper is a consequence of the good collaboration between the Portuguese partners of the TeleCommunity project: the Telemática group of INESC

Aveiro, the FMH, and the TLP. However, the field trials in real-life scenarios were only possible due the invaluable support of various Portuguese social care Institutions. The authors would like to express their acknowledgement to Santa Casa da Misericórdia de Lisboa, Direcção Regional de Educação de Lisboa, Fenacerci, Cercizimbra, Cerci Mira-Sintra, Cerci Lisboa, Centro Feliciano de Castilho and Centro de Dia de Telheiras.

REFERENCES

- [1] "TeleCommunity Technical Annex", November 1991;
- [2] Leonor Moniz Pereira, Paula Lebre, João Purificação, "Definition of users and servicers requirements", TeleCommunity Deliverable, March 1992;
- [3] "Description of Equipment its Operation and Facilities", TeleCommunity Deliverable, RACE 2033, March 1993.

Ethernet Traffic Measurements: A Case Study

José Luís Oliveira, Joaquim Arnaldo Martins, Joaquim Sousa Pinto

Resumo- As redes locais de comunicação tem vindo a ser instaladas a um ritmo exponencial. Esta expansão, aliada ao crescimento de cada LAN, pela introdução de novos e mais exigentes sistemas, implica um aumento no tráfego de rede. O estudo destas variações é um dado importante, quer para a gestão, quer para o planeamento futuro da rede. Neste artigo são apresentadas algumas ferramentas que permitem estudar o tráfego numa rede Ethernet e algumas medidas de tráfego nomeadamente, tráfegos diários e semanais, impacto de novos sistemas e distribuições de tempos entre chegadas dos pacotes.

Abstract- Local Area Networks have been spread over the world during the last decade with an increasing installation rate. The expansion of each LAN, caused the introduction of new systems, usually implies more network traffic. It's important to the network manager to know the characteristics of this traffic in order to study, by simulation, the expected grow on data interchange and to plan future redesign of the network segments. This paper presents a set of tools to make network traffic analysis and to predict some resources requirements, over an Ethernet network. It presents as well some traffic measures, namely, daily utilization rate, impact of new hosts on network traffic and a study of the inter-packet time distribution.

I. INTRODUÇÃO

The increasing number of installed Local Area Network puts new demands to the network manager when dealing with the control and the planning of the network configuration. Besides the crescent quantity of attached systems, there are a growing set of distributed services which have an important impact on the network traffic. Systems like UNIX, MSDOS PC, MAC, terminal servers, print servers, file servers, diskless workstations, remote printers and services such as NFS applications, E-mail, XWindows and distributed multimedia may be found in a nowadays network. All this panoply of systems and services have embedded different communication protocol's stacks (TCP/IP, OSI, AppleTalk, Netware, ...). The management of those types of networks has been mater of study and development during the past years either by leader vendors either by governmental standardization comities.

The ISO organization proposes a management scenario appropriate to the OSI layered architecture. In spite of

being claimed more powerfully than other proposals, the standardization process is not complete which delays the development of practical solutions. The management information model is based on managed objects, abstract entities that represents data and communications resources. An object oriented design allows a single system to have several managed objects. These managed objects are able to report and notify information to another object acting as a manager, through a management protocol - the CMIP (Common Management Information Protocol).

The Internet community has proposed a set of RFCs (Request For Comments) that have been adopted by a crescent group of developers and users. The management information organization is based on a standard information model - the MIB (Management Information Base). The management protocol, SNMP - Simple Network Management Protocol, has already been included in the operating system of several hardware platforms. A second generation of the SNMP, the SNMPv2, is already defined and it incorporates some functionality's that allows some proximity with the OSI management protocol.

In both management architectures, Internet and ISO, the approach has been "start from the top", i.e., specify the models of protocols and system management architecture, namely the composition of the management information that is specific to each system.

Although the definition of these models is decisive to the beginning of development work, the network management task must "start from the bottom", i.e., it must perform some low level tasks in order to maintain a clean end-to-end connectivity. Examples are: monitor the communication medium, detect link anomalies and analyze the intensity of traffic.

After initial developments around the management models and on the systems management, the normalization groups begun some work that looks at first to the management of the network media. Examples of that work are the RMON MIB[1], the TR-RMON MIB[2], the BRIDGE MIB[3] and a few other ones that concentrate on the network and on the communication equipment monitoring [4]. The network traffic analyzes is crucial to plan the network short-term necessities and to evaluate the impact of new added equipment's.

While the Ethernet monitoring wasn't normalized we have developed some add-hoc tools, around a MSDOS PC environment, that help the characterization of the network

activity [5,6]. This work was adapted to a RMON MIB agent which portable graphical user interface allows to construct a distributed monitoring scenario. In this paper we will describe briefly these developments, an analyzer for packet interarrival time and some network measures coming from these tools.

II. MONITORING TOOLS

The first step towards an integrated monitoring architecture was the development of a Traffic Generator and a Traffic Analyzer, in order to evaluate how the network supports several load conditions and in order to characterize the network activity. This leads to a better understanding of the network behavior helping the manager to predict some areas of potential faults. The Traffic Generator allows to test the network or a specific machine behavior under different loads. The Traffic Analyzer reads information about all circulating packets like packet length, number of errors, number of broadcasts, protocol types, and the observed hosts' physical addresses. Another important task of the Analyzer is the capability to measure the packets' interarrival times. Since these tools are based on a common commercial PC, the number of simultaneous operations are limited to a few depending on the required processing power (Figure 1).

The Analyzer was adapted to an RMON Agent and consequently some others important capabilities have been integrated (like the Alarm and Event Report information) [7,8]. Besides the RMON compliance, the system still performs as an independent monitor since there are some monitoring tasks that aren't part of RMON MIB like the packet generation and the interarrival time measures (Figure 2).

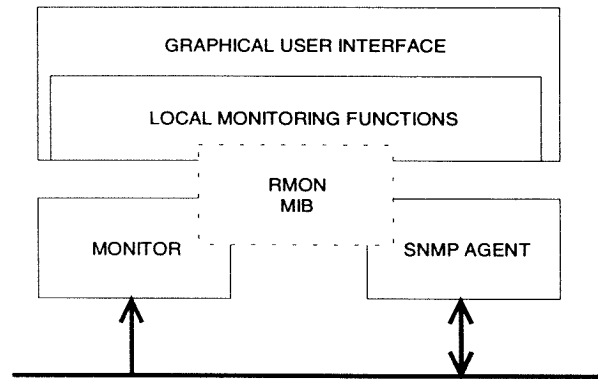


Figure 2 - The RMON Agent architecture

III. MEASURES IN AN ETHERNET SEGMENT

The University network is mainly composed by Ethernet LANs and there are several Class C TCP/IP networks (usually one for each department) that are interconnected through transparent bridges to the campus backbone. Due to the bridge forwarding method, the monitoring action performed by a RMON agent only applies in one side of the bridge considering this side of the topology as a single logical unbroken bus. The following considerations and measures were made in one of those network (inside the Electronics and Telecommunications Department). Besides the TCP/IP traffic there are also traffic coming from Novell and AppleTalk connections which allows to extend the total number of interconnected systems above the IP Class C capability. Nevertheless, the current quantity is near to 170 nodes distributed as follows: 4 Netware servers (with a total of 260 licenses), 30 UNIX workstations, 4 diskless workstations, 80 PC running Netware, TCP/IP or even a XWindows server, 40 Mac

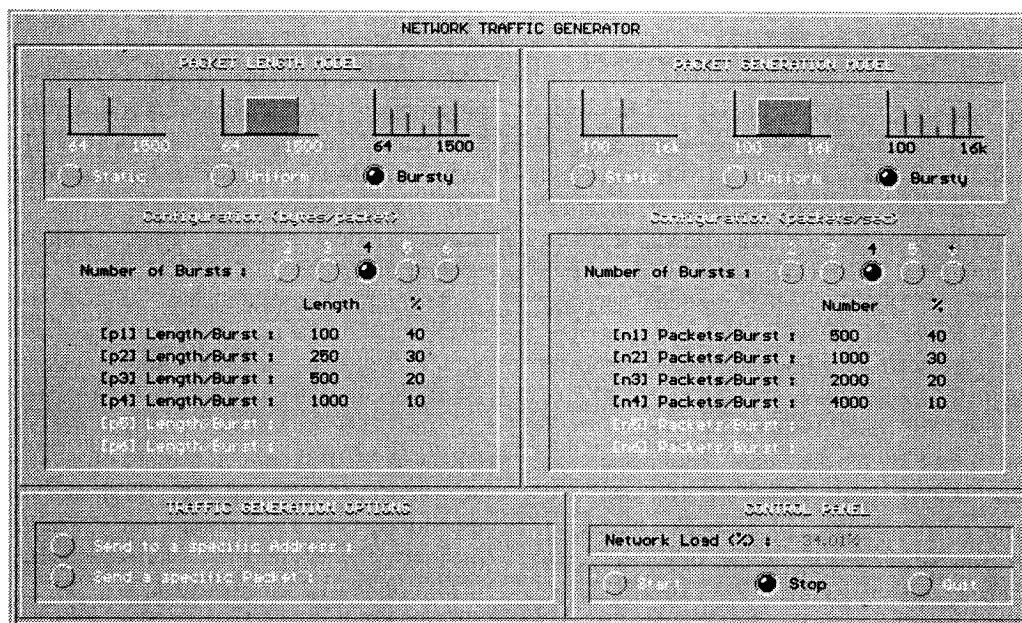


Figure 1 - The Traffic Generator user interface.

which use mainly Netware and 5 terminal servers. A reference must be made to some consuming services like NFS, to a VLSI classroom where 10 workstations and one server are running a centralized CAD application (at some discrete week intervals) and PC based classrooms which use the Netware infrastructure.

A. Traffic Measures

The measure of utilization rate on continuous mode allows to know how the network is used, what are the higher traffic intensity periods and their correspondent values. This information can be obtained either by the Traffic Analyser, either remotely from the RMON agent consulting its *Statistics* group objects.

The Figure 3, shows a weekday sampling, where it is emphasized the workday traffic if compared with the night lower traffic. The sampling was made over consecutive 30 minutes periods where can be detected 3 intervals with mean traffic above the 20%. The total mean was 6.13%, considering the overall period, and 9.12% if considering only the period among the 9 to 21 hours.

On Figure 4 it is plotted one selected workday measure over a 24 hours sampling and also the mean day computed along the week. We can see, for the mean day curve, that the afternoon is the more used network period with the

peak load hour reached at 15 to 16:00, with a mean value of 13.91%.

B. Packets' Arrival Time

Measuring the packets' arrival time gives us two type of observations. The first, that we call *Packets' Interarrival Time*, consists in the measure of the time between two consecutive packet preambles. The other, the *Packets' InterGap Time*, measure the time space that separates two consecutive packets, since the end of the first one to the beginning of the second one (Figure 5).

The major problem on the measure of the packet generation time, with a software package, is the constraint for real-time operations. Considering the Ethernet transmission rate (10 Mbps) it is needed a time resolution of $0.1 \mu s$ that is a higher rate than a normal PC can handle. The Traffic Analyzer system can achieve a $20 \mu s$ resolution which is made by directly program the PC timer to generate 50,000 interrupts per second to construct a clock counter. This resolution represents the maximal absolute error for the packet time generation measurements.

However the statistic treatment that is associated with the resulting measures is typically based on histogram graphics which can attenuate partially this drift.

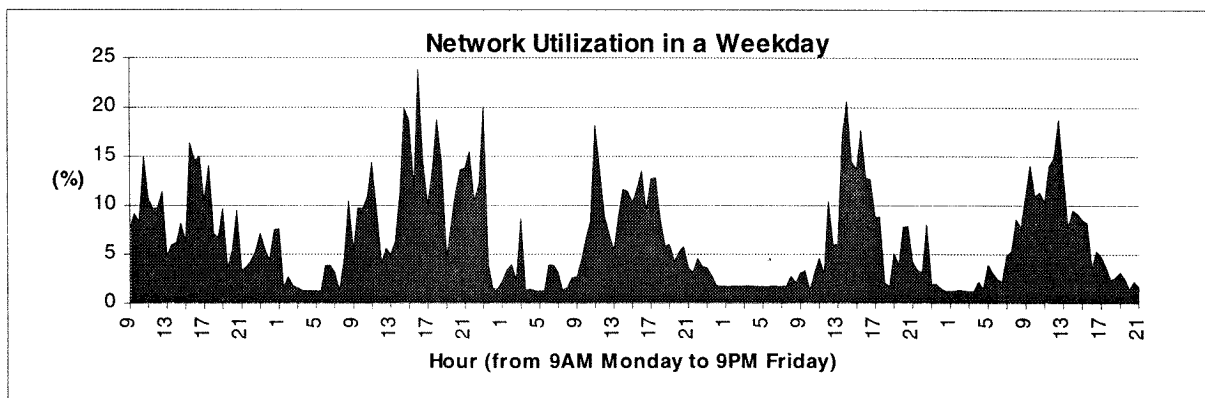


Figure 3- A Traffic sampling over a week period

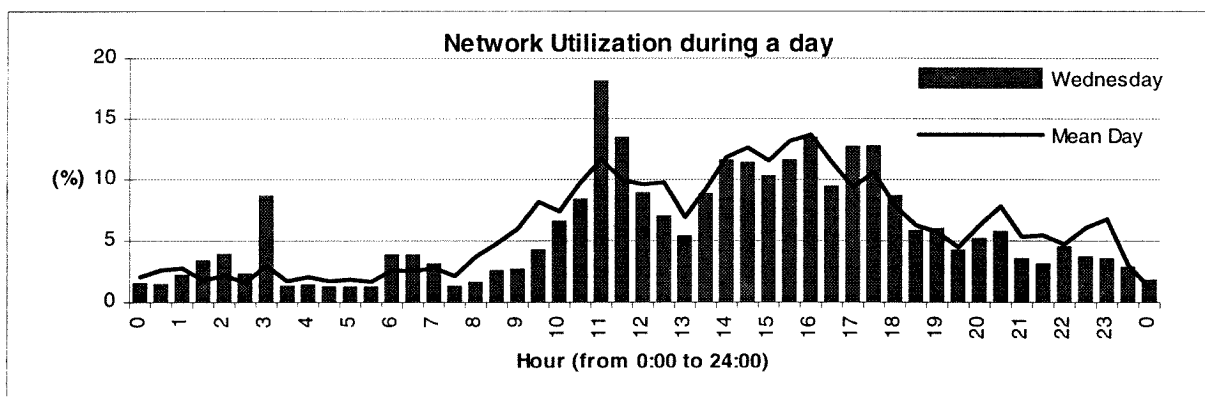


Figure 4 - A Traffic sampling over a 24 hour period

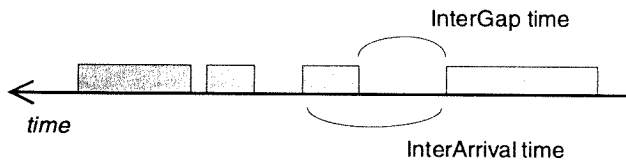


Figure 5 - Packets' InterArrival and InterGap Times

Nevertheless, the empirical statistic mean computed from these values tend to be a bit increased, due to the error associated with the measurement process.

Performing some measures during half a hour intervals we have arrived to the following results. The Interarrival time probability density function follows an unexpected curve that can be explained by the medium access method. When a packet transmission begins the medium remains unavailable until the end of the packet has been reached. After this, the network is free again for another transmission. So the delay between two consecutive headers depends on, and it is correlated to, the packet length. The overhead introduced by the packet transmission time can vary from a few μ s, resulting from small packets (collisions) to 1220 μ s the time needed for a 1514 byte length packet. Since the packet length isn't uniformly distributed the interference on the Interarrival time distribution is affected with some peaks that are caused by the peaks of the packet length distribution (major peaks normally at 64, 1078 and 1500 bytes that implies delays respectively of 51, 862 and 1200 μ s).

The Figure 6 shows a comparison between the measurement of interarrival time and intergap time empirical distribution. The main discrepancies, besides the spreading, is centered around 800 and 1200 μ s.

Measuring the InterGap Times seems a good way to remove the packet length interference. As we will see some new particularities were found on this later method. The previous samplings were repeated using the same 30 minutes' interval but applying a Gap measurement. The Figure 7 represents the Probability Density Function (PDF) for these results. In spite of the new curve don't includes the old spikes the result is not so famous as it will be expected (a curve similar to the exponential probability density function). This drive us to other conclusions which are related with the way each service transfer information over the network. As the synchronous nature of a computer systems, the transmission mechanism also follows regular time intervals. For instance a file transfer operation is split over several packets which are send to the network in regular intervals, depending on the interface performance, and only affected by some other packets traveling between others systems and that momentarily monopolize the network media. It can be a bit ambitious to approximate the packet generation probability density function to an exponential one, since the packets, as we have discuss, are, most of the time, correlated. The no-correlation condition is fundamental when consequent events are supposed to follow this type of distribution [9].

On the graphic, it was plotted also an exponential curve (*Exp*) with the same mean as the empirical results one. The differences are caused by an higher density of the empirical PDF at higher time intervals. The curve *Exp2* shows a second exponential but for a mean that have an half value that the previous one. If the values above 3000 μ s are removed, the curve fits better on the experimental PDF.

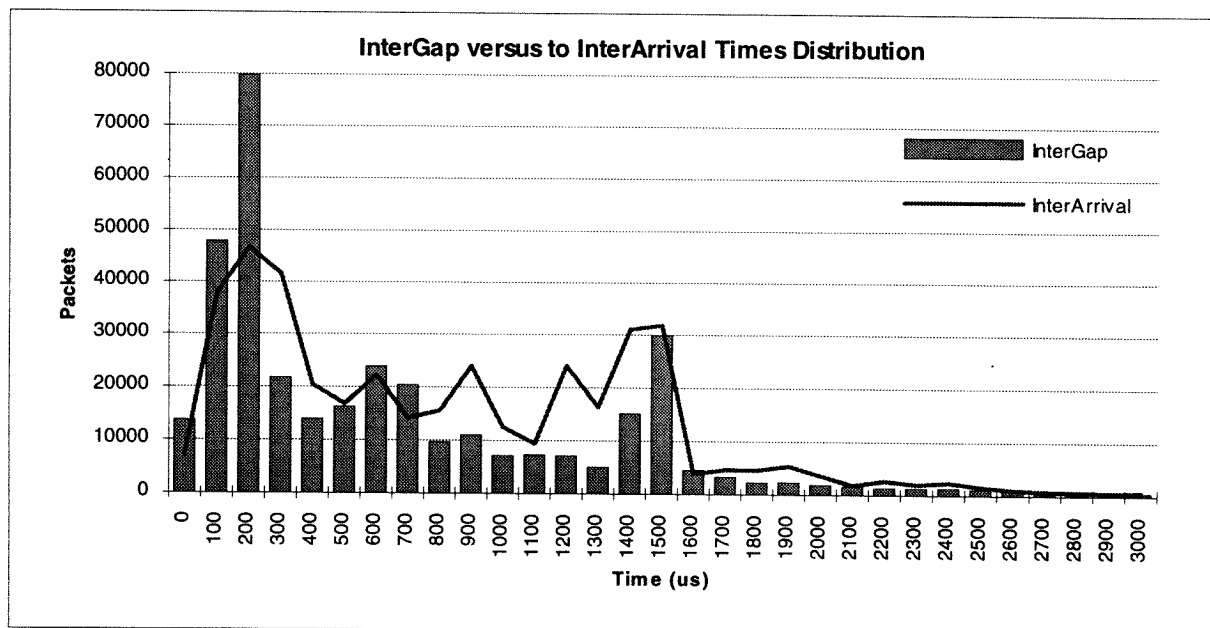


Figure 6 - Comparison between Empiricals PDF for packets interarrival and intergap times.

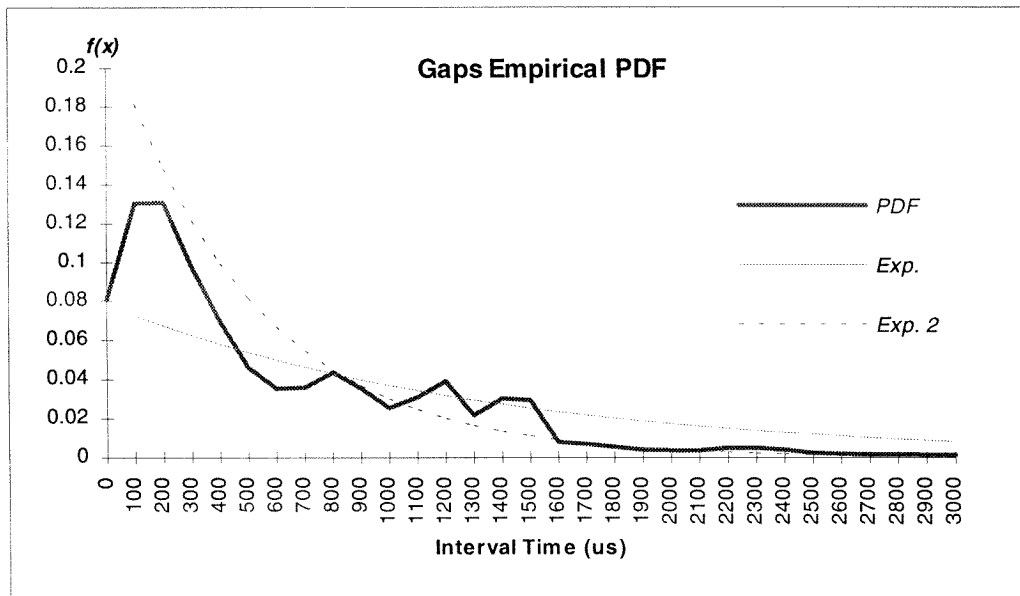


Figure 7 - Empirical Probability Density Function for packets' intergap times.

C. Hosts Study

The host addresses analyzer, study the number of hosts by each time interval and relates it with the measured traffic load. Other informations retrieved are the relations between packets and network load and also the influence of the network load on the errors occurrence. On Figures 8 and 9 we can see a plot of these measures, made using 10 seconds sampling periods. While the packet number follows a curve near to linear, the error number increases quickly for values of load above the 40% showing a exponential increasing rate.

The Figure 10 represents a hosts per traffic curve. It shows that for mean values, of active hosts, upper to 80, the network load is above the 50% of utilization rate.

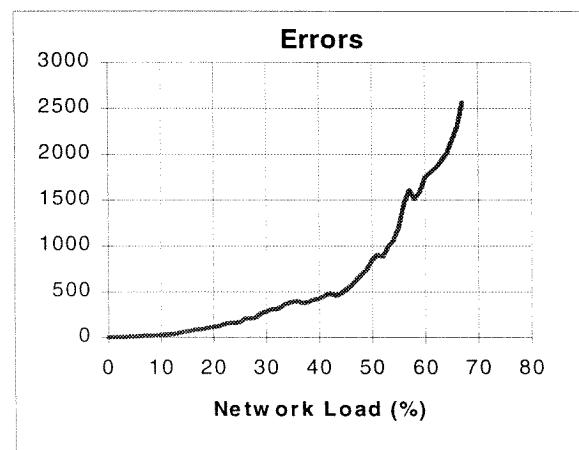


Figure 9 - Number of Errors by Network Load

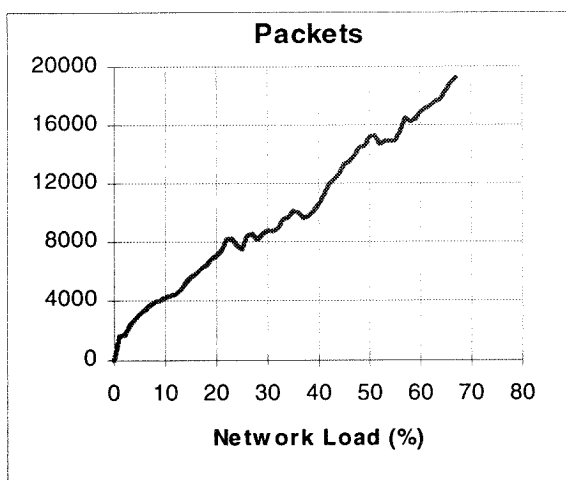


Figure 8 - Number of Packets by Network Load

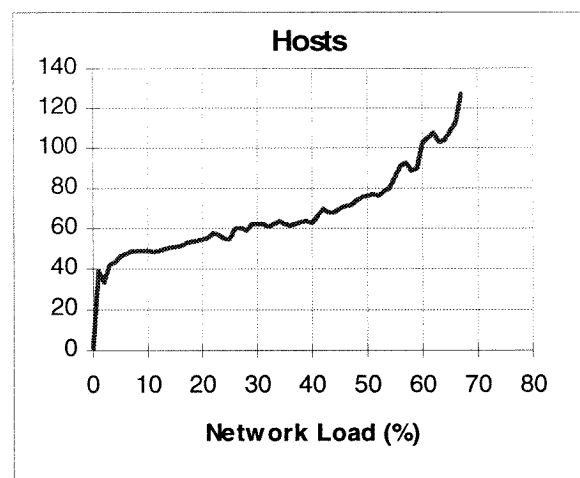


Figure 10 - Number of Hosts by Network Load

IV. CONCLUSIONS

Network Management must take as a priority the network monitoring to achieve a clean of errors media. Some standards have already been delivered on this field enabling the development of compactible products. We have present some of those tools, namely a Network Analyser that works either as a autonomous package either as a RMON agent. Some functions of this Analyser were explored like the measurement of the weekday traffic, interarrival and intergap times and hosts due to network load.

V. REFERENCES

- [1] S. Waldbusser, "Remote Network Monitoring Management Information Base", *Internet Request for Comments 1271*, Carnegie Mellon University, Nov. 1991.
- [2] S. Waldbusser, "Token Ring Extensions to the Remote Network Monitoring MIB", *Internet Request for Comments 1513*, Carnegie Mellon University, Sep. 1993.
- [3] E. Decker, K. McCloghrie, P. Langille and A. Rijsinghani, "Definitions of Managed Objects for Source Routing Bridges", IAB, Sep. 1993.
- [4] F. Baker, D. Lostick and K. Tesink "Work Group Synopses", *The Simple Times*, November/December, 1993.
- [5] J. Sousa Pinto, José L. Oliveira and J. Arnaldo Martins, "PC Workstation for Ethernet Monitoring and Management", *SBT/IEEE International Telecommunications Symposium*, Brazil, Sep. 1990.
- [6] J. Arnaldo Martins, J. Sousa Pinto and José L. Oliveira, "Measuring, characterizing and generating LAN traffic", *IEEE ICCS'90 - International Conference on Communications Systems*, Singapore, Nov. 1990.
- [7] Fernando Pipa, José L. Oliveira and J. Arnaldo Martins, "Desenvolvimento de um Agente de Monitoração para Redes TCP/IP", *Revista do DETUA*, Vol. 1, Nº 1, Janeiro 1994.
- [8] Pedro Costa, Maurício Domingues, J. Arnaldo Martins and José L. Oliveira, "Monitorização e Visualização de Recursos num Ambiente Distribuído", *Revista do DETUA*, Vol. 1, Nº 1, Janeiro 1994.
- [9] Averill Law and David Kelton, *Simulation Modeling and Analysis*, McGraw-Hill, 1982.

Linearização de Amplificadores de Potência por Realimentação Activa

José Carlos Pedro*, Jorge Perez⁺ e José Rocha Pereira*

* DETUA

⁺ E.T.S.I.T. U.P. Madrid

Resumo- O trabalho que neste artigo se descreve, enquadra-se no plano de estudos de doutoramento do seu autor [1]. É composto por quatro blocos fundamentais que visam, em primeiro lugar a motivação do trabalho, uma abordagem teórica do projecto de um linearizador por realimentação não-linear, estudo de modelos não-lineares de GaAs MESFETs, e, finalmente, a validação experimental, em tecnologia monolítica de microondas - MMIC -, das hipóteses formuladas.

Abstract- This paper describes the work developed under the Ph.D. course of its author [1]. It is divided in four main parts which refer to the motivation, theoretical background of amplifier linearization circuits design, GaAs MESFET modeling, and finally, experimental validation. That was achieved by the design and test of a MMIC power amplifier, linearized by means of active feedback.

I. INTRODUÇÃO

Uma das características fundamentais próprias aos sistemas não-lineares é a apresentação de respostas com componentes espectrais não existentes nos estímulos a que são sujeitos. Num sistema de telecomunicações de banda estreita típico, estas componentes são geralmente divididas em duas partes: componentes que aparecem dentro da banda do sinal, e fora dela. Esta divisão tem a sua justificação no facto de que, enquanto o primeiro grupo se pode eliminar facilmente por filtragem linear; para o segundo este procedimento é simplesmente inaplicável, já que estas componentes de distorção são, na frequência, indistintas do sinal útil. Resta pois, como única alternativa à degradação da relação sinal-ruído, evitar que esta forma de distorção não-linear, ou **distorção de intermodulação** - IMD -, como é vulgarmente apelidada, seja sequer gerada. Para realizar este objectivo, reduzem-se as suas origens, as não-linearidades próprias ao sistema, por adição de circuitos auxiliares, que, devido à sua função, se chamam **linearizadores**.

Em todos os sistemas de linearização de amplificadores de potência em microondas, anteriores a este trabalho, seguiu-se um procedimento que se pode apelar de linearização *a posteriori*. Com efeito, em primeiro lugar projecta-se e caracteriza-se o amplificador a linearizar,

como se se tratasse de um amplificador livre, e só depois se constrói um linearizador que posteriormente lhe irá ser aplicado. Este processo conduz a sistemas volumosos, pesados, difíceis de ajustar, muitas vezes pouco eficientes, e, devido aos componentes não integráveis que usam, são caracterizados por comprimentos eléctricos equivalentes muito elevados, o que lhes reduz drasticamente a largura de banda em que são efectivos.

Um amplificador de RF ou microondas pode ser dividido em três blocos distintos: o dispositivo activo, e as malhas de adaptação de entrada e saída. É interessante verificar que, na maior parte das implementações, são as malhas de adaptação que em última análise determinam o comportamento em banda do amplificador. No entanto, só o dispositivo activo contribui para a distorção de intermodulação. Este conhecimento, associado à crescente necessidade de aumento da largura de banda efectiva dos sistemas de telecomunicações, induziu um novo conceito de linearização de IMD, em que o linearizador não é mais aplicado ao amplificador completo, mas directamente aos terminais do seu dispositivo activo [2].

O principal objectivo do presente trabalho é a apresentação deste novo método de "Linearização ao Nível do Dispositivo Activo", como meio de atingir uma solução integrada para o problema da linearização de amplificadores de Potência em RF e microondas.

II. LINEARIZAÇÃO POR REALIMENTAÇÃO ACTIVA

O estudo do linearizador por realimentação não-linear constitui o corpo teórico do trabalho. A opção por este tipo de linearizador deve-se a que é um dos menos explorados, e por se caracterizar por uma reduzida banda de frequências em que é efectivo. Este inconveniente abre uma importante área de teste à possível melhoria introduzida pela aplicação da nova técnica de linearização ao nível do dispositivo activo.

Na análise não-linear do sistema, representado pelo diagrama de blocos da Fig. 1, adoptou-se a Técnica das Séries de Volterra [3], por permitir representar cada um dos blocos por um conjunto de Funções de Transferência Não-Lineares, ou Operadores de Volterra.

Na topologia de realimentação mais vulgar em RF, a tensão-paralelo, a amostragem do sinal de saída é feita em

tensão, e a injeção à entrada, em corrente. Logo, no sistema da Fig. 1, X representa uma corrente no domínio da frequência, Y uma tensão, e M_i e M_o - as Malhas de Adaptação de Entrada e Saída - ganhos em corrente e tensão, respectivamente.

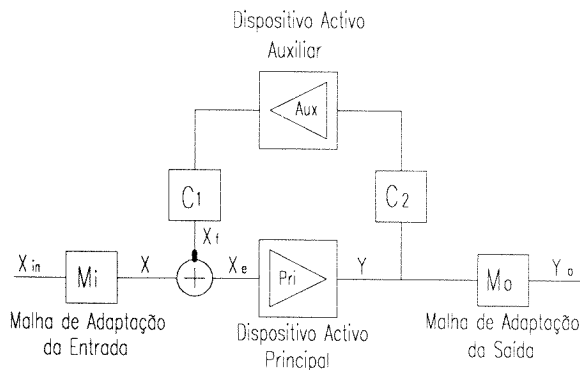


Fig. 1 - Diagrama funcional do sistema de linearização por realimentação activa.

Em concordância, o Amplificador Principal, Pri, deve ser representado em termos da sua transimpedância, e o Dispositivo Não-Linear Auxiliar, Aux, da sua transadmitância. C_1 representa um ganho de corrente e C_2 um ganho de tensão.

Para definir exactamente estes parâmetros admita-se que em malha aberta o conjunto gerador - M_i , bem como todo o circuito visto da saída de Pri, são substituídos pelos seus equivalentes de Norton e Thevenin, respectivamente. O primeiro incluirá uma fonte de corrente I_s , em paralelo com uma impedância Z_s , enquanto no segundo só existe a impedância Z_L . Supondo para variável independente a corrente, $X = I_s = X_{es}$, resultado da combinação de dois tons, e para variável de saída, ou resposta às frequências do sinal de excitação, a tensão $Y_s = V_{os}$ presente aos terminais de Z_L , elas admitem uma representação do tipo :

$$\begin{aligned} X &= M_i \cdot X_{in} = \\ &= \frac{1}{2} \cdot (A_1 \cdot e^{j\omega_1 t} + A_1^* \cdot e^{-j\omega_1 t}) + \frac{1}{2} \cdot (A_2 \cdot e^{j\omega_2 t} + A_2^* \cdot e^{-j\omega_2 t}) = \\ &= \frac{1}{2} \cdot (x_1 + x_1^*) + \frac{1}{2} \cdot (x_2 + x_2^*) \\ e \\ Y_s &= \frac{1}{2} \cdot (B_1 \cdot e^{j\omega_1 t} + B_1^* \cdot e^{-j\omega_1 t}) + \frac{1}{2} \cdot (B_2 \cdot e^{j\omega_2 t} + B_2^* \cdot e^{-j\omega_2 t}) = \\ &= \frac{1}{2} \cdot (Y_{s1} + Y_{s1}^*) + \frac{1}{2} \cdot (Y_{s2} + Y_{s2}^*) \end{aligned} \quad (1)$$

Nestas condições, a transimpedância de Pri pode ser definida por :

$$KG_s^{(1)} = HG^{(1)}(\omega) = \frac{Y_s}{X_{es}} = \frac{V_{os}}{I_s} \quad (2)$$

Assim, vem para ω_1 e ω_2 em malha aberta :

$$Y_s = HG^{(1)}(\omega_1) \cdot x_1 + HG^{(1)}(\omega_2) \cdot x_2 = Y_{s1} + Y_{s2} \quad \text{ou}$$

$$Y_s = KG_s^{(1)} \cdot X \quad (3)$$

para $2\omega_1$:

$$Y_h = HG^{(2)}(\omega_1, \omega_1) \cdot x_1^2 \quad \text{ou} \quad Y_h = KG_h^{(2)} \cdot x_1^2 \quad (4)$$

e, finalmente para $2\omega_1 - \omega_2$:

$$Y_i = HG^{(3)}(\omega_1, \omega_1, -\omega_2) \cdot x_1^2 \cdot x_2^* \quad \text{ou} \quad Y_i = KG_i^{(3)} \cdot x_1^2 \cdot x_2^* \quad (5)$$

em que $HG^{(n)}(\omega_1, \dots, \omega_n)$ e $KG_{s,h,i}^{(n)}$ são os operadores de Volterra de ordem n que produzem sinais a s : $\pm\omega_1$ e $\pm\omega_2$; h : $2\omega_1$ e i : $2\omega_1 - \omega_2$.

O sinal de entrada do dispositivo auxiliar, Aux, é dado por $C_2 Y$. Assim, o sinal de saída do ramo de realimentação, X_f , vem: $X_{fs} = C_{1s} \cdot [KH_s^{(1)} \cdot C_{2s} \cdot Y_s]$, (6) o que permite definir o ganho de transcondutância de Aux como :

$$KH_s^{(1)} = HH^{(1)}(\omega) = \frac{X_{fs}}{C_{1s} \cdot C_{2s} \cdot Y_s} = \frac{I_{fos}}{V_{fs}} \quad (7)$$

em que I_{fos} é a corrente de saída de Aux e V_{fs} a tensão presente na sua entrada.

Para os sinais de IMD a $2\omega_1 - \omega_2$ consideram-se três contribuições distintas. Uma, presente à saída de Pri e amplificada por $KH_i^{(1)}$ de Aux, outra, resultante da mistura de segunda ordem de sinais a $2\omega_1$ e $-\omega_2$, também provenientes de Pri, e, finalmente, a componente de IMD directamente gerada no dispositivo auxiliar por $KH_i^{(3)}$:

$$\begin{aligned} X_{fi} &= C_{1i} \cdot [KH_i^{(1)} \cdot C_{2i} \cdot Y_i] + \\ &+ C_{1i} \cdot [2 \cdot KH_i^{(2)} \cdot C_{2h} \cdot C_{2s}^* \cdot Y_h \cdot Y_{s2}^*] + \\ &+ C_{1i} \cdot [KH_i^{(3)} \cdot C_{2s} \cdot C_{2s}^2 \cdot Y_{s1}^2 \cdot Y_{s2}^*] \end{aligned} \quad (8)$$

Quando a malha é fechada, muitos outros produtos de IMD são gerados por combinações múltiplas entre os sinais nela presentes. No entanto, um grande número deles é de ordem superior à terceira, o que permite a sua exclusão de uma análise por Séries de Volterra com não-linearidades fracas.

Considerando $KG_{s,h,i}$ e $KH_{s,h,i}$ funções representativas dos operadores não-lineares de Pri e Aux, as expressões que aproximadamente descrevem o comportamento do sistema em malha fechada são:

$$Y_s = \frac{KG_s^{(1)}}{D_s} \cdot X = S_s^{(1)} \cdot X \quad (9)$$

$$\text{com } D_s = 1 - C_{1s} \cdot C_{2s} \cdot KG_s^{(1)} \cdot KH_s^{(1)}.$$

$$\begin{aligned} Y_i &= \frac{1}{D_i} \cdot [KG_i^{(3)} \cdot \frac{1}{D_s} \cdot \frac{1}{D_s} + \\ &+ C_{1i} \cdot C_{2s} \cdot C_{2s}^2 \cdot KG_i^{(1)} \cdot KH_i^{(3)} \cdot \frac{KG_s^{(1)}}{D_s} \cdot \frac{KG_s^{(1)}}{D_s}] \cdot x_1^2 \cdot x_2^* = \\ &= Si^{(3)} \cdot x_1^2 \cdot x_2^* \end{aligned} \quad (10)$$

Se os condicionadores de sinal, C_1 e C_2 , puderem ser arbitrariamente escolhidos, é possível, pelo menos teoricamente, eliminar por completo a distorção IMD do conjunto. Isto é conseguido para :

$$C_1 \cdot C_2 \cdot |C_2|^2 = - \frac{KGi^{(3)}}{KHi^{(3)}} \cdot \frac{1}{KGi^{(1)} \cdot KGs^{(1)} \cdot |KGs^{(1)}|^2} \quad (11)$$

Nesta situação, a distorção de 3ª ordem viria totalmente cancelada, restando uma IMD "residual" proveniente dos termos de ordem superior, de entre os quais, o de 5ª ordem será dominante. Assim, dado que a relação entre as contribuições de 3ª e 5ª ordem diminui com o quadrado da amplitude do sinal de entrada, deverá obter-se grande melhoria de C/I para os mais baixos níveis de sinal. Por esta razão designou-se esta situação de "Condição de Linearização em Baixo Nível".

Nos parágrafos acima, mostrou-se como com a Condição de Linearização em Baixo Nível, se podia cancelar completamente a distorção de 3ª ordem. Explicou-se também que seria de esperar o aparecimento de uma IMD "residual", devida aos termos de ordem superior, predominantemente o de 5ª ordem. Esta última contribuição tem uma dependência com o nível de sinal de entrada, que degrada a C/I em 4dB, por dB de aumento da amplitude da excitação. Esta característica, que em baixo nível é praticamente insignificante, pode constituir severa limitação nos níveis de sinal mais elevados. Com o objectivo de controlar este comportamento, propõe-se de seguida, uma extensão aproximada à análise do sistema.

Admita-se, por hipótese, que ao linearizar em baixo nível, não se pretende a condição de cancelamento total, permitindo manter um determinado nível de IMD de 3ª ordem, que se adicionará à residual de 5ª ordem. Em favor da simplicidade da argumentação, suponha-se ainda que a relação de fases entre estas duas componentes é tal que, ou estão em fase, ou em oposição. Observe-se agora o que se passa com o sinal de IMD de 3ª ordem, quando por exemplo o módulo de C_1 é aumentado desde zero, até um valor máximo bastante superior ao imposto pela condição de cancelamento total. O sinal de IMD de 3ª ordem começa por ser devido integralmente ao Pri livre, diminuindo depois a sua amplitude, à medida que se vai produzindo o cancelamento. Nessa altura, essa componente passa por zero, começando logo depois a subir novamente, sendo agora devida, principalmente, à gerada no dispositivo auxiliar, Aux. Neste processo a distorção de 3ª ordem tomou várias amplitudes e inverteu a sua fase. Assim, de acordo com os pressupostos acima estabelecidos, é natural esperar que numa das duas zonas (abaixo ou acima da condição de cancelamento total) exista interferência construtiva entre as componentes de 3ª e 5ª ordem, enquanto na outra zona, se verifique interferência destrutiva. Dado que a relação de amplitudes entre estas duas contribuições de distorção, varia com o quadrado da amplitude da excitação do sistema, a interferência destrutiva apresentará uma forte sensibilidade ao nível do sinal de entrada.

A extensão da análise que agora se expõe, tenta, de uma forma aproximada, sistematizar o aproveitamento deste efeito, como nova condição de compensação de IMD. Esta, por oposição à anteriormente descrita, e porque se destina a melhorar as características do linearizador na zona de amplitudes de sinal mais elevadas, foi designada por "Condição de Linearização em Alto Nível".

O uso da mesma argumentação utilizada quando da dedução de uma expressão aproximada da IMD de 3ª ordem, conduziu à conclusão de que, como primeira aproximação, se podem desprezar os termos de ordem inferior, na determinação do produto de 5ª ordem a $2\omega_1 - \omega_2$. No caso mais vulgar da excitação com dois tons de igual amplitude, $Yi^{(5)}$ pode ser representado unicamente pelos termos de ordem 5 devidos a Pri e Aux :

$$Yi^{(5)} = Si^{(5)} \cdot x_1^2 \cdot x_2^* \cdot |x|^2 \quad (12)$$

onde

$$Si^{(5)} \approx \frac{1}{Di} \cdot \left[KGi^{(5)} \cdot \frac{1}{Ds} \cdot \left| \frac{1}{Ds} \right|^4 + \right. \\ \left. + KHi^{(5)} \cdot C_{1i} \cdot C_{2s} \cdot |C_{2s}|^4 \cdot KGi^{(1)} \cdot \frac{KGs^{(1)}}{Ds} \cdot \left| \frac{KGs^{(1)}}{Ds} \right|^4 \right] \quad (13)$$

Assim, dentro deste novo contexto, tanto Pri e Aux como o sistema realimentado completo deverão ser descritos pela combinação de dois termos de 3ª e 5ª ordens, e será pela intersecção entre estas duas componentes que se poderão explicar os comportamentos que caracterizam a IMD em alto nível. Como se viu qualitativamente, pode existir um valor de $|X_H|$ para o qual se verifique :

$$Si^{(3)} + Si^{(5)} \cdot |X_H|^2 = 0 \quad (14)$$

que corresponde à Condição de Linearização em Alto Nível. Dado que Aux está, em princípio, polarizado numa zona mais não-linear e que, além disso, é sujeito a maior nível de sinal que Pri, será de esperar que (13) se possa aproximar pelo termo envolvendo $KHi^{(5)}$. Segundo esta aproximação, a amplitude da corrente equivalente de entrada $|X_H|$ a que corresponde a Condição de Linearização de Alto Nível, deve verificar :

$$|X_H|^2 = - \frac{Si^{(3)}}{Si^{(5)}} = \\ = - \frac{[KGi^{(3)} + C_{1i} \cdot C_{2s} \cdot |C_{2s}|^2 \cdot KGi^{(1)} \cdot KHi^{(3)} \cdot KGs^{(1)} \cdot KGs^{(1)}]^2}{[C_{1i} \cdot C_{2s} \cdot |C_{2s}|^4 \cdot KGi^{(1)} \cdot KHi^{(5)} \cdot KGs^{(1)} \cdot |KGs^{(1)}|^4]} \cdot |Ds|^2 \quad (15)$$

Para estudar o aproveitamento das condições de compensação de IMD em baixo e alto nível, no projecto de um sistema real, admita-se que este vem especificado segundo um critério de margem mínima de C/I, C/I_m , para potências de saída abaixo de um determinado valor P_o .

Suponha-se também, como descrito na Fig. 2, que o amplificador Pri isolado não cumpre as especificações para $P_o > P_{o1}$. (Neste exemplo supõe-se uma IMDo própria de Pri representável somente por uma componente de 3ª ordem).

A Fig. 2 mostra o comportamento das duas condições de compensação perante a mesma especificação. A condição de compensação em baixo nível, responsável pela curva C/I_B , mostra ser útil enquanto a potência de saída requerida for inferior a P_{o2} . A curva C/I_A correspondente a uma compensação em alto nível, apresenta-se como possível solução para especificações de potência abaixo de P_{o3} . Da comparação entre as curvas C/I_B e C/I_A , claramente se observa um compromisso entre ganho em margem de C/I , para os distintos níveis de sinal. De facto, o ganho em C/I oferecido pela compensação em alto nível, para potências de saída compreendidas entre P_{o2} e P_{o3} , é conseguido à custa de uma significativa perda nos mais baixos níveis de sinal, que, em certos casos, pode ser tal que a C/I obtida seja inferior à C/I_o do amplificador livre.

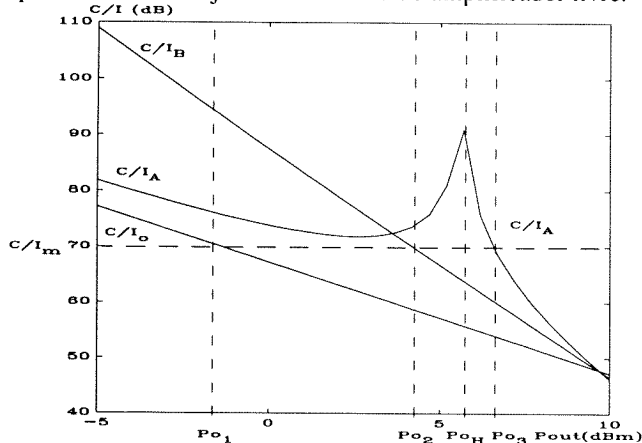


Fig. 2 - Condições de Linearização em Baixo e Alto Nível.

III. MODELAÇÃO DOS COMPONENTES ACTIVOS

O conceito de linearização ao nível do dispositivo activo em microondas, está claramente orientado à implementação em tecnologia monolítica, dado pretender-se aplicar o linearizador (electricamente) tão perto quanto possível do dispositivo a linearizar. No presente momento, os mais divulgados serviços de produção de circuitos monolíticos de microondas (e os que havia acessíveis na altura da realização do trabalho laboratorial), MMIC *custom foundry*, estão baseados no MESFET de GaAs, daí que todo o trabalho de validação experimental fosse dirigido exclusivamente a este dispositivo.

A análise não-linear de um circuito por Séries de Volterra exige que as não-linearidades sejam descritas por expansões em Série de Taylor em torno do ponto de repouso ou polarização. Dentro deste contexto, pode atribuir-se uma relação directa entre os coeficientes destas séries e o comportamento em sinal fraco (linear e não-linear) da não-linearidade. Dado que de qualquer

dispositivo, ou função matemática que o modele, é quase sempre possível obter uma destas séries, os seus coeficientes adquirem um grande valor na avaliação dos diferentes modelos empíricos, razão pela qual lhe é prestada tanta atenção.

Do estudo de um modelo físico de operação do dispositivo, pode concluir-se que as não-linearidades essenciais ao estudo de um amplificador em classe-A com MESFETs são a corrente dreno-fonte, $I_{ds}(V_{gs}, V_{ds})$, e a carga armazenada na zona de depleção, $Q_G(V_{gs}, V_{ds})$ [1]. É possível concluir também que nesta, a dependência com V_{ds} na zona de saturação pode ser desprezável, conduzindo a uma capacidade porta-fonte não-linear e dependente de V_{gs} , $C_{gs}(V_{gs})$, e a uma capacidade porta-dreno aproximadamente linear, i. e., independente das tensões de controlo. Assim, a modelação da parte não-linear do FET exige a determinação de duas expansões em série de Taylor, uma bidimensional, $I_{ds}(V_{gs}, V_{ds})$, e outra unidimensional, $C_{gs}(V_{gs})$.

Para I_{ds} ter-se-á :

$$I_{ds}(V_{gs}, V_{ds}) = I_{DS} + G_m.v_{gs} + G_{ds}.v_{ds} + G_{m2}.v_{gs}^2 + G_{md}.v_{gs}.v_{ds} + G_{d2}.v_{ds}^2 + G_{m3}.v_{gs}^3 + G_{m2d}.v_{gs}^2.v_{ds} + G_{md2}.v_{gs}.v_{ds}^2 + G_{d3}.v_{ds}^3 \quad (16)$$

em que todos os coeficientes correspondem ao cálculo das respectivas derivadas de n-ésima ordem no ponto de repouso. De igual forma, para $C_{gs}(v_{gs})$ tem-se :

$$C_{gs}(v_{gs}) = C_{gs} + C_{g2}.v_{gs} + C_{g3}.v_{gs}^2 \quad (17)$$

O significado físico dos coeficientes de (16) é importante quer para a avaliação das suas contribuições, quer para o futuro procedimento de extracção de parâmetros. G_m e G_{ds} são a transcondutância e condutância de saída do FET, representando as dependências de 1º grau de I_{ds} com V_{gs} e V_{ds} , respectivamente. G_{m2} e G_{m3} descrevem a variação da transcondutância com V_{gs} , e G_{d2} e G_{d3} desempenham o mesmo papel em relação à condutância de saída e V_{ds} . Seguindo este raciocínio, G_{md} e G_{m2d} representam a dependência não-linear de 1º e 2º graus de G_{ds} com V_{gs} , e G_{md2} a dependência de G_m com V_{ds} .

A este respeito vale a pena referir que até ao presente trabalho, os termos cruzados, i. e., envolvendo produtos cruzados entre v_{gs} e v_{ds} , eram postos a zero e as não-linearidades de I_{ds} modeladas por duas fontes dependentes, uma só controlada por V_{gs} , e a outra só por V_{ds} . Assim, o procedimento usual tratava I_{ds} como uma transcondutância e uma condutância de saída não-lineares. Nos artigos publicados por A. Crosmun e S. Maas [4], e S. Maas e D. Neilson [5], o modelo completo de I_{ds} foi pela primeira vez apresentado, tendo sido alertada a possível importância dos termos cruzados. No entanto, só com o desenvolvimento do método de medida proposto por J. Pedro e J. Perez [6,7,8] foi possível caracterizar estes

coeficientes, incluí-los numa análise de intermodulação, e avaliar a sua importância.

Nas figuras Fig. 7 e Fig. 10-12, da próxima secção, podem ver-se graficamente os valores que os coeficientes de (16) e (17) podem assumir num MESFET típico, em função de V_{GS} , e para V_{DS} constante, confortavelmente dentro da zona de saturação.

A principal propriedade que parece caracterizar todas as curvas, é a semelhança de padrões dos coeficientes de igual grau. Por outro lado, verifica-se em G_{m3} a existência de três zonas distintas na variação de I_{DS} com V_{GS} . Não obstante serem difíceis de observar directamente na corrente, e até por vezes na transcondutância, são perfeitamente claras das regiões de sinal positivo e negativo deste coeficiente de 3º grau.

Importa também ressaltar o efeito que o perfil de dopagem gaussiano da fronteira canal-substrato, produz nos coeficientes, conferindo-lhes uma dependência com V_{GS} muito mais complexa que a que teriam se o perfil fosse sempre uniforme. Essa complexidade aumenta com o grau do coeficiente (o que contraria claramente alguns modelos polinomiais) e impõe, junto ao corte, uma característica do tipo exponencial, já algumas vezes tentada modelar por uma tangente hiperbólica, ou por funções mais elaboradas como é o caso de $x + \log[\cosh(x)]$ [9].

Finalmente deve fazer-se referência às origens dos produtos de intermodulação gerados num MESFET. As fontes de distorção, geralmente atribuídas à malha de entrada do modelo do FET, são bem visíveis de G_{m3} e C_{g3} , e decorrem das já estudadas dependências de I_{DS} e Q_G com V_{GS} . Como foi dito, era prática comum dividir $I_{DS}(V_{GS}, V_{DS})$ em duas não-linearidades, justificando as fontes de IMD da saída à dependência de G_{DS} com V_{DS} . No entanto, de acordo com o estudo do funcionamento do dispositivo, G_{DS} em saturação é praticamente constante com V_{DS} , exibindo não desprezável variação com V_{GS} (comparem-se as grandezas relativas de G_{m1} e G_{m2} - Fig. 11 - com as de G_{d2} e G_{d3} - Fig. 12).

IV. CARACTERIZAÇÃO DO DISPOSITIVO - EXTRACÇÃO DE PARÂMETROS -

Na secção anterior fez-se uma breve revisão do funcionamento do GaAs MESFET, tendo-se apresentado um modelo não-linear que visa a sua descrição. Esse modelo é realmente um circuito equivalente que inclui elementos lineares e não-lineares, representados por expressões funcionais do seu comportamento tensão-corrente. Fez-se notar também, que qualquer modelo pressupõe um conjunto de parâmetros (os referidos coeficientes das expansões em série), cuja função é permitir a aproximação da resposta do modelo matemático, à resposta obtida com um particular dispositivo físico. A presente secção ocupar-se-á do estudo dos procedimentos necessários a uma conveniente extracção de parâmetros.

Com o objectivo de ilustrar as técnicas experimentais que se usam no procedimento de extracção de parâmetros, usou-se um dispositivo encapsulado, o FLC091WF da Fujitsu, destinado à amplificação de potência na banda C. Este é um MESFET capaz de fornecer cerca de 27dBm no ponto de 1dB de compressão, e que ilustra bem o tipo de transistor a usar como dispositivo activo principal numa malha de realimentação não-linear.

A. Extracção do Circuito Equivalente do MESFET

O método escolhido no presente trabalho para a caracterização da parte linear do modelo, i. e., a identificação do seu circuito equivalente, usa a combinação de um procedimento inicial, analítico, que segue de perto o trabalho de G. Dambrine *et al.* [10], seguido de posterior optimização.

A técnica analítica seguida divide-se em duas fases. A primeira parte corresponde à chamada *cold modeling*, e baseia-se em medidas de parâmetros S de um FET não polarizado ($V_{DS}=0V$). Com estas medidas obtêm-se, na zona de baixas frequências, as resistências parasitas, e, numa região de frequências mais elevadas, as capacidades e indutâncias extrínsecas. Na segunda fase faz-se *hot modeling*, o que permite a extracção dos restantes elementos, com base em medições de parâmetros S em toda a banda de trabalho do dispositivo, e para as condições de polarização encontradas em operação normal. Esta sequência produz um circuito equivalente com capacidade de descrever um FET não encapsulado, até pelo menos 26.5GHz [10].

O FLC091WF foi caracterizado desde 45MHz a 8GHz num sistema automático de medida de parâmetros S. Com ele se extraíram os distintos valores dos elementos do modelo intrínseco, correspondentes a um varrimento de V_{GS} desde -4.0V a 0.0V, com intervalos de 0.1V, e V_{DS} fixo a 6.0V. Os valores obtidos encontram-se representados nos gráficos das figuras Fig. 3 (C_{gd} , C_{gs}) e Fig. 4 (G_m , G_{ds}).

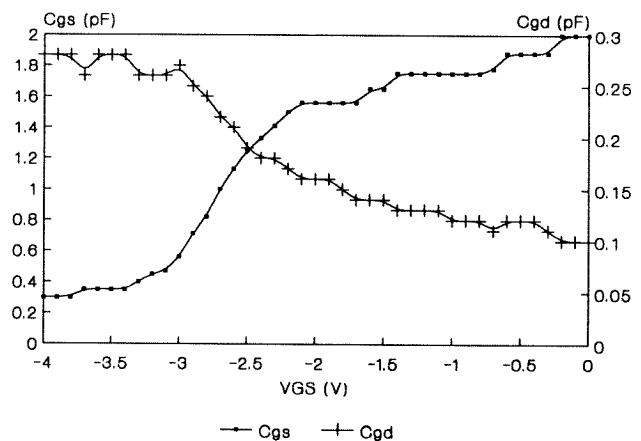


Fig. 3 - C_{gs} e C_{gd} extraídos do FLC091WF
para $-4.0V < V_{GS} < 0.0V$ e $V_{DS} = 6.0V$.

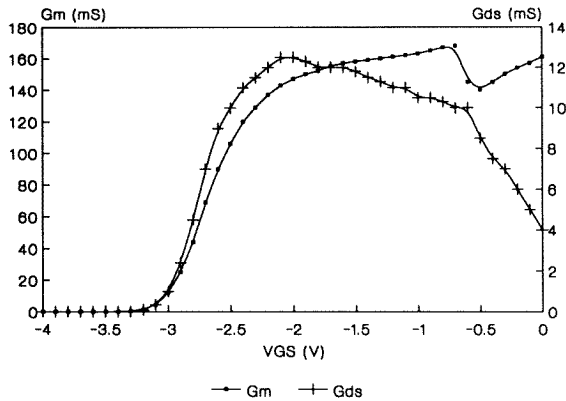


Fig. 4 - Gm e Gds extraídos do FLC091WF para $-4.0V < V_{GS} < 0.0V$ e $V_{DS} = 6.0V$.

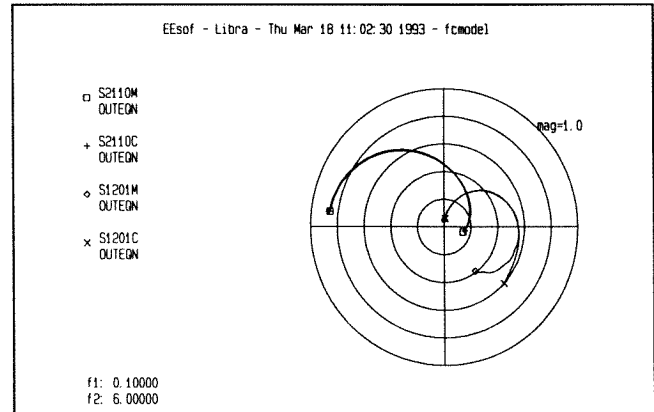


Fig. 6 - Comparação entre os parâmetros S_{12} ($\times 10$) e S_{21} ($\times 0.1$) do modelo de circuito equivalente do FLC091WF e os medidos no dispositivo.

Como se disse, o método seguido para a extração dos elementos do modelo de circuito equivalente seguiu duas fases: obtenção de valores iniciais dos elementos e optimização final. Escolhido o ponto de polarização conducente a uma corrente dreno-fonte próxima da aconselhada pelo fabricante ($I_{DS} \approx 190mA$, $V_{GS} = -1.8V$), optimizaram-se manualmente os valores do modelo obtido acima, com vista à reprodução dos parâmetros S medidos no dispositivo. Os valores resultantes da optimização são os descritos no circuito da Fig. 9.

Os resultados medidos no FLC091WF, e simulados com o modelo optimizado, encontram-se nas Fig. 5 (S_{11} e S_{22}) e Fig. 6 (S_{12} e S_{21}). A validade do modelo obtido aparece bem clara da observação destas figuras.

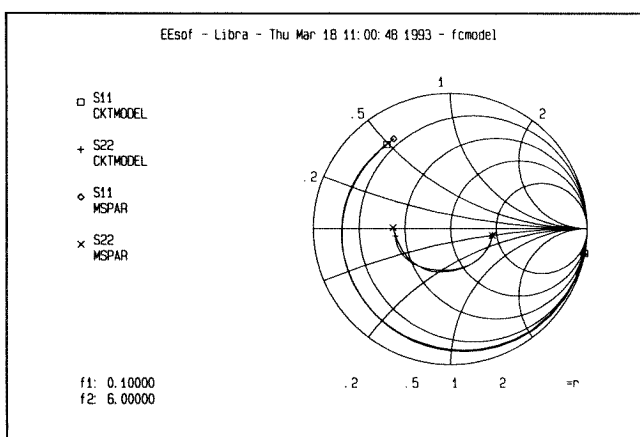


Fig. 5 - Comparação entre os parâmetros S_{11} e S_{22} do modelo de circuito equivalente do FLC091WF e os medidos no dispositivo.

B. Obtenção dos Coeficientes de $C_{gs}(V_{gs})$

Como se viu nas secções anteriores, a caracterização das não-linearidades do dispositivo para simulações de IMD em sinal fraco, resume-se à extração dos coeficientes das expansões em Série de Taylor de $I_{ds}(V_{gs}, V_{ds})$ e de $C_{gs}(V_{gs})$.

Para a obtenção de C_{gs} , C_{g2} e C_{g3} , propõe-se a derivação sucessiva da curva de C_{gs} obtida na determinação do modelo linear (Fig. 3). Na Fig. 7 encontram-se representadas duas curvas de C_{gs} . Uma corresponde aos valores medidos, e a outra foi extraída desta por suavização manual da ondulação induzida pelos erros de medida. Esta última serviu de base à posterior derivação numérica que gerou os coeficientes de ordem superior.

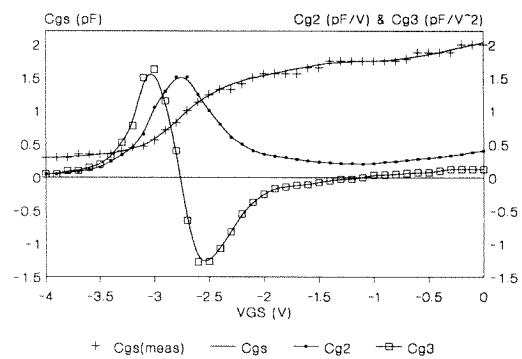


Fig. 7 - C_{gs} , C_{g2} e C_{g3} do FLC091WF, obtidos por derivação sucessiva.

C. Obtenção dos Coeficientes de $I_{ds}(V_{gs}, V_{ds})$

A obtenção dos coeficientes da expansão em série de Taylor de $I_{ds}(V_{gs}, V_{ds})$ tem vindo a ser melhorada ao longo dos últimos vinte anos, tendo-se proposto diversos métodos de extração. Por vezes ajustam-se estes coeficientes a medidas DC de $I_{ds}(V_{gs}, V_{ds})$ e às suas derivadas, ou mais vulgarmente, constroem-se polinómios

que aproximam a dependência de G_m com V_{gs} e G_{ds} com V_{ds} , segundo um critério de minimização dos erros médios quadráticos. Foram também publicados trabalhos, em que a determinação dos coeficientes não-lineares é feita por um processo de optimização que tenta aproximar previsões de distorção de intermodulação obtidas com o modelo, a medidas previamente realizadas.

Recentemente, Maas e Crosmun [11] propuseram um método baseado em medições de distorção harmónica de baixa frequência, que permite extrair directamente G_m , G_{m2} e G_{m3} . Com o objectivo de minimizar as contribuições devidas aos termos de saída, esta foi carregada com uma resistência muito baixa: $R_L \ll 1/G_{ds}$. Dado que este procedimento tem o inconveniente de diminuir muito a tensão desenvolvida em R_L , o que conduz a más relações sinal-ruído, os referidos autores abandonaram esta restrição em trabalho apresentado posteriormente [4]. Com isto eliminaram a necessidade da aproximação acima, mas assumiram implicitamente que os termos de saída influem somente a componente linear da resposta do FET.

Com o fim de criar um sistema de medida capaz de medir directa e separadamente todos os sete coeficientes não-lineares que caracterizam $I_{ds}(V_{gs}, V_{ds})$, foi concebido um método de medida que tira partido da unilateralidade do dispositivo em baixas frequências. De facto, a injeção de um sinal de teste directamente pelo porto de saída permite avaliar, quase independentemente, os termos da saída.

O método usa a montagem experimental representada pelo diagrama da Fig. 8. As frequências dos dois sinais foram escolhidas de modo a eliminar a distorção induzida por $C_{gs}(V_{gs})$, e a influência dos elementos reactivos do circuito equivalente, bem como os efeitos dispersivos de baixa frequência do MESFET. Assim, a montagem final usou $f_1=145\text{MHz}$ e $f_2=155\text{MHz}$. Estes sinais de baixa amplitude são previamente "limpos" de qualquer distorção harmónica residual introduzida pelos geradores, por meio de filtros passa-baixo elípticos de 5ª ordem (LPF), cujos zeros de transmissão foram cuidadosamente colocados a $2f_0$ e $3f_0$.

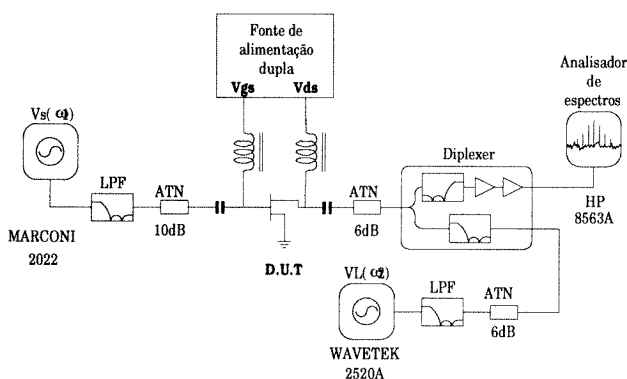


Fig. 8 - Diagrama de blocos do banco de teste de medidas de distorção harmónica.

O diplexer serve dois objectivos. O ramo passa-baixo permite a injeção do sinal a f_2 no dreno, e garante um nível de impedância de carga ao FET próximo dos 50Ω , quer a f_1 quer a f_2 . O ramo passa-alto é, como o anterior, elíptico de 5ª ordem, com um dos zeros de transmissão colocado a 150MHz , e logo seguido de um amplificador de banda larga construído com dois amplificadores monolíticos de UHF.

Com o sistema apresentado mediram-se as relações de potência entre $P_o(\omega_1)$ e $P_o(\omega_2)$, e os produtos de mistura de 2ª ordem a $2\omega_1$, $\omega_1+\omega_2$ e $2\omega_2$, e os de 3ª ordem a $3\omega_1$, $2\omega_1+\omega_2$, $\omega_1+2\omega_2$ e $3\omega_2$.

Das primeiras 3 relações de potência gera-se um sistema linear de 3×3 , em que previsões de distorção harmónica de 2ª ordem são comparadas com os valores medidos. A técnica de análise proposta é o Método das Correntes Não-Lineares das Séries de Volterra [3]. O sistema de equações que descreve a resposta linear, ou de 1ª ordem, do circuito representado na Fig. 9 é :

$$\begin{cases} \begin{bmatrix} I_1 \\ I_2 \\ V_{gs} \\ V_{ds} \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & AI_{D1} \\ Y_{21} & Y_{22} & AI_{D2} \\ AV_{G1} & AV_{G2} & AR_G \\ AV_{D1} & AV_{D2} & AR_D \end{bmatrix} \times \begin{bmatrix} V_1 \\ V_2 \\ NI_{ds} \end{bmatrix} \\ V_1 = V_s - Z_s \cdot I_1 \\ V_2 = V_L - Z_L \cdot I_2 \end{cases} \quad (18)$$

Esta equação matricial relaciona os fasores das tensões de controlo, V_{gs} e V_{ds} , com as tensões e correntes dos portos de entrada e saída do dispositivo V_1 , V_2 e I_1 , I_2 , e as componentes não-lineares de ordem n da fonte de corrente I_{ds} , $NI_{ds}^{(n)}$. Os Y_{ij} são os parâmetros Y do FET, e os AV 's, AI 's e AR 's são ganhos em tensão, corrente e transimpedância. O seu cálculo é imediato aplicando o teorema da sobreposição de fontes a V_1 , V_2 e NI_{ds} . As outras duas equações traduzem as condições fronteira dos respectivos portos, impostas pelas fontes (V_s, Z_s) e (V_L, Z_L) .

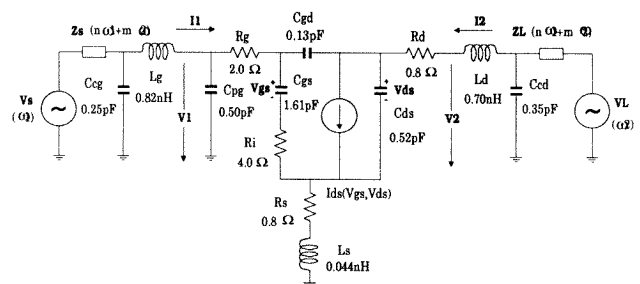


Fig. 9 - Circuito equivalente usado na análise não-linear.

Expressando V_{gs} e V_{ds} como função dos estímulos V_s , V_L e NI_{ds} , obtém-se :

$$\begin{cases} V_{gs} = K_{GS} \cdot V_s + K_{GL} \cdot V_L + K_{RG} \cdot NI_{ds} \\ V_{ds} = K_{DS} \cdot V_s + K_{DL} \cdot V_L + K_{RD} \cdot NI_{ds} \end{cases} \quad (19)$$

onde as constantes K_{GS} , K_{GL} , K_{RG} , K_{DL} e K_{RD} se obtêm directamente da solução de (18).

A título de exemplo, a componente de 2ª ordem a $\omega_1 + \omega_2$, é dada por:

$$NI_{ds}^{(2)}(\omega_1 + \omega_2) = \left\{ 2 \cdot G_{m2} \cdot K_{GS} \cdot K_{GL} + G_{md} \cdot [K_{GS} \cdot K_{DL} + K_{GL} \cdot K_{DS}] + 2 \cdot G_{d2} \cdot K_{DS} \cdot K_{DL} \right\} \cdot \frac{V_s(\omega_1) \cdot V_L(\omega_2)}{2} \quad (20)$$

As tensões de saída a estas frequências são então calculadas resolvendo (19) com $V_s^{(2)} = V_L^{(2)} = 0$. Daqui, as potências de saída $P_o(2\omega_1)$, $P_o(\omega_1 + \omega_2)$ e $P_o(2\omega_2)$, são facilmente obtidas, bem como as suas relações com $P_o(\omega_1)$ e $P_o(\omega_2)$. Igualando estes resultados aos obtidos experimentalmente, gera-se um sistema de 3 equações a 3 incógnitas, cuja solução é o terno (G_{m2}, G_{md}, G_{d2}) .

Os valores destes coeficientes são posteriormente utilizados na construção de um sistema similar ao anterior, mas de 4 equações nas incógnitas G_{m3} , G_{m2d} , G_{md2} e G_{d3} .

As Fig. 10 a Fig. 12 ilustram a aplicação do método ao transistor FLC091WF em estudo. Nelas apresentam-se os resultados obtidos para todos os coeficientes não-lineares, desde o corte, $V_{GS} = -4.0V$, até $V_{GS} = -0.6V$.

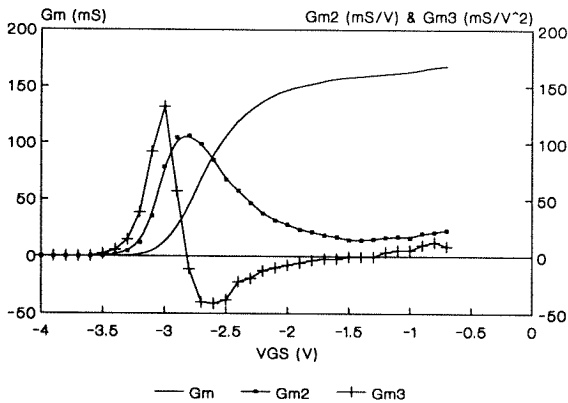


Fig. 10 - Coeficientes de I_{ds} , G_m , G_{m2} e G_{m3} , medidos no FLC091WF @ $V_{DS} = 6.0V$.

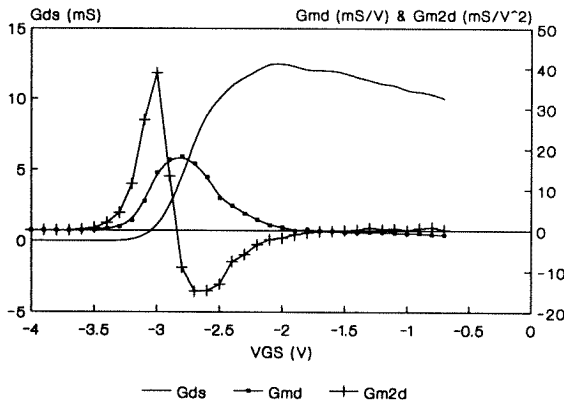


Fig. 11 - Coeficientes de I_{ds} , G_{ds} , G_{md} e G_{m2d} , medidos no FLC091WF @ $V_{DS} = 6.0V$.

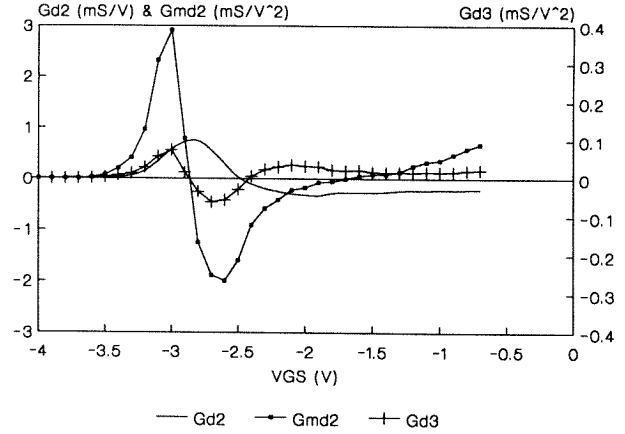


Fig. 12 - Coeficientes de I_{ds} , G_{d2} , G_{md2} e G_{d3} , medidos no FLC091WF @ $V_{DS} = 6.0V$.

V. VALIDAÇÃO EXPERIMENTAL

De forma a validar experimentalmente as ferramentas teóricas desenvolvidas, foi projectado, construído e testado, um amplificador a transistores GaAs MESFET, linearizado mediante a aplicação de realimentação activa ao nível do dispositivo. Este circuito, montado sobre tecnologia monolítica de Arsenieto de Gálio, MMIC (Monolithic Microwave Integrated Circuit), constitui estado-de-arte no respeitante à tecnologia e aos métodos de projecto utilizados, já que é o primeiro sistema linearizado em microondas, completamente integrado numa única pastilha MMIC [2,12].

O diagrama esquemático, e o respectivo *layout*, do circuito implementado, encontram-se representados nas figuras Fig. 13 e Fig. 14.

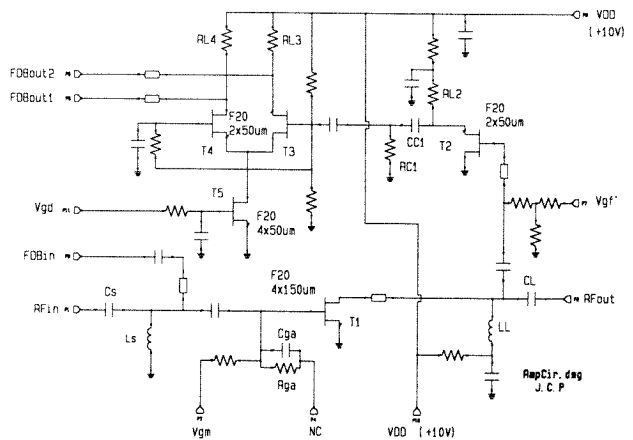


Fig. 13 - Diagrama esquemático do amplificador linearizado.

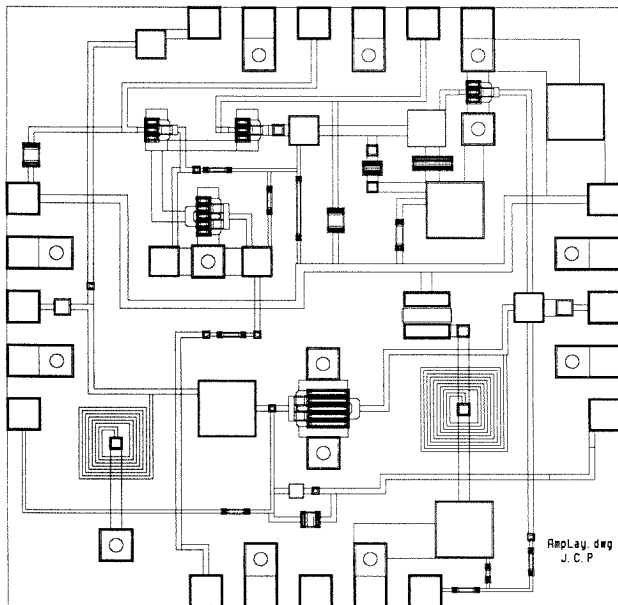


Fig. 14 - Layout da pastilha MMIC implementada.

O circuito consiste num amplificador linearizado por realimentação activa, seguindo portanto, o diagrama de blocos geral da Fig. 1.

O Amplificador Principal, Pri, é um simples amplificador sintonizado para a banda S, de um só andar, em que o elemento activo é um FET F20_4x150, T_1 , e as malhas de adaptação são constituídas por redes passa-alto C-série L-paralelo (C_S/L_S , C_L/L_L).

O linearizador propriamente dito, é constituído por um Amplificador Auxiliar, Amp Aux, seguido de um circuito linear de condicionamento de sinal, C_1 . Este inclui uma rede RC passa-alto (R_{C1}/C_{C1}) e um Amplificador Diferencial com controlo de ganho, Amp Dif. Com o objectivo de reduzir a variação de fase do ramo de realimentação com a frequência (determinante na largura de banda efectiva do linearizador), a simplificação do circuito de implementação de C_2 foi levada ao limite de o fazer fixo e aproximadamente unitário.

O núcleo do linearizador é o Amp Aux. Ele é simplesmente um FET F20_2x50 μ m, T_2 , cuidadosamente polarizado para baixo ganho e elevada distorção de intermodulação.

Como se pôde deduzir do estudo teórico realizado, para garantir simultaneamente invariância das características lineares de Amp Pri, e estabilidade da malha, o Amp Aux deveria apresentar baixo ganho de transadmitância, e alta IMD. Isto conseguiu-se escolhendo a estrutura MESFET disponível de menor área (menor G_m), polarizando-a numa zona de elevada relação G_{m3}/G_m . Note-se que, estando este FET ligado em derivação com a saída do Amp Pri, o facto de ser de pequena área e estar polarizado a baixo nível de corrente I_{ds} diminui também a sua influência sobre o ramo principal.

Os estudos de modelação não-linear realizados ao longo do trabalho, revelaram que um GaAs MESFET apresenta pelo menos uma zona de G_{m3} positivo (junto ao

ponto de V_{GS} de entrada em condução), e outra de G_{m3} negativo (junto a $I_{dss}/2$). Isto justifica a sua versatilidade como gerador de IMD auxiliar, aproveitada em vários linearizadores de Pré-Distorção. Também num linearizador por realimentação activa esta propriedade pode ser utilizada, permitindo construí-lo com realimentação positiva ou negativa, consoante a zona de polarização do dispositivo de Amp Aux. Para comprovar experimentalmente esta hipótese, desenhou-se a pastilha de forma a permitir a escolha de uma, entre duas configurações opcionais do ramo de realimentação, proporcionadas pelas saídas em oposição de fase do amplificador diferencial de C_1 , FDBout1 e FDBout2.

Escolhido o dispositivo e o seu ponto de polarização (aproximadamente 20% de I_{dss}), o projecto de Amp Aux resume-se à selecção da impedância de carga, Z_L . Ao contrário do requerido para Amp Pri, em Amp Aux deseja-se elevada distorção e baixo ganho. Assim, simulações de load-pull indicaram como óptimo, um baixo valor de Z_L , o qual minimiza a compensação das contribuições de IMD devidas a G_{m3} e G_{m2d} . Um valor baixo de Z_L , em conjunto com a impedância de entrada do andar seguinte, constituem um divisor de corrente que diminui ainda mais o ganho total de transadmitância. No outro ponto de polarização, i. e., com o dispositivo activo auxiliar quase cortado, a discussão acima é praticamente irrelevante, pois que, tendo-se aí $G_m \approx 0$, a tensão de saída de sinal será tão baixa que a IMD se torna insensível a variações de Z_L .

De forma a estudar o comportamento do linearizador em banda, sujeitou-se este a um teste de dois tons, separados de 2MHz, e de frequência central variável entre 1400MHz e 2050MHz. A Fig. 15 representa os resultados obtidos, com todas as tensões de polarização ou ajuste fixas.

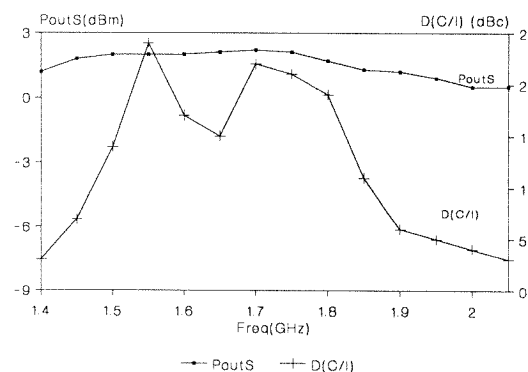


Fig. 15 - Resultados medidos de melhoria de C/I com a frequência.

É visível uma melhoria da relação portadora-intermodulação de 15dB, numa largura de banda relativa superior a 12%. Este é um resultado notável para um linearizador por realimentação (por exemplo quando comparado com os cerca de 0.25% publicados para realimentação linear), e que demonstra a utilidade da técnica de linearização ao nível do dispositivo activo.

Para ilustrar as duas condições de linearização em alto e baixo nível, previstas na abordagem teórica, T_2 foi

polarizado a cerca de 20% de I_{dss} , e o ganho de Amp Dif (amplitude de C_1) ajustado a quatro valores distintos. Como se pode observar da Fig. 16, onde são representados resultados experimentais e de simulação, à medida que a amplitude do sinal de IMD auxiliar injectado à entrada de Amp Pri sobe (V_{gd} crescente), produz cada vez maior nível de cancelamento da IMD de 3ª ordem. Até que, para $V_{gd} = -1.5V$, esta se anula completamente atingindo-se a referida "Condição de Compensação em Baixo Nível" (repare-se no declive de 5dB/dB, sintoma da única contribuição residual da distorção de ordem 5). Aumentando ainda mais $|C_1|$, atinge-se a "Condição de Compensação em Alto Nível" identificada pelo pico de cancelamento da IMD perto de -4dBm de potência do sinal de entrada.

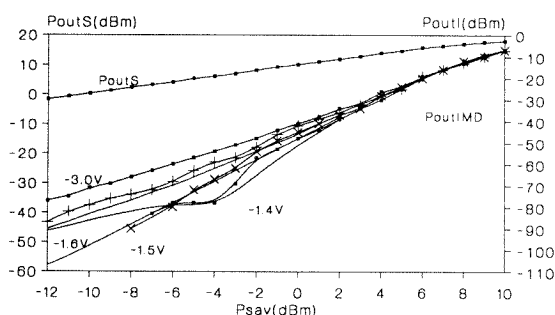


Fig. 16 - Resultados simulados (—) e medidos (.,+,x) de compensação de IMD para quatro valores de ganho da malha: $V_{gd} = -3.0V$ (malha aberta), $-1.6V$, $-1.5V$ e $-1.4V$; $V_{gf} = -3.0V$.

A hipótese de que um linearizador por realimentação não-linear tem efeito desprezável sobre o ganho do amplificador a linearizar (ao contrário da técnica de linearização por realimentação linear), é agora amplamente confirmada pelo facto de que em todos os casos acima, a máxima redução observada na potência de saída (tendo a situação em malha aberta - $V_{gd} = -3.0V$ - como referência) ter sido inferior a 0.5dB. Além disso, parte desta variação (só por si diminuta face aos ganhos em margem de C/I alcançados), deve ainda ser atribuída a variações na admitância de saída de Amp Dif, que, estando colocado sobre a entrada de Amp Pri, se manifestam como uma fonte variável de perdas e desadaptação.

VI. CONCLUSÕES

Em conclusão, apresentou-se uma possível solução para o problema da integração do amplificador de potência e seu linearizador. Esta solução foi suportada por um novo conceito de linearização ao nível do dispositivo activo, e por um conveniente estudo teórico do circuito de realimentação não-linear. Métodos originais de extracção de modelos não-lineares do GaAs MESFET, permitiram o uso de ferramentas de CAD/CAE (especialmente desenvolvidas para o efeito) no projecto do primeiro amplificador linearizado, totalmente integrado numa única pastilha MMIC. Os resultados de simulação e

experimentais obtidos com este circuito, validaram os métodos teóricos desenvolvidos, e demonstraram a utilidade das hipóteses formuladas.

AGRADECIMENTOS

Os autores gostariam de agradecer o apoio concedido pela E.T.S.I. de Telecomunicacion de la Universidad Politécnica de Madrid, e a Bolsa de Estudos para Doutoramento, da Comissão Permanente INVOTAN.

REFERÊNCIAS

- [1] J. C. Pedro, Técnicas de Linearização de Amplificadores de Potência em Microondas, Dissertação de Doutoramento, Universidade de Aveiro, Agosto 1993.
- [2] J. C. Pedro e J. Perez, "A novel concept of intermodulation distortion linearization", Microwave and Optical Technology Letters, vol. 6, nº 7, pp.410-415, Junho 1993.
- [3] S. A. Maas, Nonlinear Microwave Circuits, Artech House Inc., Norwood, MA, 1988.
- [4] A. Crosmun e S. Maas, "Minimization of intermodulation distortion in GaAs MESFET small-signal amplifiers", IEEE Trans. on Microwave Theory and Tech., vol. MTT-37, nº 9, pp.1411-1417, Setembro 1989.
- [5] S. A. Maas e D. Neilson, "Modeling MESFETs and diodes for intermodulation calculations", Integrated Nonlinear Microwave and Millimeterwave Circuits (INMMC'90), pp.81-94, Universitat Duisburg, Outubro 1990.
- [6] J. C. Pedro and J. Perez, "An improved MESFET model for the prediction of intermodulation load-pull characterization", 1992 Intern. Microwave Theory and Tech. Symposium Digest, pp.825-828, Albuquerque, Junho 1992.
- [7] J. C. Pedro e J. Perez, "Complete characterization of MESFET Ids nonlinearities for intermodulation analysis", Microwave and Optical Technology Letters, Vol. 5, No. 8, pp.377-380, Julho 1992.
- [8] J. C. Pedro e J. Perez, "Accurate simulation of GaAs MESFET's intermodulation distortion using a new drain-source current model", IEEE Trans. on Microwave Theory and Tech., vol. MTT-42, nº 1, pp.25-33, Janeiro 1994.
- [9] J. C. Pedro e J. Perez, "A novel nonlinear GaAs FET model for intermodulation analysis in general purpose harmonic balance simulators", 23rd European Microwave Conference Proc., pp.714-716, Madrid, Setembro 1993.
- [10] G. Dambrine, A. Cappy, F. Héliodore e E. Playez, "A new method for determining the FET small-signal equivalent circuit", IEEE Trans. on Microwave Theory and Tech., vol. MTT-36, nº 7, pp.1151-1159, Julho 1988.
- [11] S. Maas e A. Crosmun, "Modeling the gate I/V characteristics of a GaAs MESFET for Volterra-Series analysis", IEEE Trans. on Microwave Theory and Tech., vol. MTT-37, nº 11, pp.1134-1136, Julho 1989.
- [12] J. C. Pedro e J. Perez, "An MMIC linearized amplifier using active feedback", 1993 IEEE International Microwave and Millimeter-Wave Monolithic Circuits Symposium Digest, pp.113-116, Atlanta, Junho 1993.

Determinação de Mapas de Actividade Cerebral numa Banda de Frequências

Carlos Loff Barreto, Francisco Vaz

Resumo- Apresenta-se neste trabalho um novo método para a determinação de mapas de actividade cerebral baseado na estimação da potência do sinal filtrado nos diferentes canais, substituindo a sua tradicional e mais lenta determinação através da DFT. Desta forma pode-se diminuir o tempo de cálculo permitindo obter uma melhor visualização dinâmica da evolução dos potenciais cerebrais sobre o escalpe.

Abstract- In this paper we present a new method to compute brain maps. It is based on the power estimation of the filtered signal on all channels which is faster to compute than the traditional approach using DFT. Reducing the time needed for computation we can have a better dynamic display of the evolution of the brain potentials on the scalp.

I. INTRODUÇÃO*

O electroencefalograma (EEG), sinal recolhido sobre o escalpe e resultante da actividade bioeléctrica do cérebro, é ainda o principal meio não invasivo de investigação e de diagnóstico das patologias cerebrais. Habitualmente a recolha do sinal é feita recorrendo a 16 (ou mais) eléctrodos colocados sobre o escalpe segundo uma grelha espacial normalizada conhecida como o sistema 10-20 [1].

A análise do EEG pode ser feita, e é-o em grande parte das aplicações, por análise visual da evolução temporal dos 16 canais observados e registados graficamente em papel. Atendendo a que em muitas situações é necessário registar algumas horas de sinal, é clara a vantagem que os meios de tratamento automático podem trazer a esta disciplina clínica.

O médico especialista procura no traçado dos sinais a ocorrência de formas de ondas típicas e a partir das suas características (amplitude, frequência, duração, etc) e de correlações entre canais, extrai a informação que lhe permite o diagnóstico. Um dos métodos usados para mais facilmente visualizar as correlações espaciais é o chamado mapa de actividade cerebral que é uma representação das linhas equipotenciais no escalpe determinadas por métodos de interpolação a partir dos valores do sinal medido nos 16 ou mais canais disponíveis. Estes mapas aparecem como uma imagem colorida, correspondendo cada cor ao intervalo entre dois valores de tensão do sinal. Atendendo à tão baixa frequência de amostragem espacial, o mapa a duas dimensões que se obtém é uma representação não muito precisa mas que se revela muito

útil por resumir numa única imagem o comportamento conjunto dos diferentes canais.

Em geral o simples mapa determinado a partir dos potenciais instantâneos não se revela muito útil, preferindo-se o mapa da potência do sinal em certas bandas de frequência.

A determinação da potência do sinal numa banda de frequência é feita calculando a transformada de Fourier de uma janela temporal do sinal (com uma duração dependente da resolução na frequência que se pretende) e efectuando em seguida a integração entre os limites de frequência desejados. Deslocando-se a janela poder-se-á então determinar a evolução temporal da potência do sinal, e a partir dos valores assim determinados para todos os canais construir um mapa da actividade cerebral que varia ao longo do tempo.

Neste trabalho propomos a determinação da potência do sinal por um outro método que consiste na filtragem directa do sinal na banda desejada e posterior estimativa de potência. O processamento directo no tempo permite obter um valor estimado de potência para cada instante de amostragem, independente de uma janela temporal como no método tradicional. Todo o sistema será construído em torno de um computador pessoal baseado num microprocessador Intel 80486, usando a linguagem C como ferramenta de programação e integrando este trabalho num sistema mais vasto, o LIne - Laboratório Integrado de Neurofisiologia- que tem vindo a ser desenvolvido em conjunto pelo grupo INESC do DETUÁ e o Serviço de Neurofisiologia do Hospital Geral de S. António no Porto [2].

II. MÉTODOS

O EEG é um sinal que na sua actividade de fundo tem um espectro que varia com $1/f$ e apresenta uma ou mais ressonâncias espectrais sempre que haja actividade rítmica. Considera-se que acima de 35 Hz não há em geral conteúdo espectral significativo. Desta forma é corrente amostrar o EEG a 100Hz a que corresponde um período de amostragem de 10ms. O nosso objectivo será efectuar as seguintes operações de processamento que permitem a obtenção de um novo mapa, no mais breve espaço de tempo, por forma a obtermos um efeito de animação na visualização dos mapas:

- i) Aquisição de sinal
- ii) Filtragem
- iii) Estimação da potência

* Trabalho realizado no âmbito da disciplina de Projecto.

- iv) Construção do mapa
- v) Visualização do mapa

A aquisição de sinal é feita usando uma placa DT821 da Data Translation e a sua descrição pode ser encontrada em [2]. O sinal é digitalizado em 12 bits e representado segundo o formato definido por SIGIF [3] ao longo de todo o processamento, mantendo assim a referida compatibilidade com o sistema já desenvolvido.

A. Filtragem

A filtragem faz-se usando dois filtros, um passa-alto e um passa-baixo com frequências de corte seleccionáveis segundo a Tabela 1. O utilizador poderá assim escolher a banda de frequências mais apropriada à sua aplicação. Os filtros implementados são FIR de ordem 61 e foram projectados usando o método "directo" incluído no programa Monarch [4] que corresponde ao método das séries de Fourier [5].

TABELA 1
FREQUÊNCIAS DE CORTE DOS FILTROS PASSA-BAIXO E PASSO-ALTO.

Frequências de corte dos Filtros passa-baixo	Frequências de corte dos Filtros passa-alto
15.0 Hz	10.0 Hz
10.0 Hz	5.0 Hz
7.5 Hz	3.0 Hz
5.0 Hz	2.0 Hz
3.0 Hz	1.0 Hz

B. Estimação da potência

A aplicação de uma não linearidade seguida de uma filtragem passa-baixo com uma frequência de corte apropriada, é um simples método para a estimação da potência. Neste trabalho usou-se, como se esquematiza na figura 1, um rectificador seguido de um filtro IIR passa-baixo com um único polo correspondente a uma frequência de corte de 0.05 Hz (a que corresponde uma constante de tempo de 3.18s). Este valor foi escolhido tendo em atenção que a variação da potência do sinal é, em geral, muito lenta.

A implementação deste filtro é feita através da seguinte equação de diferenças:

$$y(n) = a.x(n) + b.y(n-1)$$

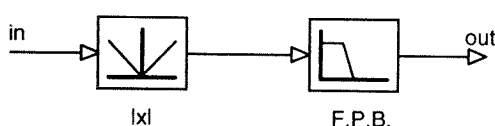


Figura 1- Estimador de potência.

C. Construção dos mapas

O mapa de actividade não vai ser determinado sobre a superfície do escalpe mas sobre um círculo. Uma vez mais se está a usar uma aproximação grosseira no sentido de se simplificarem os cálculos. O círculo é discretizado num número finito de elementos - *pixels* - e em vez de se proceder ao cálculo das equipotenciais, determina-se o potencial para cada um dos *pixels* e atribui-se-lhe uma cor de acordo com uma dada escala.

No nosso caso o número de *pixels* é de 4344, correspondente a um raio de 37 *pixels*, número determinado por um compromisso entre o tempo de cálculo e a resolução da imagem obtida.

Para calcular o potencial num ponto usa-se uma interpolação muito simples: o potencial é determinado pela média dos potenciais nos quatro eléctrodos mais próximos, pesados pelo inverso das distâncias aos eléctrodos. [5].

Para cada ponto *i* teremos então um potencial V_i dado por

$$V_i = \sum_{j=1}^4 \frac{V_j}{d_{ij}} / \sum_{j=1}^4 \frac{1}{d_{ij}}$$

que facilmente se pode desenvolver em

$$V_i = V_{i1} P_{i1} + V_{i2} P_{i2} + V_{i3} P_{i3} + V_{i4} P_{i4}$$

com

$$P_{ij} = \prod_{k=1, k \neq j}^4 d_{ik} / D_i$$

e

$$D_i = d_{i2}d_{i3}d_{i4} + d_{i1}d_{i3}d_{i4} + d_{i1}d_{i2}d_{i4} + d_{i1}d_{i2}d_{i3}$$

Como se pode observar para cada ponto é necessário fazer sempre o mesmo tipo de operações: determinar os 4 electrodos mais próximos, determinar os respectivos pesos e fazer a interpolação. As duas primeiras operações podem ser feitas previamente e usadas sempre que se pretende determinar um mapa. Na nossa implementação foi previamente determinado para cada *pixel* quais os 4 canais mais próximos e respectivos pesos. Esta informação constitui o que chamamos matriz de referência para o método de interpolação e é carregada com o programa desenvolvido. Desta forma para cada *pixel* apenas é necessário efectuar quatro multiplicações e três adições, diminuindo muito o tempo de cálculo necessário.

D. Visualização

A visualização dos mapas obtidos pode ser feita em dois modos distintos. Num deles, pode-se visualizar simultaneamente um mapa e de 1 a 16 sinais de EEG. No

outro só é feita a visualização do mapa, mas quatro vezes maior.

No primeiro caso, cada *pixel* do mapa calculado é representado graficamente por um *pixel* do *display* enquanto que no segundo caso se faz um *zoom* $\times 4$, isto é, cada *pixel* do mapa é representado, por um quadrado de 4×4 *pixels* do *display*.

III. RESULTADOS

Os primeiros testes do sistema desenvolvido num computador pessoal com um processador 80486 da Intel e dotado de um relógio de 33 MHz, permitiram medir os seguintes tempos de processamento para cada uma das tarefas:

TABELA 2
TEMPOS DE PROCESSAMENTO

Aquisição de sinal + Filtragem + Estimação de potência	5.7 ms
Cálculo dos mapas	110 ms
Visualização de sinais EEG e mapa pequeno	35 ms
Visualização do mapa grande	190 ms

Como se pode verificar o tempo de cálculo para a determinação da potência nos 16 canais é inferior ao período de amostragem habitualmente usado. No entanto a determinação dos mapas e sua visualização excede largamente o tempo disponível entre duas amostras consecutivas. Isto não é muito grave para os nossos objectivos já que a taxa de variação da potência é em geral muito baixa e está limitada pela largura de banda do filtro passa-baixo incluído no processamento. Desta forma é possível efectuar uma forte decimação à saída dos estimadores de potência e actualizar a visualização a um ritmo muito mais baixo do que o de amostragem do sinal. Assim, atendendo aos tempos de cálculo indicados para a visualização dos mapas e à disponibilidade de 4.3ms entre amostras consecutivas é possível a taxa de actualização de 3 por segundo correspondente a uma decimação de 32:1 no caso do mapa pequeno, e no caso do mapa grande de 1.5 por segundo correspondente à decimação de 70:1.

Estas taxas são suficientes para se observar a evolução temporal dos mapas, e maiores que as habitualmente obtidas usando a DFT. Estas taxas podem evidentemente ser aumentadas usando um processador com um relógio mais rápido.

Apresentam-se em seguida alguns exemplos da visualização obtidos com o sistema desenvolvido, onde por razões óbvias se eliminou a cor e se reduziu o número de níveis dos mapas apenas a 8. Na figura 2 mostram-se os resultados do processamento temporal: um sinal de EEG, o mesmo sinal filtrado e a estimativa da sua potência na banda de frequência seleccionada. Nas figuras 2 e 3 pode-

se ver um mapa de actividade nas duas versões de visualização já referidas.

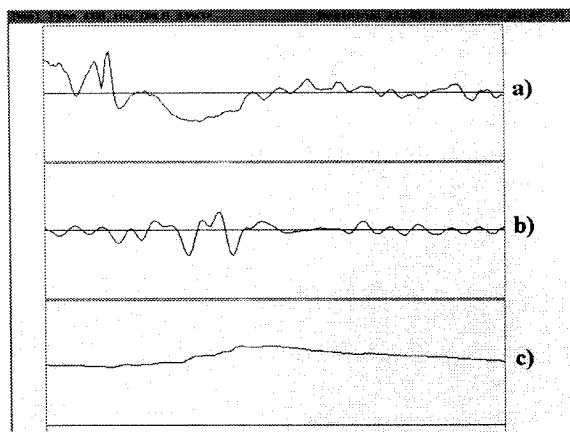


Figura 2: a) Sinal de EEG b) sinal filtrado c) potência estimada

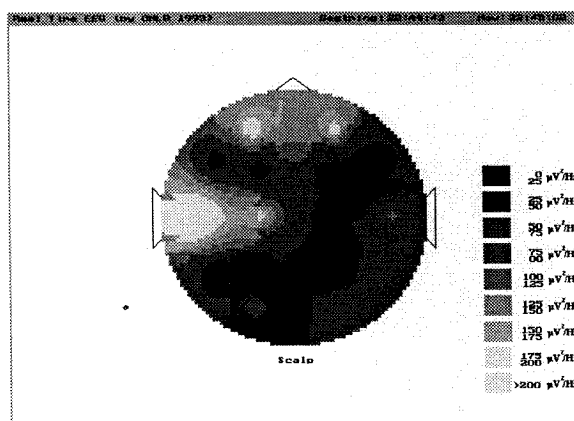


Figura 3: Mapa de actividade cerebral

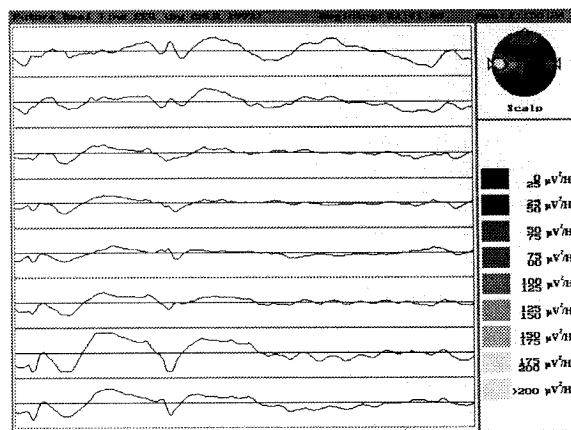


Figura 3: Mapa de actividade cerebral e sinal EEG

REFERÊNCIAS

- [1] Report of the Committee on Methods of Clinical Examination in Electroencephalography of the International Federation 1957

- [2] Cunha, J.P., Guedes de Oliveira, P., Cunha, M.B., Martins da Silva, A.: "Integration of Multimedia information in a clinical neurophysiology department" *Proceedings of the 15th Annual International Conference of IEEE-EMBS*, 624-625, San Diego, Cf, USA, 1993.
- [3] Cunha, M.B., Cunha, J.P., Oliveira e Silva, T.: "SIGIF: A digital signal interchange format for biological signals", *Proceedings of the 15th Annual International Conference of IEEE-EMBS*, 644-645, San Diego, Cf, USA, 1993.
- [4] *MONARCH: User's Manual*, The Athena Group, Inc., Gainesville, FL, USA, 1988.
- [5] Kuc, R.: *Introduction to digital signal processing*, McGraw Hill Book Company, New York, USA, 1988.
- [6] Fred, A.N.L.: *Representação topográfica e classificação automática de sinais electroencefalográficos*. Dissertação de mestrado, IST, Lisboa, 1988.

Sistema de Processamento Digital de Imagem para WINDOWS

Anabela Godinho, Osvaldo Santos, A. Sousa Pereira

Resumo- Este artigo descreve um sistema genérico de Processamento Digital de Imagem em ambiente Microsoft WINDOWSTM. São usadas várias técnicas, entre as quais Transformadas de Imagem (Fourier, Walsh, Hadamard), Processamento sobre o Histograma, Filtragem Espacial e Processamento sobre a paleta de cores. Também é descrita uma técnica de reconhecimento automático de caracteres.

Abstract- The goal of this paper is to describe a generic Digital Image Processing Package in Microsoft WINDOWSTM environment. Several techniques are used, such as Image Transforms (Fourier, Walsh, Hadamard), Histogram processing, Spatial Filtering and Color Map processing. An automatic character recognition technique is also described.

I. INTRODUÇÃO

A área de Processamento Digital de Imagem tem crescido consideravelmente de importância, com a sua crescente utilização em inúmeras aplicações, permitido pela melhoria das performances e redução de preços dos computadores e *hardware* gráfico.

O processamento de imagem desempenha hoje um papel importante em áreas tão diversas como a Medicina, Astronomia, Geografia, Indústria, Ambiente, Exploração espacial, etc.

A lista de aplicações desta área é tão vasta que só é limitada pela imaginação:

- Transmissão de imagens de sondas espaciais.
- Videofone.
- Aquisição de imagens por meios não ópticos.
- Melhoria da qualidade de imagem.
- Classificação automática de terreno, e detecção de recursos disponíveis por análise de imagens de satélite.
- Detecção de agressões ambientais por análise de imagens de satélite.
- Aquisição e reconstrução de imagens médicas.
- Realização automática de mapas a partir de imagens aéreas.
- Detecção de irregularidades em peças de máquinas, a partir de radiografias industriais.

- Reconhecimento automático de formas e objectos.
- etc.

A versão IDIPW 3.0 (*Interactive Digital Image Processing for Windows*) surge como uma versão para Windows 3.1 ou NT do sistema existente para DOS, o IDIPP 2.0.

A grande diferença para a versão DOS é que não é necessário *hardware* dedicado para o sistema funcionar. A utilização do sistema operativo Windows permite uma independência do *hardware* quase total, bem como *multitasking* [2].

Este sistema implementa de uma forma articulada, um conjunto de técnicas de processamento digital de imagem, de tal forma que estas ferramentas podem ter uma utilização genérica, independentemente do tipo de imagens sobre as quais elas são aplicadas.

Foi dado um ênfase especial na facilidade e possibilidade de interacção com o utilizador, criando um ambiente intuitivo e *user-friendly*, procurando sempre que possível um alinhamento com outras interfaces de aplicações Windows conhecidas.

Na última década, houve um interesse crescente em sistemas de reconhecimento de padrões. Dentro deste contexto, foi implementado um sistema adaptativo de aprendizagem e reconhecimento automático de caracteres, que, devido à falta de tempo para o desenvolver, não passa de um protótipo de uma ferramenta, que poderia ter imensas utilizações.

Das técnicas de processamento digital de imagem inicialmente previstas foram implementadas as seguintes:

- *Load/save* de imagens em diversos formatos.
- Definição de zonas de interesse.
- *Hardcopy*;
- *Undo/Redo*.
- Operações com o *clipboard*.
- Conversão de cores para *gray scale*.
- Segmentação por *threshold*.
- Segmentação por crescimento de regiões.
- Transformadas de imagem.
- Filtragem no domínio da frequência.
- Filtragem no domínio espacial.
- Melhoria de imagem.

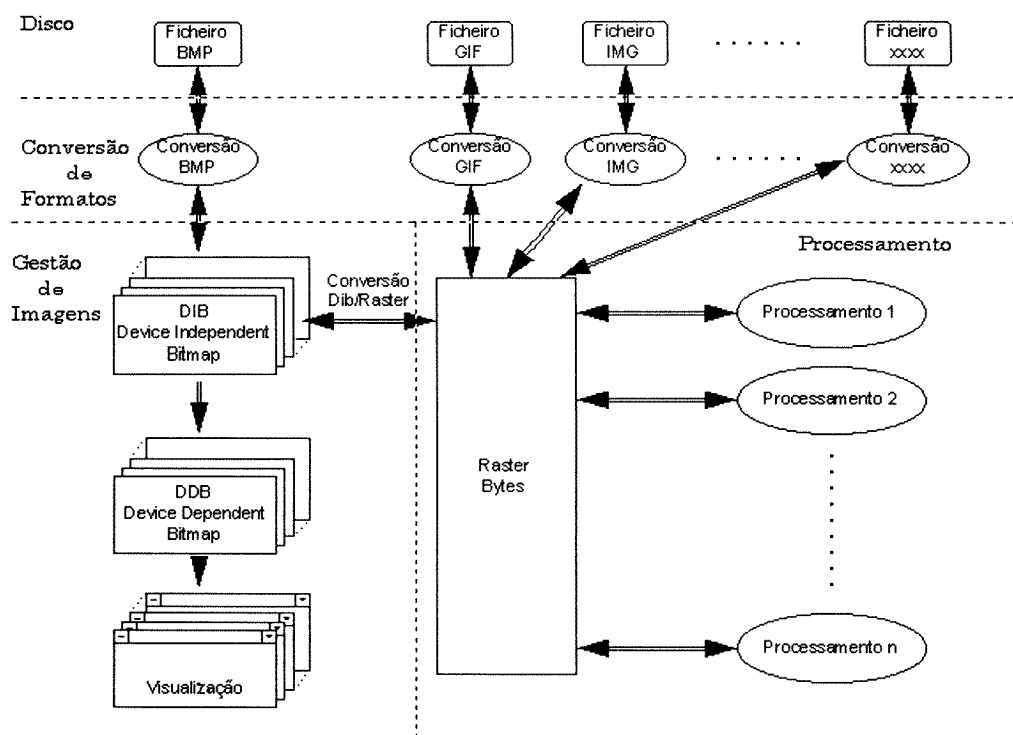


Fig. 1 - Estrutura lógica do programa.

- Visualização do histograma da imagem.
- Operações com o histograma.
- Aprendizagem de caracteres.
- Reconhecimento automático de caracteres.

II. ESTRUTURA DO SISTEMA

O sistema é composto por:

- PC 386 ou superior.
- Sistema Operativo WINDOWS 3.1 ou NT.
- Software IDIPW 3.0.

O sistema não necessita de *hardware* dedicado, no entanto é aconselhável um mínimo de 4 Mbytes de memória RAM, placa gráfica com possibilidade de visualização de 256 cores simultâneas com resolução elevada (1024 x 768 é razoável) e monitor compatível com a placa gráfica, de preferência > 14".

A estrutura lógica do programa (ilustrada na Fig. 1) foi desenvolvida de maneira a facilitar a introdução de novos módulos e ferramentas, independentemente do formato original da imagem, criando um ambiente de suporte às ferramentas de processamento. Este ambiente é responsável pela gestão de múltiplas imagens, fornecendo aos módulos de processamento a visão de uma única imagem, sempre no formato RASTER. Este formato é independente do número de bits por pixel da imagem, bem como do formato original desta. Os módulos de processamento não têm assim que se preocupar com

aspectos como por exemplo a visualização, *load*, *save*, etc.

III. INTERACÇÃO COM O UTILIZADOR

A interacção com o utilizador é feita através de uma interface *WINDOWS user friendly*, utilizando *menu* e *tool bar*, pelo que para um utilizador com alguma experiência neste ambiente não é necessário muito tempo de aprendizagem. A Fig. 2 ilustra um possível cenário de utilização do programa.

A janela principal serve de suporte a toda a interface gráfica do programa. É nesta janela que se encontram o *menu* e a *tool bar*. A *tool bar* permite um rápido acesso às ferramentas mais utilizadas, podendo ser posicionada em quatro posições diferentes, ao gosto do utilizador. É implementada a filosofia *WYSIWYG* (*What You See Is What You Get*), inibindo as ferramentas que não podem ser utilizadas em determinadas situações. Este tipo de filosofia é também aplicado ao *menu*.

IV. ALGUMAS FERRAMENTAS IMPLEMENTADAS

A. Melhoria de Imagem

Este módulo implementa rotinas de melhoria de imagem que permitem alterar dinamicamente a imagem a processar, jogando com cada uma das componentes RGB,

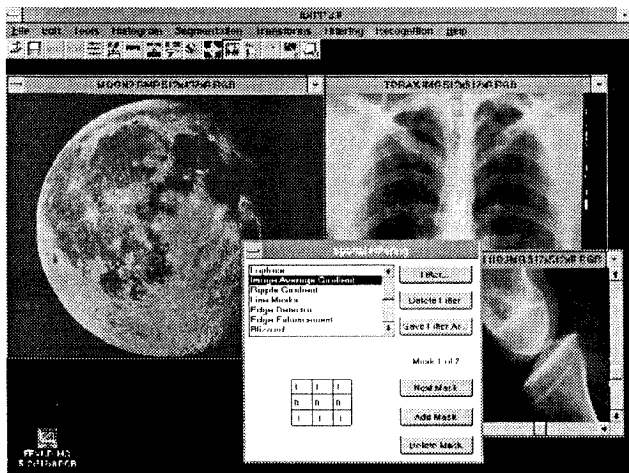


Fig. 2 - Possível cenário de utilização do programa.

ou características como o brilho, contraste e saturação. É utilizada animação dinâmica da *palette* na fase de ajuste, permitindo a visualização em tempo real das alterações efectuadas.

A interacção com o utilizador é feita através de uma caixa de diálogo com seis "potenciômetros", ilustrada na Fig. 3. Cada um destes "potenciômetros" controla uma determinada característica da imagem.

A Fig. 5 representa o resultado da aplicação desta ferramenta na melhoria do brilho e do contraste da imagem representada na Fig. 4.

B. Visualização do histograma.

A visualização do histograma de uma imagem é uma das ferramentas básicas de qualquer sistema de processamento digital de imagem. A ferramenta implementada permite visualizar o histograma de imagens de 8 *bits/pixel*, *gray scale* ou a cores. Como complemento, permite duas escalas: linear e logarítmica. Permite também a visualização do número de *pixels* de determinada entrada da *palette* e as suas componentes RGB. A Fig. 6 ilustra o aspecto da janela de visualização do histograma.

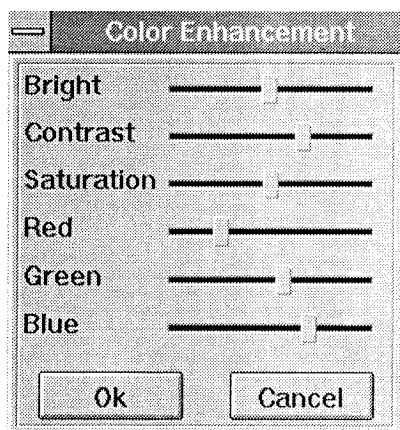


Fig. 3 - Janela da melhoria de imagem.

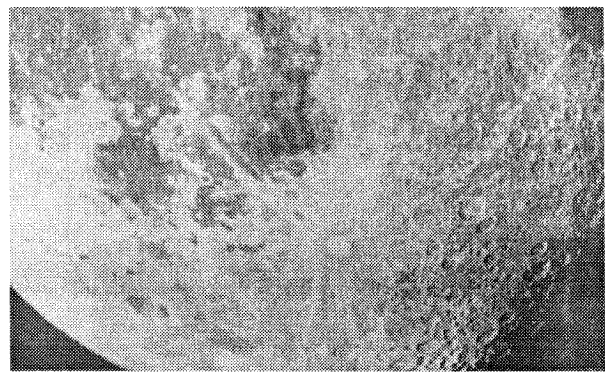


Fig. 4 - Imagem original.

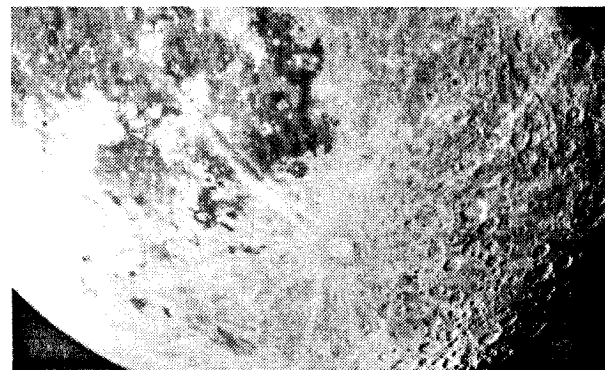


Fig. 5 - Imagem processada com melhoria de contraste.

C. Operações sobre o histograma.

Esta ferramenta permite a definição da curva de distribuição do histograma. Esta técnica permite melhorar o contraste da imagem, definindo as zonas de concentração do histograma. Para obter a lei de distribuição pretendida o utilizador pode optar por uma das quatro leis pré-definidas. Se desejar uma lei de distribuição não contemplada, ou retocar uma das existentes, poderá editá-la, desenhando a curva com o rato. A Fig. 7 ilustra o aspecto da janela de interacção.

D. Segmentação por Threshold.

A segmentação por *Threshold* é utilizada em imagens em *gray scale*, e permite redefinir o nível de cinzento dos *pixels* dentro de uma determinada gama de níveis de cinzento. A cor dos *pixels* pode ser definida pelo

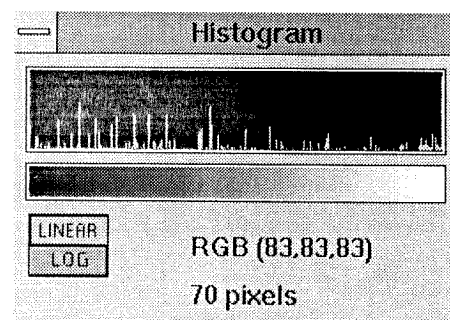


Fig. 6 - Janela de visualização do histograma.

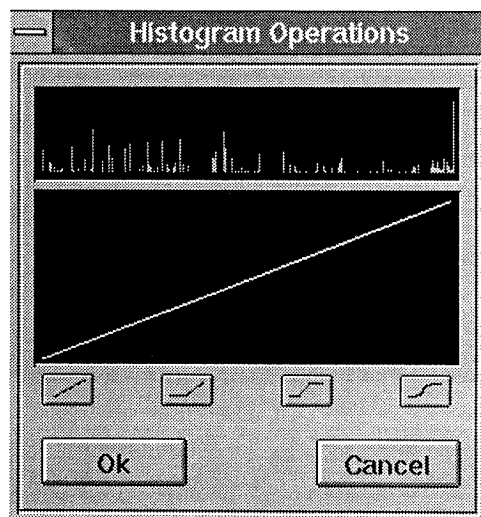


Fig. 7 - Janela das operações com o histograma.

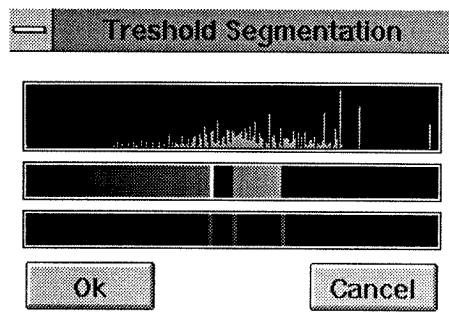
utilizador ou pode ser a escala normal (*gray scale*). Podem ser criadas até vinte gamas de segmentação diferentes.

As alterações definidas pelo utilizador (que podem ser: criação de novas gamas, movimentação das fronteiras das gamas, movimentação de toda uma gama, alteração dos níveis de cinzento dentro de cada gama, etc) reflectem-se na imagem a segmentar em tempo real, devido à utilização de animação dinâmica da *palette*. A Fig. 8 ilustra a janela de interacção com o utilizador.

E. Segmentação por crescimento de regiões.

A segmentação por crescimento de regiões agrega um conjunto de *pixels* em torno de um ponto semente com características tonais próximas. Pode-se assim utilizar esta ferramenta para isolar objectos de uma imagem, desde que estes tenham características tonais diferentes do que os rodeia.

Todo o processo começa no ponto semente. São analisados os pontos vizinhos deste, sendo incluídos ou não na imagem final, de acordo com o critério de proximidade tonal ao ponto semente. O critério de proximidade tonal baseia-se na comparação das componentes RGB do ponto a comparar com as do ponto semente. Pode ser tolerável uma variação percentual relativamente a cada componente, ou então, impôr explicitamente os limites de variação para cada

Fig. 8 - Janela de segmentação por *threshold*.

	R	G	B	%
Upper	255	255	255	51
Lower	114	95	49	

Fig. 9 - Janela de Segmentação por Crescimento de Regiões.

componente.

A Fig. 9 representa a janela de interacção com o utilizador. O número de *pixels* segmentados é visualizado, permitindo assim o cálculo de áreas relativas de objectos irregulares.

As Figuras 10 e 11 ilustram uma aplicação desta ferramenta, neste caso, a segmentação de objectos.

F. Transformadas de imagem.

Uma transformada de imagem é uma representação dessa imagem no domínio da frequência [3]. O cálculo de uma transformada é um processo lento, pelo que se geralmente se utilizam as *Fast Transforms*. A sua utilização, no entanto, não é genérica, requerendo uma área de imagem quadrada, cujo lado seja uma potência de 2. Foram

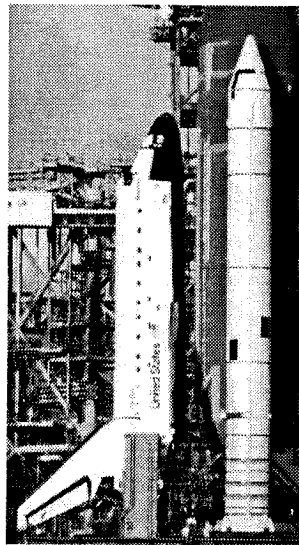


Fig. 10 - Imagem original.

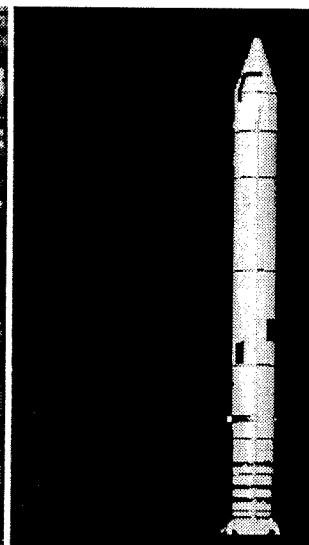


Fig. 11 - Imagem segmentada.

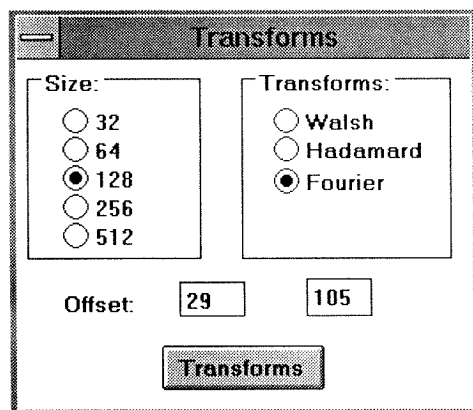


Fig. 12 - Janela das transformadas.

implementados os tamanhos de 32 a 512 pixels de lado do quadrado. As transformadas disponíveis são Transformada de Walsh, Hadamard e Fourier. A janela de interacção com o utilizador está ilustrada na Fig. 12.

A visualização da transformada calculada está a cargo duma classe de janelas (Fig. 13) que permite a visualização do módulo dos coeficientes, do módulo logarítmico, da fase e da transformada inversa. Permite também a execução de filtragem na frequência, através do desenho directo do filtro a aplicar.

Existe a possibilidade de gravação em ficheiro da transformada bem como um posterior *Load* desse ficheiro.

G. Filtragem Espacial.

A filtragem espacial é um tipo de filtragem que se baseia em operações no domínio espacial. Cada pixel da imagem filtrada é função, não só do pixel correspondente na imagem original, mas também dos seus vizinhos.

A lei de transformação, e portanto a topologia do filtro, é estabelecida na máscara de filtragem [1], que define a maneira como vão contribuir os pixels abrangidos no cálculo do pixel filtrado.

Os filtros podem conter uma ou várias máscaras. No caso de várias máscaras é feita uma filtragem aditiva, em que o nível de cinzento de cada pixel é o resultado da soma das várias filtrações (com as várias máscaras).

Devido à grande quantidade de cálculos necessários,

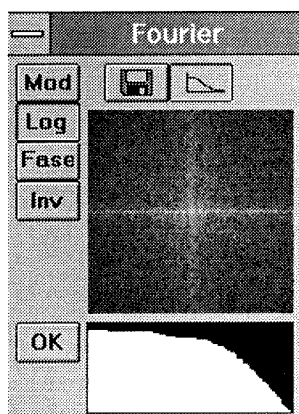


Fig. 13 - Janela de visualização das transformadas.

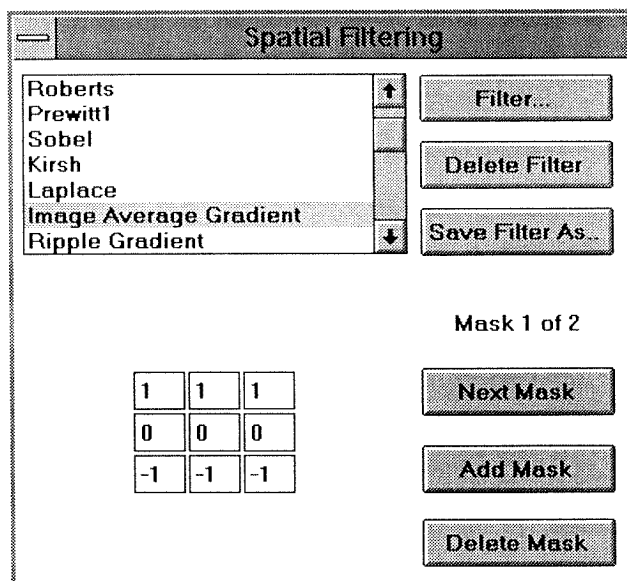


Fig. 14 - Janela da filtragem espacial.

principalmente quando os filtros são bastante elaborados, decidimos implementar as funções de baixo nível em *assembler*, tirando partido das instruções de 32 bits. De facto, os compiladores usuais geram apenas código de 16 bits, não aproveitando assim as capacidades dos processadores 386 e superiores (486, *pentium*, ...).

O código *assembler* de 32 bits, em conjunção com um algoritmo optimizado, permitiram alcançar tempos de filtragem muito curtos. A título de curiosidade, a velocidade de filtragem obtida numa máquina 486 a 33 MHz foi de 146000 pixels/s por elemento de máscara.

A Fig. 14 representa a janela de interacção com o utilizador. A Fig. 16 representa a imagem filtrada da Fig. 15, com o filtro "Image Average Gradient".

H. Reconhecimento automático de caracteres.

O reconhecimento automático de caracteres identifica o texto contido numa imagem através da comparação com

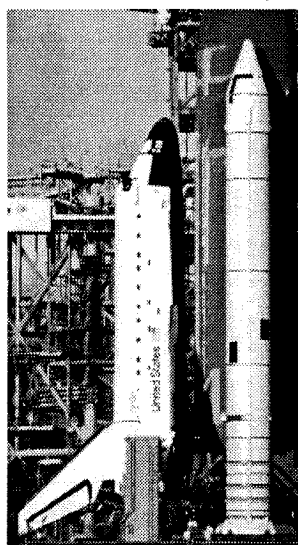


Fig. 15 - Imagem original.

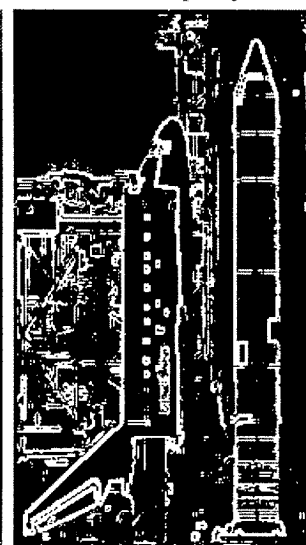


Fig. 16 - Imagem filtrada.

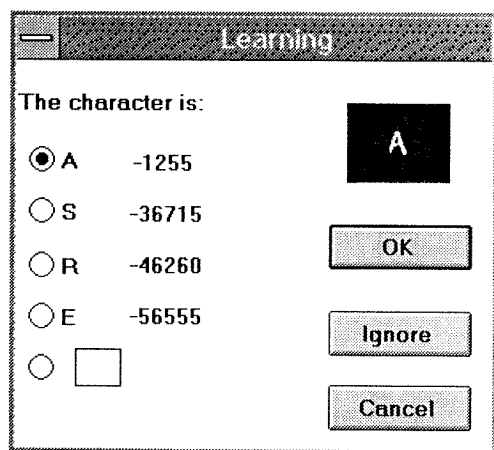


Fig. 17 - Janela da aprendizagem de caracteres.

padrões de caracteres previamente carregados em memória. A imagem pode ter sido obtida, por exemplo, através de um *scanner*.

O sistema requer uma aprendizagem prévia para definir os padrões dos caracteres. A aprendizagem é feita interactivamente, ou seja, o programa tenta reconhecer cada carácter e o utilizador deve corrigir ou confirmar a hipótese calculada pelo programa. Obviamente, quanto mais apurada for a fase de aprendizagem, maiores serão as probabilidades de sucesso no reconhecimento de um determinado texto. A estrutura do padrão de cada carácter consiste numa imagem normalizada resultante de uma média pesada das imagens normalizadas do carácter, possivelmente escrito de várias formas diferentes. A formação dos padrões é, assim, um processo de formação de uma imagem padrão de cada carácter.

É possível gravar os padrões para uso futuro e também continuar a aprendizagem em padrões previamente gravados.

Para iniciar o processo de reconhecimento é então necessário ter carregados em memória padrões que devem ser mais ou menos compatíveis com o tipo de caracteres do texto a reconhecer.

O texto, depois de reconhecido, é visualizado numa janela de texto onde pode ser editado, para correcção de eventuais erros.

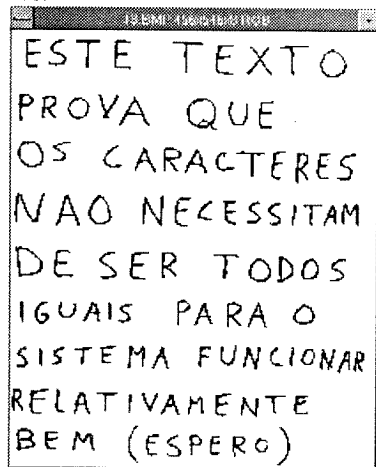
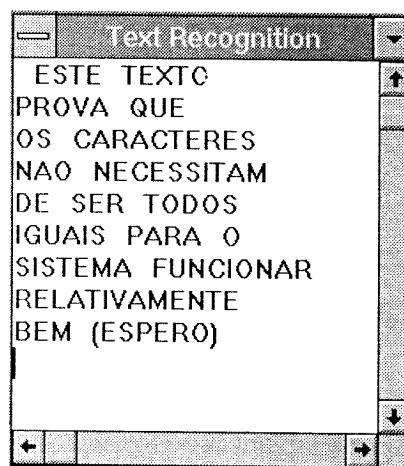
Fig. 18 - Imagem adquirida com *scanner*.

Fig. 19 - Texto reconhecido pelo programa.

A Fig. 17 ilustra a janela de interacção com o utilizador na fase de aprendizagem. A Fig. 18 representa uma imagem de um texto obtida através de um *scanner*. A Fig. 19 mostra esse texto depois de reconhecido pelo programa.

V - CONCLUSÕES.

Foi implementada a estrutura base de um sistema genérico de processamento digital de imagem, bem como algumas ferramentas. Os formatos de imagem reconhecidos pelo programa são o BMP, GIF [4] e IMG. A estrutura modular do sistema permite a fácil introdução de novos formatos de imagem, bem como novas ferramentas de processamento e/ou controle. A interacção com o utilizador é fácil e intuitiva. Um utilizador com experiência no ambiente WINDOWS™ não necessita de aprendizagem do sistema, pois são seguidos os princípios comuns à maior parte das aplicações desse ambiente.

Foram utilizados algoritmos e técnicas, que permitiram obter performances assinaláveis, nomeadamente no caso da filtragem espacial. A utilização de animação dinâmica da palette em algumas ferramentas, proporciona uma "afinação" da imagem em tempo real, ao contrário da técnica de *preview* largamente utilizada nas aplicações comerciais mais conhecidas.

VI - REFERÊNCIAS.

- [1] Jorge Saro, Sandra Jardim, "IDIPP 2.0 Manual do Utilizador".
- [2] Microsoft Press, "WINDOWS Programmer's Reference Version 3".
- [3] A. Sousa Pereira, "Apontamentos de Processamento Digital de Imagem".
- [4] Compuserve, "Graphics Interchange Format".

Receptor de Baixo Custo e Elevada Linearidade Para Sistemas de Televisão por Cabo Óptico até ao Assinante (FTTH)

Paulo Grilo Domingues, José Carlos Pedro

Resumo- É apresentado um circuito para utilização em receptores de Televisão por Cabo Óptico até ao assinante (FTTH). A arquitectura é baseada em dispositivos GaAs FET de baixo custo e propõe um novo esquema para redução da distorção não linear de 2ª ordem produzida pelos componentes activos.

Abstract- A receiver circuit for use in Fiber to the Home (FTTH) Cable TV Systems is presented. The design is based on low cost small signal GaAs FET devices and proposes a new scheme for the reduction of the 2nd order non-linear distortion generated by the active devices.

I. INTRODUÇÃO

Desde o seu início, a indústria de Televisão por Cabo (CATV) utilizou Cabo Coaxial para a distribuição de televisão em AM-VSB aos utilizadores domésticos. O surgimento das fibras ópticas trouxe a esta aplicação um meio de transmissão com notáveis características, se comparadas com o antigo cabo coaxial: muito baixa atenuação e elevada largura de banda, entre os mais importantes. No entanto, a instalação de fibras ópticas em redes de televisão por cabo tem-se restringido a ligações entre pontos de distribuição principais, devido ao elevado custo e limitado desempenho dos conversores óptico-electrico e electrico-óptico, necessários aos blocos de amplificação. Também devido a estas limitações, a utilização de ligações ópticas nas redes de televisão por cabo tem passado pelo recurso a dispendiosos circuitos de conversão de formato (AM-FM-AM) [1].

Avanços recentes em amplificadores puramente ópticos e a melhoria das características do laser e do fotodíodo tornaram possível a instalação de sistemas de distribuição puramente ópticos, trabalhando no formato de modulação AM-VSB. Em tais sistemas, as unidades não partilhadas pelos utilizadores têm um peso importante no custo total do sistema. É este o caso do receptor. O circuito aqui proposto é uma arquitectura de baixo custo usando um esquema inovador para a redução da distorção de 2ª ordem gerada pelos dispositivos activos, minimizando ao mesmo tempo a distorção de 3ª ordem.

II. REQUISITOS DO RECEPTOR E TOPOLOGIA PROPOSTA

Os requisitos tomados como referência para este trabalho são os estabelecidos pelo projecto ACCESS [2], especificados para o respectivo *link* óptico [3]. As especificações são particularmente exigentes no que diz respeito à distorção, devido à necessidade de manter a distorção produzida pelo receptor bem abaixo daquela que é produzida pelo Laser (o dispositivo menos linear do sistema). Os mais importantes valores estabelecidos são,

- Potência óptica de entrada: $-7 \dots 0$ dBm
- Nível de saída: $67 \dots 80$ dB μ V/portadora
- Índice de modulação óptico (OMI): 4.5%
- Número de canais: 40 (plano CENELEC)
- Nível de ruído equivalente na entrada: ≤ 11 pA/ $\sqrt{\text{Hz}}$
- CSO (*Composite Second Order*): < -70 dBc, 2 tons, OMI=35%/portadora
- CTB (*Composite Triple Beat*): < -74 dBc, 2 tons, OMI=35%/portadora
- Gama de frequências: 47 - 862 MHz

A seguinte topologia é proposta para o receptor: andar de entrada, baseado numa configuração de transimpedância; andar de saída baseado num circuito linearizado para baixa distorção. O andar de entrada é um esquema bem conhecido na área dos receptores ópticos, proporcionando baixo ruído sem necessidade de equalização. A utilização de realimentação contribui também para reduzir a distorção do FET. O andar de saída é um novo esquema de pré-distorção para redução da distorção de intermodulação, que será descrito na Secção seguinte.

III. ANDAR DE SAÍDA

Dado que o objectivo deste trabalho é o projecto de um receptor de baixo custo, a utilização de dispositivos baratos é uma condição de partida. A escolha de dispositivos GaAs FET é justificada pelo seu atraente comportamento não linear, sobretudo se comparado com o dos dispositivos bipolares. Contudo, os exigentes requisitos de distorção, sobretudo no que respeita à distorção de 2ª ordem, não podem ser satisfeitos pelas configurações de amplificação habituais. É certo que há esquemas de compensação bem conhecidos e referidos na

literatura, [4, 5], mas são em geral estruturas complexas, resultando em circuitos dispendiosos. Ainda assim, a configuração aqui apresentada é um caso especial do esquema geral de compensação por pré/pós-distorção, apresentado em [5]. O aspecto inovador consiste no facto das mais importantes fontes de distorção não linear de 2ª ordem produzidas pelo dispositivo GaAs FET, serem compensadas pela introdução em cascata de um novo GaAs FET mais uma malha passiva. A distorção de 3ª ordem é minimizada por uma escolha adequada das condições de polarização [6, 7].

A Figura 1 representa a topologia proposta. O comportamento não linear de cada bloco baseado num dispositivo activo pode ser caracterizado recorrendo a uma análise segundo Séries de Volterra para não-linearidades fracas [8]. De acordo com este método, a relação entre componentes de entrada e produtos de intermodulação ou harmónicas geradas, pode ser caracterizada por uma função de transferência de 2ª ordem, ou Operador de Volterra. Se a variável de entrada do bloco não linear for uma tensão constituída por um somatório de Q componentes,

$$v_s = \frac{1}{2} \sum_{q=-Q}^Q V_{sq} e^{j\omega_q t} \quad (1)$$

as componentes de tensão de 2ª ordem na saída serão dadas por,

$$v_{o2} = \frac{1}{4} \sum_{q1=-Q}^Q \sum_{q2=-Q}^Q V_{sq1} V_{sq2} H_2(\omega_{q1}, \omega_{q2}) e^{j(\omega_{q1} + \omega_{q2})t} \quad (2)$$

em que $H_2(\omega_{q1}, \omega_{q2})$ é o Operador de Volterra de 2ª ordem.

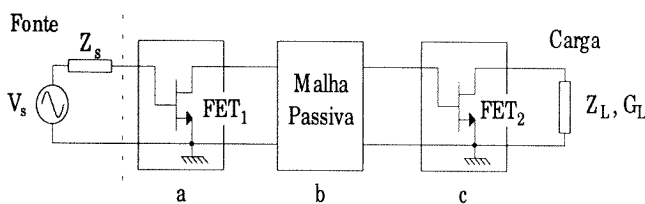


Fig. 1 - Configuração do andar de saída

A função de transferência de 1ª ordem (ganho linear) é definida da forma habitual, sendo representada por $H_1(\omega)$. Voltando à Figura 1 e relacionando cada bloco com o respectivo índice na função, as funções linear e de 2ª ordem globais são dadas por,

$$H_{1g}(\omega) = H_{1a}(\omega) \cdot H_{1b}(\omega) \cdot H_{1c}(\omega) \quad (3)$$

$$\begin{aligned} H_{2g}(\omega_1, \omega_2) = & H_{1a}(\omega_1) \cdot H_{1a}(\omega_2) \cdot H_{1b}(\omega_1) \cdot H_{1b}(\omega_2) \cdot H_{2a}(\omega_1, \omega_2) + \\ & + H_{2a}(\omega_1, \omega_2) \cdot H_{1b}(\omega_1 + \omega_2) \cdot H_{1c}(\omega_1 + \omega_2) \end{aligned} \quad (4)$$

Uma análise mais completa necessita de uma descrição mais detalhada de cada função, mas pode já assinalar-se que os termos em (4) apresentam sinais opostos. Isto resulta de H_{1c} apresentar uma inversão de fase, pelo menos para frequências baixas. Contudo, a escolha de dispositivos tipicamente utilizados na banda de micro-ondas torna-os praticamente não reactivos na banda de CATV, pelo que daí não resultam problemas graves. Por outro lado, os comprimentos electricos devidos às dimensões físicas dos componentes da malha passiva podem causar um impacto considerável no sistema de compensação. Este problema será estudado mais à frente. Os blocos activos não necessitam de ser semelhantes para ser possível produzir compensação e pode até ser demonstrado que certas diferenças podem ser aproveitadas de forma vantajosa. Deste modo, não constituindo restrições graves os pressupostos de independência com a frequência e semelhança entre dispositivos, pode-se escrever,

$$H_{1a} = H_{1c}, \quad H_{2a} = H_{2c} \quad (5)$$

operando-se a seguinte simplificação,

$$H_{2g} = H_{1a} H_{1b} H_{2a} (H_{1a} H_{1b} + 1) \quad (6)$$

Dado H_{1a} ser negativa, verifica-se de imediato que a condição $|H_{1a}| |H_{1b}| = 1$ anula a função de 2ª ordem. Dado que assumimos independência com a frequência, as funções podem ser consideradas constantes com este parâmetro, deste modo a condição anterior, escrita na forma

$$|H_{1b}| = 1/|H_{1a}| \quad (7)$$

permite concluir que a malha passiva deve apresentar uma perda (em tensão) numericamente igual ao ganho de apenas um FET. A função de transferência linear virá então $H_{1g} = H_{1a}$, i.e., o ganho do circuito reduz-se ao ganho de apenas um FET. Pode portanto concluir-se que o preço a pagar pela linearização do amplificador é a perda de ganho. Em resumo, este esquema troca o ganho de um andar pela linearidade da cadeia de amplificação.

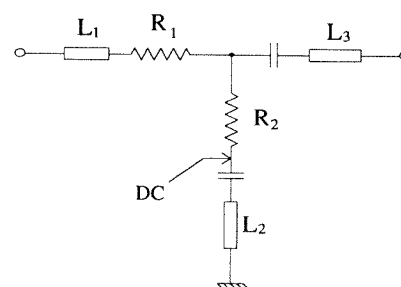


Fig. 2 - Malha passiva

A malha passiva que implementa a função acima descrita é simplesmente um divisor resistivo, representado na Figura 2 com os respectivos comprimentos electricos associados (L_1 , L_2 , L_3).

A selecção dos dispositivos apropriados é assunto da maior importância, bem como o conhecimento das suas características não lineares. (7) sugere que não é necessária informação sobre comportamentos não lineares para a determinação da malha passiva, mas de facto isso só seria verdade se ambos os FET's fossem idênticos. Dado que este não é normalmente o caso, (7) deixa de ser válida. Além disso, só determinadas regiões de polarização são apropriadas. Este aspecto foi já abordado por outros autores [8], [9]. Esta última referência é de especial importância pois apresenta um método permitindo a caracterização completa dos efeitos não lineares de baixa frequência do GaAs FET, e investiga o seu comportamento ao longo de um intervalo de polarização da tensão VGS (gate-source). Este conhecimento é da maior importância, pois é possível, pelo menos para certos dispositivos, encontrar uma zona de polarização em que as componentes de distorção de 2ª ordem atingem um mínimo, enquanto que as componentes de 3ª ordem desaparecem, ou se tornam bastante pequenas. Esta região, identificada em [9] como ocorrendo próximo de VGS = 0V é claramente vantajosa para esta aplicação. Os dispositivos NEC NE710084 satisfazem esses requisitos e foram escolhidos para o protótipo. A caracterização das unidades passa pela caracterização do seu comportamento não linear e o cálculo de um conjunto de coeficientes que descrevem completamente a corrente ids (drain-source) em função das tensões de controlo vgs e vds. A relação é dada pela seguinte expressão,

$$\begin{aligned} i_{ds}(v_{gs}, v_{ds}) = & G_m \cdot v_{gs} + G_{ds} \cdot v_{ds} + G_{m2} \cdot v_{gs}^2 + \\ & + G_{md} \cdot v_{gs} \cdot v_{ds} + G_{d2} \cdot v_{ds}^2 + G_{m3} \cdot v_{gs}^3 + \\ & + G_{m2d} \cdot v_{gs}^2 \cdot v_{ds} + G_{md2} \cdot v_{gs} \cdot v_{ds}^2 + G_{d3} \cdot v_{ds}^3 \end{aligned} \quad (8)$$

No caso em que os dois FET's são idênticos e os seus elementos parasitas ignorados, é possível encontrar um conjunto de relações simples que determinam a situação de compensação das componentes de 2ª ordem. Assumindo,

$$\begin{aligned} G_x &= 1/(R_1 + R_2) & GTX &= G_x + G_{ds} \\ GTL &= G_{ds} + G_L \end{aligned} \quad (9)$$

são elas,

$$K \equiv R_2 G_x = GTX/G_m \text{ e } GTL = GTX \quad (10)$$

No entanto, a presença de elementos parasitas, bem como as diferenças entre as duas unidades, tornam o cálculo mais complicado, pelo que uma solução numérica optimizada que usa (10) como condição de partida é mais apropriada. Para o protótipo, foram calculados os

seguintes valores para R_1 e R_2 : $R_1 = 27\Omega$ e $R_2 = 23\Omega$ (tendo-se partido de $R_1 = 31\Omega$ and $R_2 = 19\Omega$).

São apresentados resultados experimentais e de simulação mostrando o comportamento do circuito proposto, excitado por uma única tonalidade (Figura 3). Visualizam-se duas situações. Dado que a montagem experimental teve que utilizar componentes de montagem superficial (*surface mount*) discretos, não puderam ser evitadas dimensões físicas significativas (L_1 , L_2 e L_3), introduzindo estas uma degradação no mecanismo de compensação com o aumento da frequência. No entanto, se estes elementos puderem ser evitados, ou consideravelmente reduzidos - tal como aconteceria no caso de uma implementação MMIC (*Monolithic Microwave Integrated Circuits*) - o mecanismo de compensação funcionaria quase perfeitamente ao longo da banda de CATV. As curvas desenhadas correspondem a um caso simulado/experimental com $L_1=7.2\text{mm}$, $L_2=7.0\text{mm}$, $L_3=6.8\text{mm}$, e um caso simulado com $L_1=L_2=L_3=0\text{mm}$. O "eixo" de compensação refere-se à melhoria em relação ao nível de distorção harmónica produzido por um FET sozinho (a $2f_1$), admitindo que a potência da fundamental (a f_1) é a mesma nos dois circuitos. Devido à relativamente baixa carga do circuito - 50Ω - o ganho de tensão não é elevado: 6.8dB. Num sistema de 75Ω , em que um receptor de CATV opera, o ganho poderia ser aumentado para 10 dB.

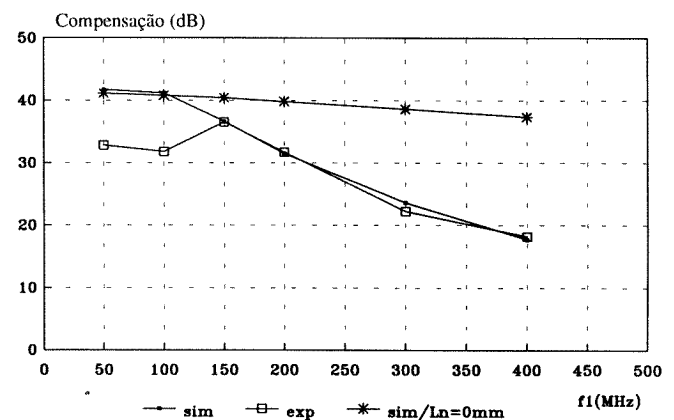


Fig. 3 - Resultados experimentais e simulados do andar de saída

IV. ANDAR DE ENTRADA

O andar de entrada é baseado numa topologia de transimpedância com realimentação e está representado na Figura 4.

Numa perspectiva de minimização do ruído, a utilização de uma configuração de transimpedância é quase imperativa, já que é a maneira mais fácil de conseguir as especificações de ruído. Alternativamente, um receptor de alta impedância com equalização da resposta em frequência teria de ser utilizado. Na configuração de transimpedância proposta, o ruído da resistência de realimentação é dominante em toda a banda, tornando

simples o cálculo da corrente equivalente de ruído à entrada do amplificador. Tal foi também realizado por simulação, incluindo outras fontes de ruído (do FET, concretamente), sendo confirmado que a corrente equivalente de ruído na entrada está abaixo de $11\text{pA}/\sqrt{\text{Hz}}$ com $R_f = 430\Omega$.

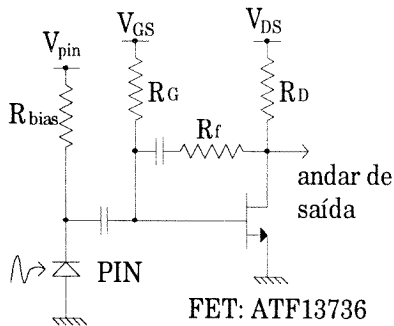


Fig. 4 - Andar de entrada

Do ponto de vista da distorção a configuração de transimpedância tem também um importante papel. De facto, uma vez que o andar de saída é de baixa distorção, o andar de entrada pode ainda gerar contribuições apreciáveis, se não se tomarem os devidos cuidados. Para este andar foi escolhido um FET Avantek de baixo custo (ATF 13736), com um valor de G_{m2} cerca de três vezes inferior ao do NEC. Por outro lado, G_{d2} é agora mais elevado do que no NEC. Isto significa que, enquanto no andar de saída a contribuição da tensão v_{ds} para a distorção era desprezável, pode agora ser dominante. E verificou-se que assim era, de facto, se ganhos de tensão demasiado elevados fossem impostos. No entanto, dado que a transimpedância do andar é determinada sobretudo por R_f , permanecendo aproximadamente independente do ganho de tensão (admitindo que o ganho da malha, controlado por R_D , é muito maior do que a unidade), este pode ser alterado com alguma liberdade, de maneira a controlar a distorção gerada por G_{d2} . Portanto, R_D foi diminuída para um valor que torna a distorção gerada pelo

andar de entrada mais baixa do que a gerada pelo andar de saída. Isto foi conseguido com $R_D = 75\Omega$. O andar descrito melhora a distorção de 2ª ordem gerada pelo FET sozinho em cerca de 20dB. O fotodiodo utilizado foi um modelo de baixa distorção EPM 700 L da Epitaxx.

V. AVALIAÇÃO EXPERIMENTAL

O circuito descrito nos capítulos precedentes foi construído e testado. O teste baseou-se num *set-up* segundo o esquema da Figura 5.

A utilização de duas unidades laser seguidas de um acoplador óptico destina-se a ultrapassar o problema da elevada distorção harmónica produzida por aqueles dispositivos, seguramente mais elevada do que a produzida pelo receptor a testar. Assim, os lasers são modulados por sinais sinusoidais com frequências, respectivamente, f_1 e f_2 , gerando harmónicas sobre cada uma delas. O receptor recebe um sinal composto, gerando, para além de harmónicas, produtos de intermodulação (frequências f_2-f_1 , f_2+f_1 , $2f_2-f_1$, etc). Como estas são unicamente devidas ao receptor, podem ser utilizadas para avaliar o seu nível de distorção.

Foram também realizadas simulações com 2 tons, no domínio da frequência [10] e no domínio do tempo [11], para o circuito global, cujos resultados são apresentados juntamente com os resultados experimentais. Como foi referido, o circuito experimental foi realizado com componentes de montagem de superfície normais, não se podendo evitar comprimentos eléctricos que degradam a compensação não-linear na parte superior da banda de CATV. Também a medida de produtos de intermodulação devido a sinais de entrada acima de 400 MHz (na oitava superior da banda) se torna agora relevante, pois que os produtos diferença ($f_2 - f_1$) podem cair dentro de partes úteis da banda. Isto significa que o circuito se deve comportar de forma aceitável num espectro de frequências mais largo do que aquele que foi considerado

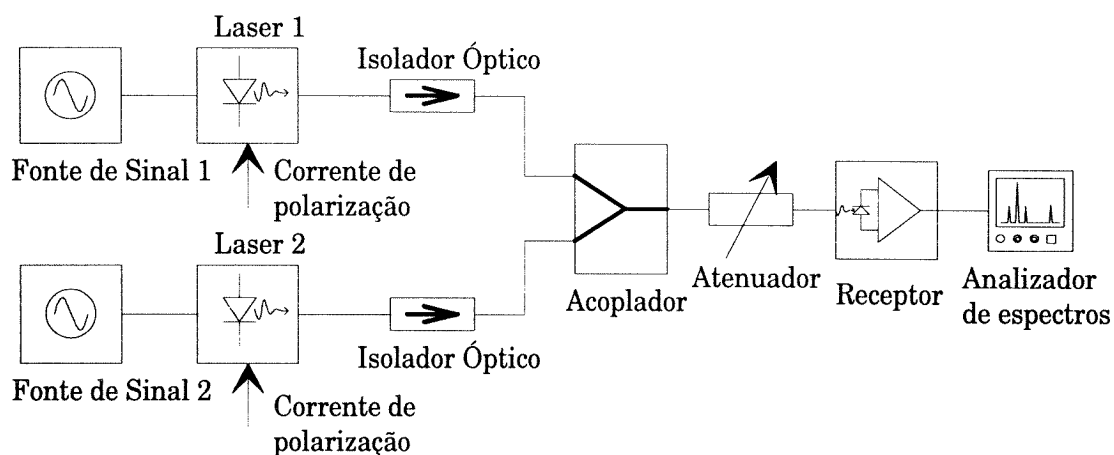


Fig. 5 - Set-up experimental

na avaliação harmónica. Os resultados, apresentados na Figura 6, mostram CSO ($P_{o_{f2+f1}}/P_{o_{f1}}$) à máxima potência de saída requerida (o ganho do receptor resultou mais elevado do que especificado, mas isso seria fácil de alterar, se necessário). Como anteriormente, duas situações simuladas são apresentadas: (1) - incluindo os comprimentos físicos experimentais, (2) - não os incluindo. Os resultados experimentais tendem a exceder as especificações, particularmente na parte superior da banda, mas o acordo com as simulações é bastante bom. É portanto razoável esperar que um protótipo baseado em micro-componentes estaria também de acordo com as simulações, satisfazendo confortavelmente as especificações.

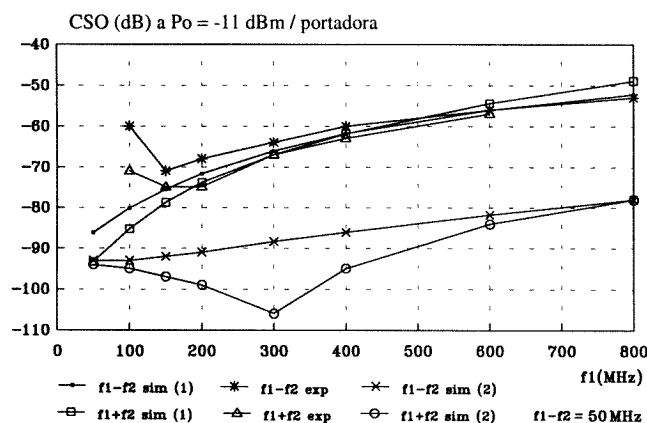


Fig. 6 - Resultados experimentais e de simulação do receptor

Faz-se notar que dispositivos GaAs FET com menor Gm_2 do que os utilizados, certamente facilitariam a tarefa aqui proposta. Mas é também importante que aquele parâmetro apresente um mínimo numa gama de tensões de polarização VGS utilizável. Isto não acontecia no caso do FET utilizado no andar de entrada, apesar de ele apresentar de facto um menor Gm_2 . A utilização do FET num mínimo de Gm_2 apresenta a dupla vantagem de minimizar a sensibilidade do circuito às condições de polarização, facilitando o ajuste e garantindo estabilidade a longo prazo, e de minimizar a distorção de 3ª ordem, de acordo com [9]. Os resultados de CTB não são apresentados, pois que se constatou estarem abaixo das capacidades de medida; de qualquer forma, bastante abaixo dos valores requeridos pelas especificações.

VI. CONCLUSÕES

Neste artigo apresentou-se uma topologia original de pré-distorção em cascata para redução dos produtos de distorção não linear de 2ª ordem gerados numa cadeia de banda larga e alto ganho. Demonstrou-se ainda a sua utilidade na arquitetura de receptores de elevada linearidade para sistemas FTTH de CATV, pela construção e teste de um protótipo laboratorial. Resultou assim ser uma configuração simultaneamente robusta e simples, mas eficiente, capaz de produzir níveis de

compensação da distorção de 2ª ordem de pelo menos 30dB em boas condições experimentais.

AGRADECIMENTOS

Os autores agradecem o apoio proporcionado pelo Prof. Dr. Oliveira Duarte do Grupo de Sistemas de Banda Larga. O trabalho apresentado foi desenvolvido no âmbito de uma Bolsa atribuída pela JNICT e contribuiu para o projecto ACCESS do programa RACE.

REFERÊNCIAS

- [1] James A. Chiddix, Jay A. Vaughan and Ronald W. Wolfe, "The Use of Fiber Optics in Cable Communications Networks", *Journal of Lightwave Technology*, vol. 11, nº 1, January 1993.
- [2] RACE (Research for Advanced Communications in Europe): Programa de Investigação da Comunidade Europeia. ACCESS: projecto RACE R.1030
- [3] Lorens Almehag, "- Specification of the TV link - Specification of the transmitter and receiver", 1991, RACE ACCESS (R.1030) document ERI1000 WC2T1F D04C.
- [4] Robert G. Meyer, et. al., "A Wide-Band Feedforward Amplifier", *IEEE Journal of Solid State Circuits*, vol SC-9, No 6, December 1974
- [5] A. Prochazca, P. Lancaster and R. Neumann, "Amplifier Linearization by Complementary Pre- or Post-Distortion", *IEEE Transactions on Cable Television*, vol. CATV-1, No.1, October 1976.
- [6] P. Domingues and J. C. Pedro, "A New Pre-Distortion Scheme for the Reduction of Intermodulation Distortion in Optical SCM TV Systems", *Proceedings of the 23rd Microwave Conference*, September, 1993
- [7] P. Domingues and J. C. Pedro, "A New Pre-Distortion Scheme for the Reduction of Intermodulation Distortion in Optical Cable TV SCM Receivers", *Microwave and Optical Technology Letters*, November, 1994
- [8] Stephen A. Maas, "Non-linear Microwave Circuits", Artech House, 1988.
- [9] J. C. Pedro and J. Perez, "Complete Characterisation of MESFET Ids Nonlinearities for Intermodulation Analysis", 1992, *Microwave and Optical Tech. Let.*, vol. 5, No. 8, pp. 377-380.
- [10] C/NL, Linear & Non-linear Microwave Circuit Analysis & Optimization Software, S. A. Maas, Artech House, 1990
- [11] Pspice: Circuit Analysis, MicroSim Corporation, 1991

Impact of Non-zero Extinction Ratio on Optically Pre-amplified Receivers

L. F. Botelho Ribeiro*, J. R. Ferreira da Rocha*, J. Lemos Pinto**

DETUA*, DFUA**

Resumo- Em sistemas práticos de comunicações ópticas os lasers emissores são polarizados com uma dada corrente. Surge assim um patamar de potência óptica no receptor. O presente trabalho inclui este fenómeno numa formulação analítica de descrição estatística de um sistema de detecção directa com pre-amplificação óptica.

Abstract- In practical optical communication systems, the transmitter laser is polarised with a finite current. A given optical power plateau is, thus, observed at the receiver. This contribution includes this phenomenon in an analytical formulation for statistical description of a optically preamplified direct-detection system.

I. INTRODUCTION

Direct detection optical communication systems can be studied quite accurately and without excessive computing effort by analytical means. One widely used tool to this purpose is the moment generating function (MGF) of the electric current at the receiver decision circuit. This function contains all the relevant statistical information and is well adapted to various techniques for evaluation of many parameters: bit error rate (BER) bounds, BER approximations, optimum decision threshold, output current mean level (conditioned on the symbol) or standard deviation (due to noise) are some examples.

Results have been reported describing the MGF for different types of receiver, where various assumptions on the noise statistical properties were made. Personick considered an optical amplifier with additive optical noise and avalanche photodiode (APD) detection, followed by an integrate-and-dump filter [1]. Yamamoto [2] has obtained expressions for the current average and noise variances after equalisation exclusively by physical considerations and then used Gaussian approximation (GA) to assess receiver sensitivity.

Da Rocha [3] used the MGF to study receiver optimization in the presence of intersymbol interference for receivers without optical amplification. A study on improved alternative performance evaluation methods is

presented by O'Reilly [4], considering Chernoff bound (CB) and modified Chernoff bound(MCB) on the final BER.

Helstrom presented an alternative approximation for the system BER using the saddle-point approximation [5] which also requires the use of the MGF. This method gives lower BER than others but its accuracy depends on the specific receiver configuration. Fyath [6] has studied the importance of laser amplifiers for the sensitivity and power penalty of direct-detection receivers.

A formulation for the MGF of optically preamplified receivers was proposed by Fyath in [7]. A different approach was followed by Ribeiro [8] leading to a more rigorous expression for the MGF.

In this contribution we face the problem of input signals with non-zero extinction ratio. This is the case for most practical systems where lasers have bias current large enough to launch some power into the fiber even during the period of time corresponding to the symbol zero. This power, after attenuation in the fiber, optical pre-amplification, filtering and detection will be processed much in the same way as the pulses for the symbol one. The impact of non-zero extinction ratio on sensitivity, optimum decision threshold and optical power penalty is evaluated according to different methods: Gaussian approximation, Chernoff bound, modified Chernoff bound and saddlepoint approximation (SPA).

II. THEORY

Consider the receiver in Fig. 1. The optical pulse entering the optical amplifier is allowed to have non-zero power at symbol "0". Then, the optical power pulse $h_p(t)$ transmitted for symbol "1", is superimposed over P_0 .

We define the normalised input pulse as follows,

$$\frac{1}{T} \int_{-\infty}^{+\infty} h_{pn}(t) dt = 1 \quad (1)$$

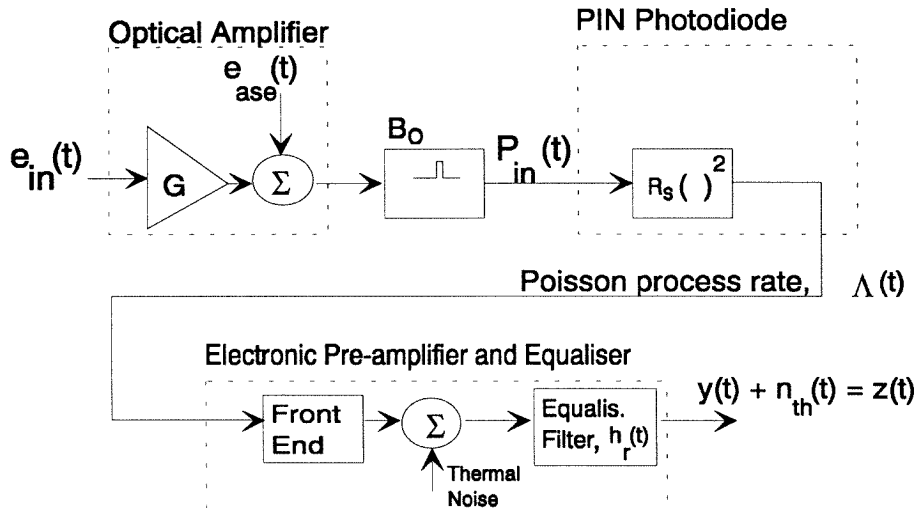


Fig. 1 - Optically pre-amplified receiver model

The actual power pulse to be used in the MGF formulation is related to the normalized pulse $h_{pn}(t)$, the average optical power S and the power extinction ratio ϵ according to,

$$h_p(t) = h_{pn}(t) \cdot 2S \frac{1-\epsilon}{1+\epsilon} \quad (2)$$

Where ϵ is the ratio of average powers on symbols "0" and "1", respectively, and the output pulse is normalised by $\max\{h_{out}(t)\}=1$. Finally we use an electronic pulse shaping filter providing an impulse response,

$$h_r(t) = \mathcal{F}^{-1} \left\{ \frac{H_{out}(\omega)}{H_{pn}(\omega)} \right\} \quad (3)$$

where $\mathcal{F}$ denotes Fourier transform and $H_{out}(\omega)$, $H_{pn}(\omega)$ are respectively the Fourier transforms of $h_{out}(t)$ and $h_{pn}(t)$. The average symbol conditioned powers are related to the extinction ratio and the average power by,

$$P_{0^n} = \frac{2\epsilon S}{1+\epsilon} \quad P_{1^n} = \frac{2S}{1+\epsilon} \quad (4)$$

We now make use of the MGF derived in [8] to obtain the new symbol conditioned MGF where the input optical power contribution is reformulated to cope with the new assumptions,

$$M_{Y_i}(s) = \exp \left\{ \int_{-\infty}^{\infty} \left[\frac{R G \left(e^{sqh_r(t-\tau)} - 1 \right) P_i(\tau)}{1 - R N_o \left(e^{sqh_r(t-\tau)} - 1 \right)} \right] d\tau + \frac{\sigma_{th}^2 \cdot s^2}{2} - \int_{-\infty}^{\infty} B_o \cdot \ln \left[1 - R N_o \left(e^{sqh_r(t-\tau)} - 1 \right) \right] d\tau \right\} \quad (5)$$

Where N_o is the unilateral Amplified Spontaneous Emission noise power spectral density of the optical preamplifier, σ_{th} is the receiver thermal noise standard deviation, G is the optical amplifier gain, B_o is the optical filter bandwidth and $R=\eta/(h\nu)$. η is PIN quantum efficiency, h is Planck's constant and ν is the optical carrier frequency. The input time-dependent power is expressed as follows,

$$P_i(\tau) = P_{0^n} + a_i \cdot h_p(\tau) \quad (6)$$

where, for transmitted symbol "0" or "1", a_i takes the values 0 and 1 respectively. Once we have obtained the symbol conditioned MGF, methods in [2],[4],[5] can be used to assess system MCB, CB, SPA and GA on the BER. Also, noise variance and optimum decision threshold can be calculated.

III. RESULTS

Throughout this work we have considered a Gaussian input pulse $h_{pn}(t) = \exp(-t^2/\alpha^2)/(\alpha\sqrt{2\pi})$ with $\alpha=0.1T$. The output pulse is a full-raised cosine. Consequently, the transfer function of the equalising filter was found by (3). For the analysis, we have considered $\sigma_{th} = 1.366 \times 10^{-5}$ A, $G=25.6$ dB and the ASE noise

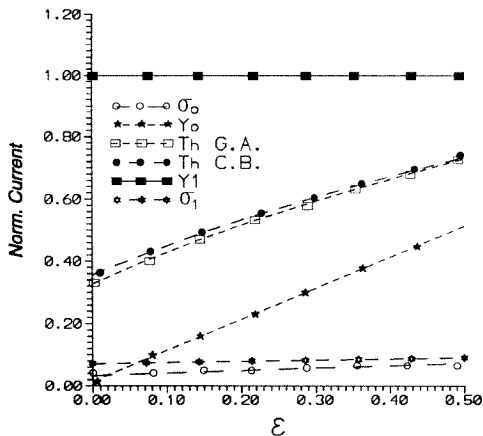


Fig. 2 - Normalised output current, optimum threshold and noise at sampling instant versus power extinction ratio ϵ .

density $N_0 = 9.46 \times 10^{-17}$ W/Hz. Parameter ϵ is varied from 0 to 0.5 keeping constant average power $S = -29$ dBm.

Output current for symbols "1" (Y_1) and "0" (Y_0) can be seen in Fig. 2 as well as optimum decision threshold T_h according to Gaussian approximation and Chernoff bound. σ_0 and σ_1 represent the total noise standard deviations for symbols "1" and "0" at the sampling instant. All results are normalized to the maximum output current.

As expected, the symbol conditioned average currents get closer as ϵ increases towards 0.5. Decision threshold estimates are quite near for both methods. Furthermore, the difference tends to decrease as ϵ increases. In this case, and for symbol "0", signal-dependent quantum noise dominates over spontaneous-spontaneous beat noise.

Looking at sensitivity results in Fig. 3, we conclude that significant performance degradation will arise for large extinction ratios. Although SPA method provides far more optimistic results, we observe approximately the same behaviour with ϵ for any method.

With the MCB method, the power penalty is found to be 7.1 dB, for an extinction ratio of 0.5.

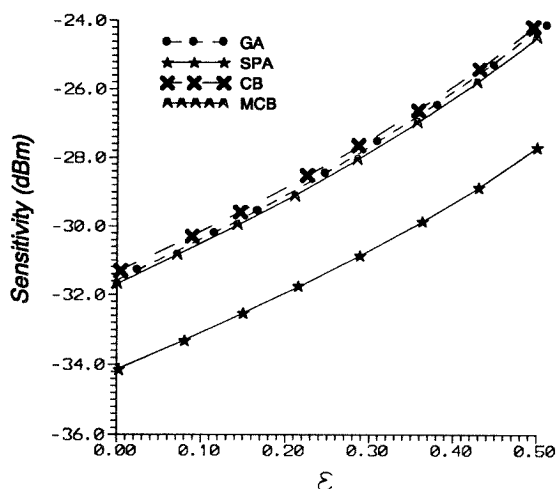


Fig. 3 - Sensitivity dependence on ϵ according to different methods.

IV. CONCLUSIONS

The power extinction ratio is a critical parameter in direct-detection communication systems. For high-speed operation, designers often choose to polarise the laser near/above threshold. If not sufficient energy is emitted for symbol "1", sensitivity degradation relative to a zero extinction-ratio system will become significant. With the new formulation, we have developed alternative analytical tools to quantify the expected degradation.

Future developments of the MGF formulation will include multiple cascaded amplifier repeaters and wavelength division multiplexing systems.

ACKNOWLEDGEMENT

This work has been supported by JNICT (Junta Nacional de Investigação Científica e Tecnológica) grant BD/619/IA.

REFERENCES

- [1] S. Personick, "Applications for Quantum Amplifiers in Simple Digital Optical Communication Systems," *Bell Syst. Tech. J.*, vol.52, No.1, pp. 117-133, January 1973
- [2] Y. Yamamoto, "Noise and Error Rate Performance of Semiconductor Laser Amplifiers in PCM-IM Optical Transmission Systems," *IEEE J. Quantum Electron.*, vol.QE-16, No.10, pp.1073-1081, October 1980
- [3] J.R.F. da Rocha and J. O'Reilly, "Linear Direct-Detection Fiber-Optic Receiver Optimization in the presence of Intersymbol Interference," *IEEE Transactions on Communications*, vol. COM-34, No.4, pp.365-374, April 1986
- [4] J.J. O'Reilly and J.R.F. da Rocha, "Improved Error Probability Evaluation Methods for Direct Detection Optical Communication Systems," *IEEE Transactions on Information Theory*, vol. IT-33, No.6, NOV/87, pp.839-848
- [5] C.H. Helstrom, "Calculating Error Probabilities for Intersymbol and Cochannel Interference," *IEEE Transactions on Communications*, vol. COM-34, No.5, May 1986, pp.430-435
- [6] R. Fyath, A. McDonald and J.J. O'Reilly, "Sensitivity and Power Penalty Considerations for Laser Preamplified Direct Detection Optical Receivers," *IEE proc. Part J*, vol.136, No.4, AUG/89, pp.238-248
- [7] R. Fyath and J.J. O'Reilly, "Comprehensive Moment Generating Function Characterisation of Optically Preamplified Receivers," *IEE proc. Part J*, vol.137, No.6, DEC/90, pp.391-396
- [8] L.F.B. Ribeiro, J.R.F. da Rocha and J.L. Pinto, "A New Statistical Formulation for Optically Preamplified Receivers," *BILCON'92 - Ankara-Turkey*, 27-28 July 1992, pp.109-115

Amplificador de Instrumentação de Sinais Biológicos

Rui Martins, Francisco Vaz

Resumo- Neste artigo é descrito parte de um sistema de aquisição de sinais EEG portátil, realizado sobre a forma de um único circuito integrado monolítico. Um muito baixo ruído e consumo são características essenciais num sistema deste tipo e as grandes metas a atingir.

Por ser o elemento mais crítico do sistema, especial atenção é dada ao amplificador de instrumentação (AI). Por ter provado ser a técnica mais adaptada a uma implementação CMOS, o AI foi desenhado usando realimentação em corrente.

O circuito foi fabricado numa tecnologia *standard CMOS 2.4- μm n-well, double poly, double metal*.

Abstract- This paper describes a monolithic EEG acquisition system. A very low noise and power consumption are essential in this systems, so they are the main targets to achieve.

Because the instrumentation amplifier (IA) is the most critical part of the system, particular attention is given to this element, where the use of current feedback techniques proved to be the best solution.

The circuit was fabricated in a standard 2.4- μm n-well, double poly, double metal technology.

I. INTRODUÇÃO

A aplicação de técnicas VLSI no campo da instrumentação médica, criou novas perspectivas de integração destes sistemas, tendo como consequência aberto o caminho da sua miniaturização e o aparecimento de sistemas portáteis.

É aqui descrita a parte de um sistema de telemetria de sinais biológicos (dedicado especialmente ao EEG - electroencefalograma, mas facilmente adaptável a outros sinais) correspondente à aquisição e condicionamento do sinal, realizada num único circuito integrado monolítico, com características que permitem fazer parte de uma unidade portátil. Esta imposição obriga a que se procure reduzir ao máximo o número de componentes exteriores (resistências, condensadores, circuitos integrados, ...), assim como manter os níveis de consumo o mais baixos possíveis, para que se garanta uma autonomia suficiente, sem aumentar

demasiadamente o peso do equipamento, muito dependente dos acumuladores ou pilhas usados.

O diagrama de blocos do sistema desenvolvido é apresentado na Fig. 1. Ele inclui 16 amplificadores de instrumentação (AI), um multiplexador 16:1, um amplificador de ganho programável, circuitos para efectuar calibração por *software* incluindo um oscilador estabilizado em tensão e frequência, o *interface* digital com o exterior e fonte de corrente/tensão de referência para polarização dos diversos circuitos.

O circuito foi realizado com base na tecnologia *CMOS MIETEC 2.4 μm , double poly - double metal*. Como ferramentas de CAD foi usado o framework *EDGE / CADENCE*. Os circuitos foram simulados usando *HSPICE / Meta-Software*.

Como características principais temos:

- Número de canais = 16
- Consumo de corrente total < 500 μA
- Ruído total ref. à entrada do AI < 1.5 μV (RMS)
- CMRR ≥ 100 dB
- $0.3 < \text{Largura de Banda} < 150$ Hz
- Elevada impedância de entrada: >1 M Ω
- Freq. de amostragem = 500 amostras/(s-canal)⁻¹
- Ganho= 500, 1000, 2000 e 5000 (programável)

II. AMPLIFICADORES DE INSTRUMENTAÇÃO

Os sinais de EEG têm níveis muito baixos e largura de

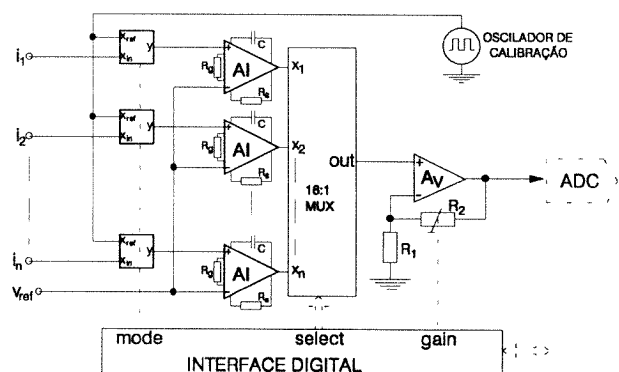


Fig. 1. Diagrama de blocos do sistema de aquisição de sinais EEG.

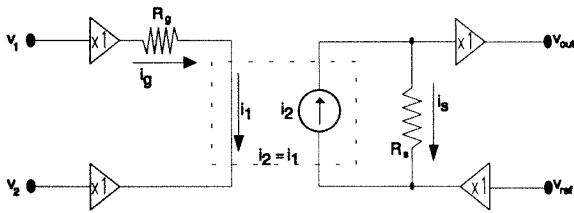


Fig. 2. Diagrama de blocos do AI com realimentação em corrente.

banda que inclui baixas frequências (0.3 - 150Hz) [1]. É assim necessário a existência nestes sistemas, de amplificadores com elevado ganho e muito baixo ruído (incluindo ruído *flicker*). Outro pormenor essencial é o elevado CMRR que precisam de possuir. Amplificadores de instrumentação com estas características são normalmente implementados em tecnologia bipolar ou JFET, devido ao mais elevado ganho e menor ruído inerente destas tecnologia, quando comparada com a tecnologia CMOS. É assim essencial aplicar novas configurações e efectuar um projecto *full-custom* para superar os problemas apresentados por esta tecnologia. É esta também a razão porque é dada especial atenção ao desenho e projecto de amplificadores de instrumentação.

A. AI com realimentação em corrente

O uso de amplificadores de instrumentação com a convencional realimentação resistiva, como a clássica topologia com 3 amplificadores operacionais, mostrou-se pouco eficaz quando o misto de baixo consumo e elevado CMRR é essencial. De facto estas estruturas necessitam amplificadores operacionais que gastam muita corrente, para serem capazes de fazer o *drive* das resistências de realimentação, que por sua vez precisam de ser muito bem emparelhadas (difícilmente realizável em tecnologia CMOS) com vista à optimização do CMRR.

Para ultrapassar estes problemas foram usados amplificadores com realimentação em corrente [2]-[6]. A Fig. 2 apresenta um diagrama genérico em que este tipo de amplificadores se baseia.

Neste tipo de amplificadores o sinal de entrada é aplicado a dois *buffers* (pelo que resulta numa impedância de entrada muito elevada) de ganho de tensão unitário. Assim é gerada uma corrente na resistência R_g

proporcional à diferença do sinal aplicado entre v_1 e v_2 e cuja constante de proporcionalidade é $1/R_g$ (o circuito de entrada funciona como um amplificador de transcondutância). Esta corrente é reproduzida fielmente por intermédio de um espelho de corrente para o circuito de saída, onde é forçada a circular na resistência R_s . A tensão nesta resistência é transmitida à saída por outro *buffer* de ganho unitário (o circuito de saída comporta-se como um amplificador de transresistência).

É interessante notar que um amplificador de instrumentação deste tipo, em contraste com os convencionais, é desenhado para funcionar em malha aberta, ou pelo menos sem que exista uma realimentação global. Este facto associado à existência de um único ponto de alta impedância (aquele que antecede imediatamente o *buffer* de saída), torna a resposta em frequência deste tipo de circuitos extremamente eficaz.

Fazendo uma análise quantitativa à figura 2 temos que: $i_g = (v_1 - v_2)/R_g$. Por seu lado na malha de saída: $v_{out} = i_s \cdot R_s + v_{ref}$. Como: $i_1 = i_2 = i_g = i_s$ obtém-se:

$$v_{out} = \frac{R_s}{R_g} \cdot (v_1 - v_2) + v_{ref} \quad (1)$$

De (1) resulta que o ganho é definido apenas pelo quociente de duas resistências (R_s por R_g), que não estão relacionadas com o CMRR.

A entrada v_{ref} permite adicionar um valor (normalmente DC) arbitrário à saída (por exemplo para compensar o *offset* do amplificador). Em configurações bipolares está normalmente ligado à massa.

B. Diferentes alternativas de realização

A realização prática do diagrama de blocos Fig. 2, pode ser efectuada usando diferentes topologias. A Fig. 3 apresenta quatro diferentes possibilidades [2]-[6]. Apesar de possuírem a mesma filosofia, o seu comportamento não é obviamente idêntico. Para se proceder à eleição da configuração mais indicada, foi feito um estudo comparativo. Neste estudo foram avaliados os seguintes itens: consumo total, ruído equivalente referido à entrada,

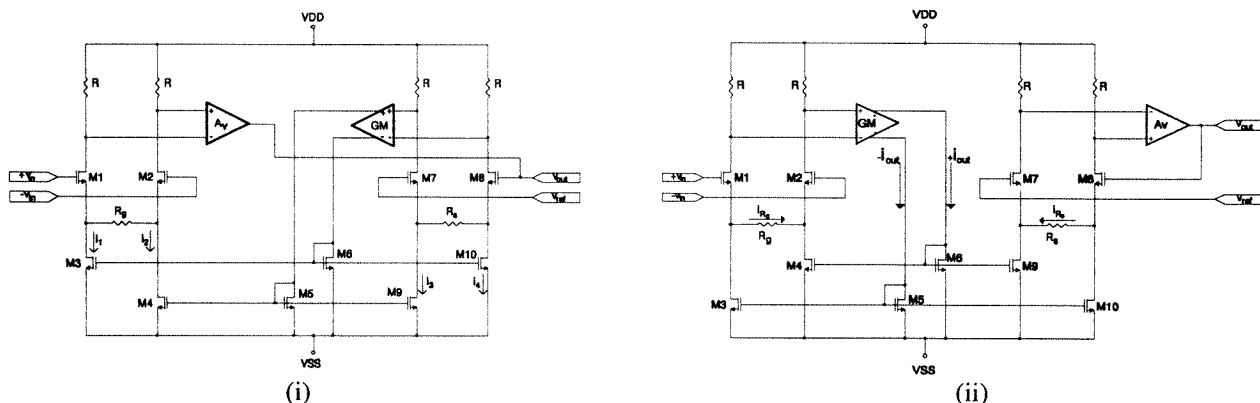


Fig. 3. Possíveis realizações do amplificador de instrumentação com realimentação em corrente.

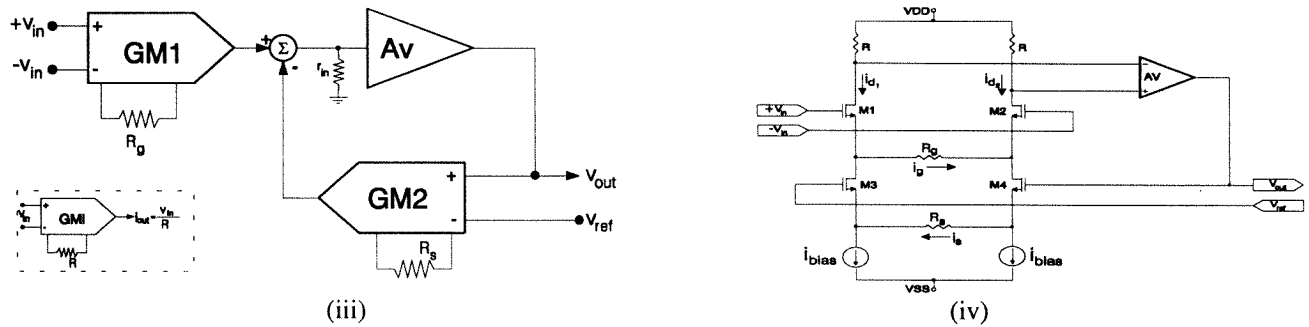


Fig. 3. Possíveis realizações do amplificador de instrumentação com realimentação em corrente (continuação).

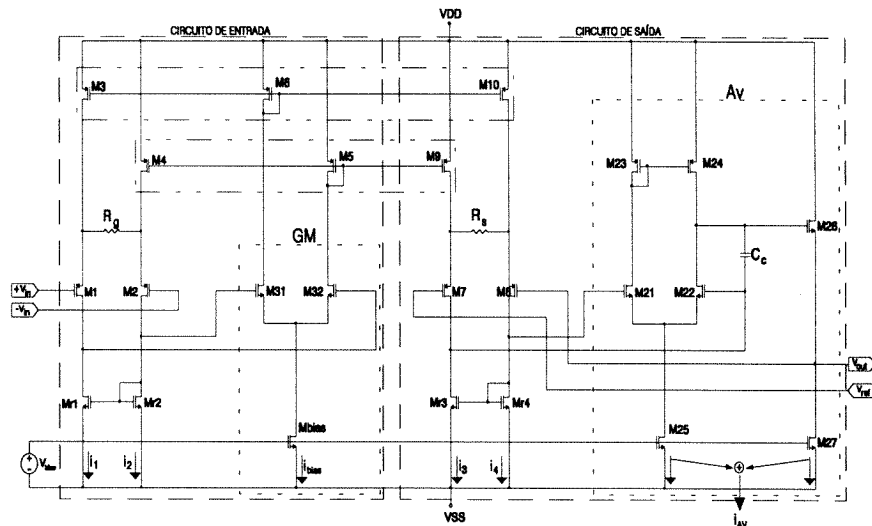


Fig. 4. Esquema do amplificador de instrumentação.

precisão do ganho, CMRR, excursões de sinal à entrada e saída (análise dc) e ganho em malha aberta. O Quadro 1 apresenta um resumo das conclusões obtidas.

	Topologia (i)	Topologia (ii)	Topologia (iii)	Topologia (iv)
CONSUMO	BOM	BOM	SUF.	Muito BOM
RUÍDO	BOM	BOM	SUF.	SUF.
PRECISÃO	BOM	BOM	Muito BOM	MAU
CMRR	BOM	BOM	BOM	BOM
ANÁLISE DC	BOM	BOM	SUF.	MAU
OPEN-LOOP	SUF.	BOM	SUF.	SUF.

Quadro 1. Resumo das características das diversas topologias.

Depois de analisadas todas as topologias, aquelas com um comportamento mais equilibrado foram sem dúvida as topologias (i) e (ii). A topologia (iv) tendo um consumo extremamente baixo, tem problemas em quase todos os restantes itens, pelo que penso que será a indicada apenas naqueles casos onde o consumo é o único ponto importante. A topologia 2 tem enormes potencialidades. Se o consumo não for um problema será uma boa escolha. O que nos leva de novo às topologias (i) e (ii). Contudo, estas têm uma diferença que não é patente no quadro: é que não apresentam o mesmo comportamento dinâmico. De facto, nesta configuração existe realimentação global, o que viola um dos pressupostos apresentados. Daqui resulta que o problema

da compensação é mais complicado com esta configuração., pelo que a escolha recai sobre (ii).

III. PROJECTO DO AMPLIFICADOR

Após escolhida aquela considerada como a melhor topologia, procedemos ao seu dimensionamento, para o qual foi seguido um percurso lógico. Na Fig. 4 é apresentado o esquema completo da topologia (ii). De notar, que foram escolhidos transistores de entrada com canal P, uma vez que apresentam menor ruído *flicker* para uma dada área de porta. Nesta figura os blocos Av e GM foram também substituídos por o seu esquema detalhado. Pode observar-se que ambos os blocos são constituídos por um amplificador diferencial, sendo Av complementado por um seguidor de fonte para fazer baixar a sua impedância de saída (M26 e M27).

Assim, tendo como base a minimização do consumo de corrente, tal como do ruído, é calculada corrente de polarização e o dimensionamento dos transistores de entrada, respectivamente. Depois é analisada a precisão do ganho do circuito de entrada e dimensionado o circuito de saída. É verificado também o comportamento na frequência e o comportamento dc. São apresentados os circuitos que realizam a função de filtragem passa-alto e passa-baixo e por último são abordados alguns detalhes de *layout*.

A. Cálculo da corrente de polarização

Estando esta corrente fortemente ligada ao consumo de potência total, que é um dos parâmetros mais importantes, face aos objectivos definidos, o seu valor é de importância capital. Mais o é, porque a corrente de polarização encontram-se inevitavelmente associadas também outras características importantes, e aqui refiro como exemplo, o facto do ganho de transcondutância de um transistor MOS crescer com a corrente de polarização, o que por sua vez influencia outros parâmetros como a estabilidade, ganho, ruído, etc.

Assim impondo um $(W/L)_{max} = 25$, temos que a corrente de polarização num dado ramo, deverá ser para transistores de canal-P: $I_D = 2 \mu A$. Para chegar-se a este valor, foi decidido que os transistores deviam trabalhar num ponto intermédio entre a zona de inversão moderada e a inversão forte, onde o $(V_{GS} - V_{th}) \approx 0.18V$, onde existe um bom compromisso entre características do transistor e baixo consumo [7].

Usando o valor de cima para os transistores M31 e M32 (implica $I_{bias} = 4 \mu A$) e fazendo o factor de amplificação dos espelhos de corrente (M3-M6-M10) e (M4-M5-M9) igual a 2, resulta: $I_1 = I_2 = I_{bias} = I_3 = I_4$, pelo que se também $I_{AV} = I_{bias}$, temos que:

$$I_{total} = 6 \cdot I_{bias} = 24 \mu A$$

B. Dimensionamento dos transistores de entrada

Os transistores de entrada M1 e M2 em primeiro lugar e os transistores de carga Mr1 e Mr2 depois, são os componentes mais críticos do amplificador de instrumentação, uma vez que determinam o ruído e o CMRR, assim como o seu *offset*. Um bom comportamento nestes últimos aspectos é conseguido fazendo estes transistores com dimensões muito superiores às mínimas e usando técnicas de *layout* especiais (como será apresentado mais adiante) e a minimização do ruído tem que ter em atenção a componente ruído *flicker*, muito importante em amplificadores CMOS que funcionam a baixas frequências.

A densidade de ruído (equivalente à entrada) é dado (1º termo - ruído térmico ; 2º termo - ruído *flicker*) por:

$$v_i^2 = \frac{8kT}{3g_m} + \frac{K_f}{C_{ox}^2 WL} \cdot \frac{1}{f} \quad (2)$$

Assim usando o termo de (2) correspondendo ao ruído térmico, obtem-se para uma razão $(W/L) = 25$ dos transistores de entrada, o que corresponde a $g_m = 60 \mu S$, uma densidade de ruído de $13 nV/\sqrt{Hz}$ por transistor

Por se verificar que a regra prática usualmente utilizada para calcular os transistores de carga, que afirma que a transcondutância dos transistores de entrada deve ser 3

vezes a dos transistores de carga, não dar bons resultados quando se considera o ruído *flicker*, foi feita uma investigação mais cuidada onde foi calculado o parâmetro *Y-noise excess factor*, que normaliza densidade total de ruído à entrada com aquela obtida para apenas um dos transistores de entrada. Para o nosso caso temos (e ruído *flicker* Y_F):

$$Y_F = \frac{v_{n_{total}}^2}{v_{n_{M1}}^2} \Big|_{Flicker} = 2 \left[1 + \frac{Kf_N}{Kf_P} \cdot \left(\frac{g_{m_{Mr1}}}{g_{m_{M1}}} \right)^2 \cdot \frac{(WL)_{M1}}{(WL)_{Mr1}} \right]$$

Para que a contribuição dos transistores de carga seja inferior a $\approx 10\%$ no ruído total é preciso verificar:

$$\frac{g_{m_{M1}}}{g_{m_{Mr1}}} > 3 \sqrt{\frac{Kf_N}{Kf_P} \cdot \frac{(WL)_{M1}}{(WL)_{Mr1}}} \quad (3)$$

A condição (3) é muito difícil de obter, para transistores Mr1 e Mr2 com dimensões não muito elevadas, pelo que no nosso caso foi escolhido $Y_F = 4$ (o ruído *flicker* gerado pelos transistores de entrada é igual ao dos transistores de carga). Assim para um ruído total integrado na banda (0.3 - 150 Hz) de $1.2 \mu V$, usamos transistores com $(W/L)_{M1} = (600/25)$ e $(W/L)_{Mr1} = (24/200)$, em que todas as dimensões estão em μm . Para estes valores o efeito dos transistores de carga no ruído térmico é muito baixo, pelo que pode ser desprezado. De facto neste caso temos: $Y_T = 2 \cdot (1 + g_{m_{Mr1}}/g_{m_{M1}}) = 2.17$ (notar que o valor mínimo é 2 - todo o ruído é gerado nos transistores de entrada).

C. Precisão do ganho do circuito de entrada

A função do circuito de entrada é a de um amplificador de transcondutância com ganho $1/R_g$. Na prática só se consegue uma aproximação a este valor ideal que importa examinar. O erro relativo em função da transcondutância do bloco GM é apresentado na Fig. 5.

Conforme se pode ver da Fig. 5 os valores calculados anteriormente (ponto P) levam a um erro de $\approx 0.4\%$, que se considera satisfatório, pelo que não é preciso fazer nenhuma alteração ao projecto inicial.

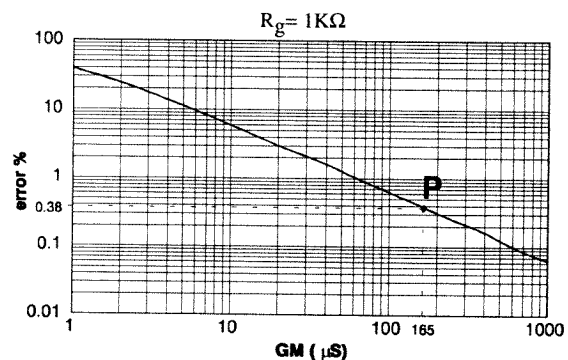


Fig. 5. Erro relativo vs. ganho de transcondutância do bloco GM.

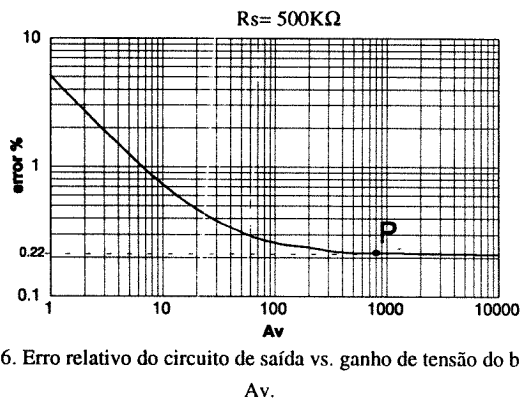


Fig. 6. Erro relativo do circuito de saída vs. ganho de tensão do bloco A_v .

D. Dimensionamento do circuito de saída

O circuito de saída não é tão crítico como o circuito de entrada, pois a sua influência em pontos como o ruído e o CMRR é muito reduzida. É no entanto importante garantir tal como fizemos no circuito de entrada, que a sua precisão (na definição do ganho de transresistência - idealmente R_s) seja elevada. Por isso temos que escolher um ganho A_v suficientemente elevado. No caso de $(W/L)_{M7,M8}=(240/10)$ o que é equivalente em termos de g_m ao dos transistores M1 e M2 (a razão é igual apesar das dimensões serem reduzidas) temos a curva da Fig. 6.

Conforme apresentado em Fig.6 o amplificador A_v foi projectado para um ganho 800, correspondendo a um erro de 0.22%. Entrando com o erro do circuito de entrada, o erro total máximo é de cerca de 0.6%, dentro portanto das especificações.

E. Análise na frequência

Como o amplificador de instrumentação é constituído por dois blocos distintos (um amplificador de transcondutância e um amplificador de transresistência), sem que exista realimentação de um para o outro, a resposta na frequência depende directamente do comportamento de cada um dos blocos individuais, pelo que a análise pode ser feita separadamente.

Para fazer esta análise calculamos os polos em malha aberta do bloco a analisar, considerando no entanto o efeito da carga que a malha de realimentação representa. Depois usando apenas os dois polos com frequência

MALHA ABERTA			MALHA FECHADA					
A_0	s_1	s_2	A_f	s_1	s_2	Q^2	f_{pico}	A_{pico}
0.27	-2230	-106	10^3	$-54 \cdot 10^3 \pm j170 \cdot 10^3$		2.7	220	4.6
A/V	Hz	KHz	A/V	Hz			KHz	dB

Quadro 2. Resposta em frequência do circuito de entrada.

MALHA ABERTA			MALHA FECHADA					
A_0	s_1	s_2	A_f	s_1	s_2	Q^2	f_{pico}	A_{pico}
$8 \cdot 10^{10}$	-1.5	-300	500	$-150 \cdot 10^3 \pm j370 \cdot 10^3$		1.8	340	3.3
V/A	Hz	KHz	KV/A	Hz			KHz	dB

Quadro 3. Resposta em frequência do circuito de saída

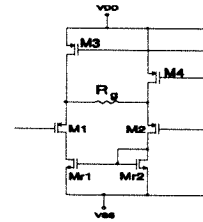


Fig. 7. Entrada do amplificador de instrumentação.

angular mais baixa, admitindo que cada amplificador pode ser aproximado por um sistema de 2ª ordem, e com base na teoria deste tipo de sistemas, calculamos os polos em malha fechada. Com este método são obtidos os resultados seguintes: (Quadros 2 e 3).

Estes resultados foram comprovados pelas simulações efectuadas e mostram a disponibilidade deste tipo de amplificadores de instrumentação, para serem usados em sistemas que requerem muito superiores larguras de banda. Obviamente, nesta aplicação foram adicionados circuitos ao esquema base da Fig. 4 para uma largura de banda compatível com os sinais EEG (ver circuitos de filtragem).

F. Análise dc

Um ponto importante quando se projecta um amplificador é a especificação do seu CMR (*Common Mode Range*), definido como a gama de valores de tensão de modo comum à entrada para o qual amplificador continua a amplificar fielmente o sinal diferencial. Esta característica depende da maneira como o circuito de entrada está polarizado.

Para um funcionamento correcto é preciso que todos os transistores trabalhem na zona de saturação. Analisando o circuito da Fig. 7 resulta que para essa situação acontecer é preciso verificar as seguintes condições:

$$V_{G2} > V_{SS} + V_{IO_{NMOS}} + \sqrt{\frac{2 \cdot I_{bias} \cdot L_{Mr2}}{K_N \cdot W_{Mr2}}} - V_{IO_{PMOS}} - \gamma \left(\sqrt{2|\phi_F| + V_{SB_{M2}}} - \sqrt{2|\phi_F|} \right)$$

$$V_{G1} < V_{DD} - \sqrt{\frac{2 \cdot I_{bias} \cdot L_{Mr3}}{K_P \cdot W_{Mr3}}} - \left(\sqrt{\frac{2 \cdot I_{bias} \cdot L_{M1}}{K_P \cdot W_{M1}}} + |V_{IO_{PMOS}}| + \gamma \left(\sqrt{2|\phi_F| + V_{SB_{M2}}} - \sqrt{2|\phi_F|} \right) \right)$$

Substituindo os valores obtemos para o amplificador de instrumentação: $-3.8V < CMR < 1.5V$. A excursão máxima do sinal de saída, dada a simetria do amplificador de instrumentação, é semelhante à gama de modo comum do circuito de entrada. Reparar que apesar de M7 e M8 não terem as mesmas dimensões de M1 e M2, possuem a mesma razão W/L. É evidente que admitimos que não existem problemas com o seguidor de fonte.

G. Circuitos de filtragem

Com a intenção de reduzir o ruído ao mínimo, o amplificador de instrumentação inclui circuitos que o tornam um filtro passa-banda (0.3 - 150Hz). A acção

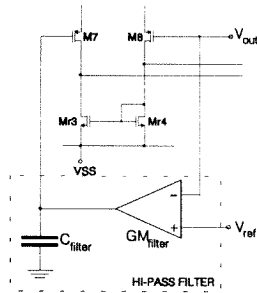


Fig. 8. Solução usada para realizar o filtro passa-alto.

passa baixo é realizada colocando um condensador C_s em paralelo com R_s . Isto causa um polo à frequência:

$$f_H = 1 / (2\pi \cdot R_s \cdot C_s)$$

A característica passa-alto é mais difícil de ser realizada. O uso de um filtro passivo RC não é uma boa solução para uma frequência de corte tão baixa, pelo que foi decidido usar outra malha de realimentação no circuito de saída, como é mostrado na Fig. 8.

Dentro da nova malha de realimentação existe um integrador constituído por GM_{filter} e C_{filter} . O primeiro age como uma resistência, mas oferece duas vantagens em relação a uma verdadeira: 1- Como é possível fazer a sua transcondutância baixa, uma resistência equivalente de valor elevado pode ser obtida ($>1M\Omega$). 2- Não há uma carga resistiva na saída do IA e como é evidente da Fig. 8 um terminal de referência v_{ref} continua a existir. Com esta configuração obtém-se uma frequência de corte à frequência.

Dentro da nova malha de realimentação existe um integrador. GM_{filter} age como uma resistência, mas oferece duas vantagens em relação a uma verdadeira: Primeiro, como é possível fazer a sua transcondutância baixa, uma resistência equivalente de valor elevado pode ser obtida ($>1M\Omega$). Segundo, não há uma carga resistiva na saída do IA e como é evidente da Fig. 8 um terminal de referência v_{ref} continua a existir. Com esta configuração obtém-se uma frequência de corte à frequência:

$$f_L = GM_{filter} / (2\pi \cdot C_{filter})$$

H. Técnicas de layout

Muitas vezes circuitos analógicos que funcionam excelentemente nas simulações realizadas, falham totalmente quando testados experimentalmente. Se nos abstermos de problemas relacionados com defeitos de fabrico, a grande causa de desvios em relação aos resultados esperados está ligada a aspectos práticos, onde cálculos teóricos e as simulações pouco ajudam.

Uma das questões mais importantes que se levantam é o emparelhamento de dispositivos (transistores, condensadores e resistências). A compreensão integral deste ponto é um requisito fundamental para o projecto de sistemas que exijam elevada precisão [8]. No projecto de

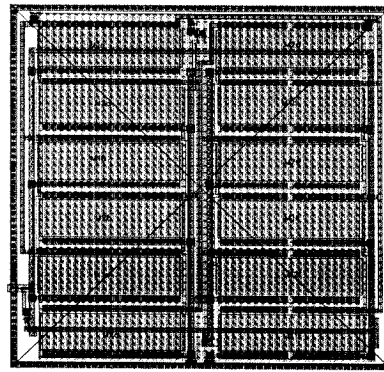


Fig. 9. Layout dos transistores de entrada.

um amplificador de instrumentação, obviamente que estes problemas existem e têm que ser tratados com atenção. Aqui a chave para se atingir um elevado CMRR, é um bom emparelhamento dos transistores de entrada, pelo que no desenho do layout destes componentes foram tomados cuidados especiais. Assim, para além de possuírem dimensões muito superiores às mínimas, para desta maneira combater-se as flutuações aleatórias da periferia dos transistores, foram usadas estruturas com centróide comum [9].

Foi também procurado fazer as conexões entre as diversas partes simétricas, mesmo que para tal tivéssemos que usar linhas que não eram precisas. Deste modo procurou-se fazer com que as capacidades parasitas, fossem o mais iguais possíveis nos dois transistores. Isto optimiza o comportamento na frequência de parâmetros como o CMRR.

Por último convém acrescentar que foram colocados inúmeros contactos de substrato e à N -well para reduzir as possibilidades de latch-up. Existe ainda uma série de contactos ao substrato, ao longo do perímetro do amplificador de instrumentação, para desta maneira o acoplamento resistivo (com outros componentes, especialmente digitais) através do substrato ser minimizado.

III. OUTROS CIRCUITOS REALIZADOS

A. Amplificador final

Quando descrevemos o sistema de aquisição, referimos que o ganho total é distribuído pelo amplificador de instrumentação e por um amplificador final (ver Fig. 1) que deveria ter um ganho variável entre 1 e 10. Como à saída do AI o nível de sinal já é elevado, a precisão e as características de ruído que o amplificador de saída tem que apresentar são bastante relaxadas. Por esta razão foi decidido usar uma configuração simples - Fig. 10, com dois andares de ganho, baseada no amplificador operacional de Miller, com um andar de saída em classe AB (Fig. 10), que se ligou na sua configuração não inversora.

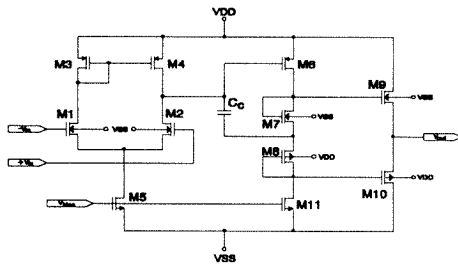


Fig. 10. Esquema do amplificador final.

A escolha de um andar de saída em classe AB, prende-se com a maior eficiência em termos de corrente consumida que apresenta, quando comparada com a configuração em classe A. Para este, a capacidade de *sinking* é limitada pela corrente de polarização, estando à partida definida. Com a montagem em *push-pull* esta limitação não existe. De facto é possível polarizar M9 e M10 com uma dada corrente muito mais baixa, do que aquela que podem fornecer sob condições dinâmicas. Assim temos que uma configuração em classe AB apresenta uma impedância de saída baixa, uma capacidade de *source* e *sink* de corrente elevada (e simétrica), enquanto o consumo é mantido baixo.

Conforme referido o amplificador operacional encontra-se ligado numa configuração não inversora, e o ganho é programado actuando num multiplexador de 4:1 que liga à massa de sinal, uma de 4 resistências com o valor conveniente para o ganho pretendido.

B. Oscilador de calibração

É impossível, por mais cuidados que se tenham que todos os canais tenham exactamente a mesma resposta, nomeadamente no que diz respeito ao ganho. Este facto pode no entanto ser resolvido, se antes de se proceder a uma aquisição efectiva, injectarmos um sinal de referência comum em todos os canais, que usado com o *software* conveniente, faça a sua calibração. Nos sistemas EEG, este sinal é normalmente uma onda quadrada, com uma frequência de cerca de 10Hz, e uma amplitude de

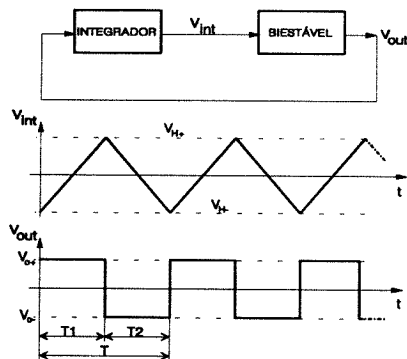


Fig. 11. Oscilador não sintonizado: $T = 4 \cdot V_H / K \cdot V_O$
(K é o ganho do integrador).

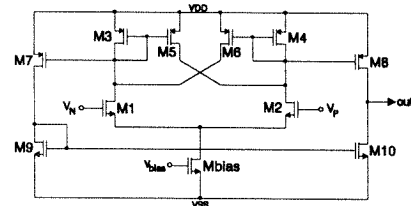


Fig. 12 Comparador com histerese (circuito biestável).

aproximadamente 50μV. Um oscilador pode ser realizado ligando um integrador e um biestável em cascata como na Fig. 11.

Para redução da área e baixo consumo de corrente foi decidido que o biestável não deveria possuir resistências. O circuito adoptado [10] é apresentado na Fig. 12.

Um integrador analógico felizmente pode ser realizado de uma maneira muito simples. Basta injectar corrente num condensador: a tensão resultante é o integral da corrente. Com este princípio é possível realizar um oscilador com a configuração da Fig. 13.

O funcionamento do circuito é simples e idêntico ao do esquema genérico da Fig. 11. Aqui a integração é feita pelo condensador C e resulta que o período de oscilação é dado por: $T = 4 \cdot C \cdot V_H / I_{bias}$. Invertendo esta equação e substituindo V_H , resulta a frequência de oscilação:

$$f_{osc} = \frac{I_{bias}}{4 \cdot C \cdot \sqrt{\frac{2 \cdot I_{bias}}{K \cdot S_1 \cdot (S_3 + S_5)}} \cdot (\sqrt{S_5} - \sqrt{S_3})}$$

Nas saída da Fig. 13, a frequência não depende da tensão de alimentação, mas a sua amplitude não está obviamente estabilizada. Por outro lado como o valor da tensão de calibração deve ser muito baixo conforme referido no início. Estas duas questões foram resolvidas usando um circuito semelhante ao do integrador, mas substituindo o condensador por uma resistência (ver Fig. 14). A amplitude do sinal de saída fica: $V_{out} = R \cdot I_{bias} \cdot (S_{Msink} / S_{Mbias})$ e é independente das tensões de alimentação.

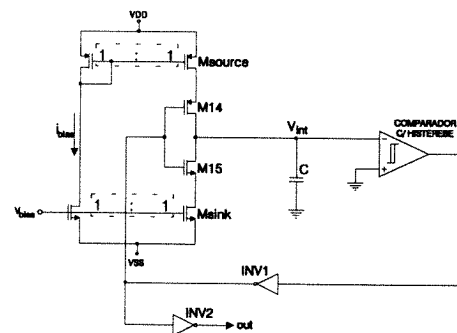


Fig. 13. Esquema simplificado de um oscilador alternativo.

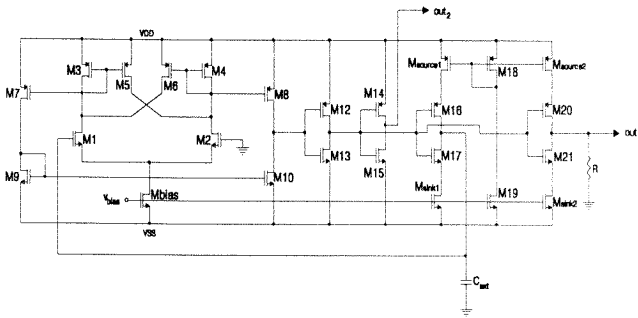


Fig. 14. Esquema completo do oscilador.

C. Gerador de Tensão/corrente de referência

Como vimos em todos os circuitos houve necessidade de ter um tensão ou uma corrente estável. Como o sistema é alimentado com pilhas ou baterias, esta não pode ser retirada directamente das alimentações.

Existem diversos circuitos que geram tensões/correntes de referência. O tipo mais usado - *the bandgap reference*, tem a vantagem de apresentar um coeficiente de temperatura muito baixo [11], mas exige contudo o uso de transistores bipolares. Foi decidido usar uma alternativa que só usasse transistores MOSFET. Para além disso, na aplicação em questão, não existe necessidade de se ter uma estabilidade com a temperatura muito alta (como por exemplo em conversores A/D e D/A). Assim foi usada a configuração da Fig. 15 [8],[12].

Nesta configuração o que se faz é aproveitar o facto de a tensão V_{GS} do transistor M3 ser constante se a sua corrente de dreno for fixa. Isto acontece porque a corrente de dreno de M3 é o espelho da corrente em R (por acção de M1 e M2), que admitindo V_{GS} constante, é obviamente também constante: $I_R = V_{GS3}/R$. Temos assim que o comportamento deste tipo de fontes (também chamadas *bootstrapped*, devido ao fenómeno apontado) é descrito por duas equações:

$$V_{GS3} = R \cdot I_{REF} \quad \text{e} \quad V_{GS3} = V_{th} + \sqrt{\frac{2 \cdot I_{REF}}{S_{M3}}}$$

O ponto de funcionamento está na intersecção destas duas equações. Infelizmente como se pode ver também na Fig. 15 existe uma solução trivial (ponto T) em que a corrente assim como V_{GS3} é nula, que deve ser evitada.

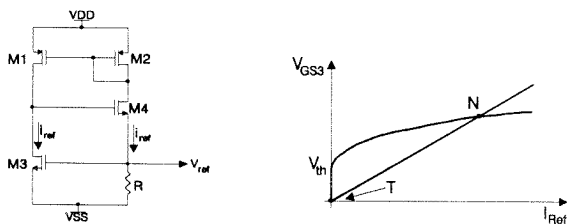
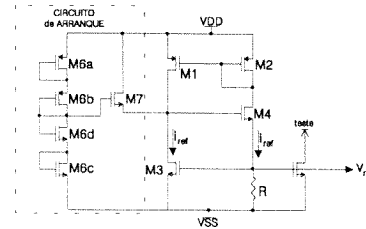
Fig. 15. Fonte de tensão *self-biased*.

Fig. 16. Esquema incluindo o circuito de arranque.

Para garantir que quando se liga a fonte, se vá para o ponto normal de funcionamento (N), é preciso adicionar um circuito de arranque. A solução que foi adoptado é apresentada na Fig. 16.

O circuito de arranque consiste num transistor - M7, que está polarizado com uma tensão de porta, tal que no início seja capaz de fornecer uma corrente que obrigue o circuito a deslocar-se para o ponto de funcionamento N, e que fique cortado à medida que N vai sendo atingido, de tal maneira que neste ponto não desempenhe qualquer papel no circuito. Para obter a tensão de polarização de M7, foi usada uma cadeia de 4 transistores (M6a-d), para minimizar a corrente consumida pelo próprio divisor (e a sua área).

IV. RESULTADOS EXPERIMENTAIS

Acabamos com um chip com área total de aproximadamente 25 mm² (Fig. 17), incluindo *input/output pads*. Esta podia ser reduzida se o número de pinos externos fosse reduzido, pois o chip é *pad limited*, o que é aliás um bom tópico para trabalho futuro.

Os resultados experimentais são apresentados nas próximas figuras, onde é ilustrado o bom funcionamento dos diversos blocos do circuito integrado. Nalguns tópicos são confrontados com os resultados das simulações. No fim os resultados mais importantes são sumarizados.

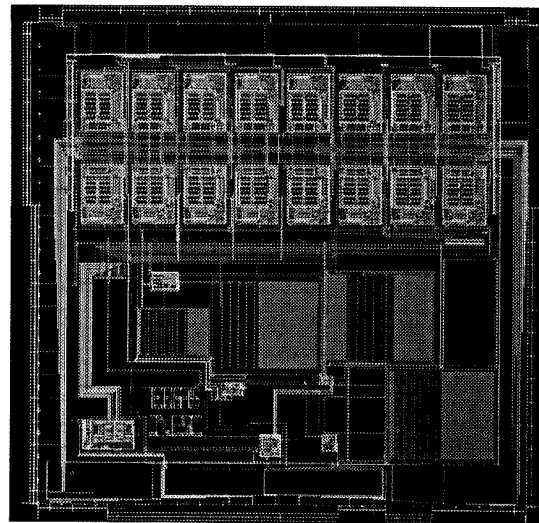
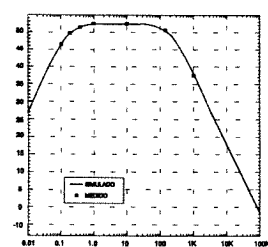
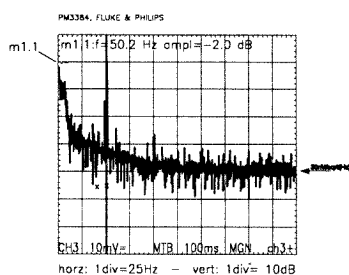


Fig. 17. Layout do circuito realizado

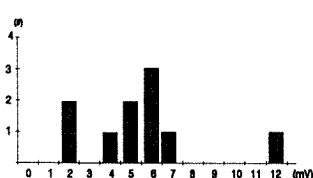
A. Amplificador de instrumentação



(i) Resposta na frequência.

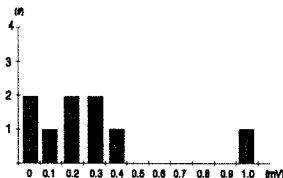


(ii) Ruído- FFT.

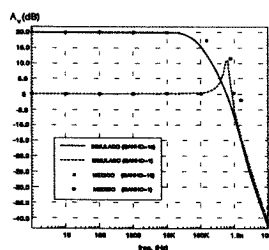


Offset - histograma (ver nota 1).

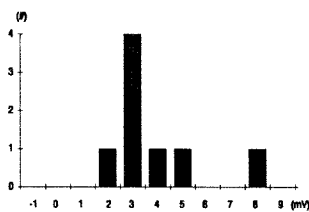
Esquerda: com circuito filtragem ; Direita: Sem circuito filtragem.



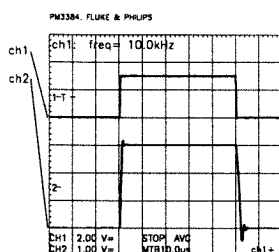
B. Amplificador final



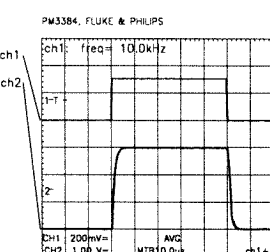
(i) Resposta na frequência.



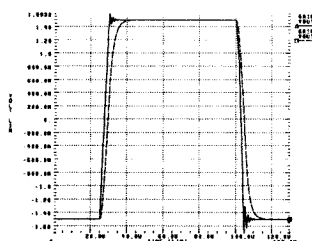
(ii) Tensão de offset (entrada).



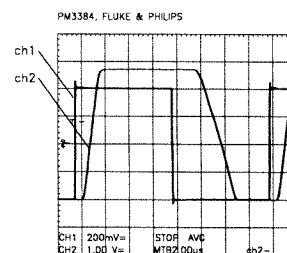
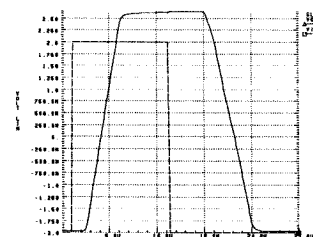
(ganho= 1)



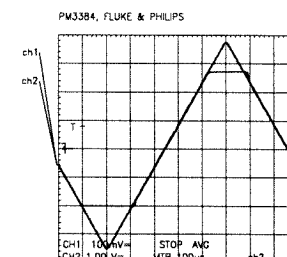
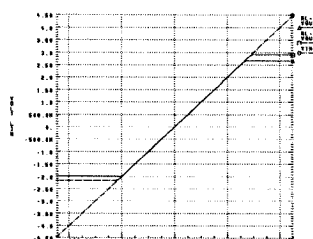
(ganho=10)

(iii) Resposta transiente ($C_L=4.7\text{pF}$) - CH1: V_{in} ; CH2: V_{out} .

(simulação)

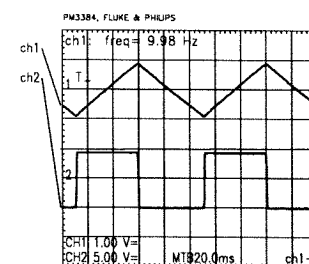
(iv) Slew-rate - CH1: V_{in} ; CH2: V_{out} .

(simulação)

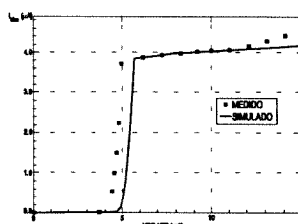
(v) Resposta dc - CH1: V_{in} ; CH2: V_{out} .

(simulação)

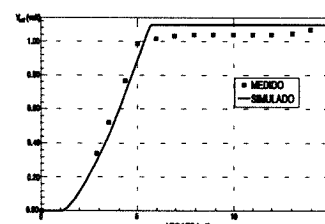
C. Oscilador de calibração

(i) Oscilador - CH1: Sinal no condensador C ; CH2: Tensão de saída V_{out2} .

D. Gerador de Tensão/Corrente



(i) A corrente versus (VDD-VSS).

(ii) V_{bias} versus (VDD-VSS).

nota 1) O gráfico da direita refere-se ao *offset* de entrada dos amplificadores de instrumentação, quando o circuito de filtragem passa-alto está desactivado. O gráfico da esquerda refere-se ao *offset* devido ao circuito de filtragem passa-alto. Este *offset* é medido directamente na saída e não depende do ganho. É obvio que apesar do *offset* à entrada do amplificador ser anulado pelo filtro passa-alto, é este que tem maior importância, pois mostra qual é o equilíbrio do AI.

E. Resumo das principais características

Os resultados estão apresentados no Quadro 4.

PARÂMETRO	valor	Observações
Área activa	24 mm ²	
Tensão de alimentação	±4.5 V	
Corrente consumida	520 µA	valor típico
Erro máximo do ganho	0.9 %	Qualquer ganho
Ruído total na banda (entrada)	1.4 µV (RS)	0.3 < LB < 150 Hz
CMRR	99 D.B.	valor típico
Offset (à entrada)	0.29 mV	valor médio
Offset (à saída com filtro)	5.7 mV	valor médio
PSRR	40 dB	valor típico
Gama de modo comum (AI)	-3.8 < CMR < 1.5 V	
idem (Amp. final)	-2 < CMR < 2.5 V	R _L = 56K
Slew-Rate (Amp. final)	0.5 V/µS	
f _{osc}	9.8 Hz	Oscilador de calibração
Amplitude	56 µV	Oscilador de calibração
Regulação de linha (I _{out} = 4 µA)	50 nA/V	tensão/corrente de ref.
Frequência máxima de clock	50 KHz	do multiplexador

Quadro 4. Comportamento típico.

V. CONCLUSÕES

É possível integrar num circuito integrado monolítico de dimensões médias (5 × 5 mm²), usando uma tecnologia puramente CMOS, um sistema de aquisição de sinais EEG com 16 canais, consumindo menos do que 0.5mA (excluindo ADC), solução que apresenta (comparativamente a implementações discretas) vantagens em termos de autonomia, volume, peso e custos (dado o baixo custo da tecnologia utilizada).

Foi ainda mostrado que os amplificadores de instrumentação com realimentação em corrente podem ser integrados em tecnologia CMOS com sucesso, apesar das limitações que esta tecnologia possui.

AGRADECIMENTOS

Este trabalho foi realizado no âmbito do projecto JNICT: PMCT/CISAU/806/90 e com o apoio da bolsa de mestrado da JNICT no âmbito do programa CIÊNCIA: BM/1772/91.

NOTAÇÃO USADA

W = largura do canal de um transistor MOSFET

L = comprimento do canal de um transistor MOSFET
 S = W/L
 V_D = tensão dreno-substrato
 V_S = tensão fonte-substrato
 V_G = tensão porta-substrato
 I_D = corrente de dreno
 k = constante de Boltzman
 T = temperatura (°K)
 q = carga do electrão
 $U_T = kT/q$
 C_{ox} = Capacidade do óxido da porta por unidade de área
 μ = mobilidade do canal
 $K' = \mu \cdot C_{ox}$ (parâmetro de transcondutância)
 V_{to} = Tensão de limiar com $V_G = 0V$
 V_{th} = Tensão de limiar
 N_B = Concentração de impurezas do substrato
 n_i = Concentração intrínseca de portadores
 $\Phi_F = U_T \ln(N_B/n_i)$ Potencial de Fermi

REFERÊNCIAS

- [1]- R. Cooper, J. W. Osselton and J. C. Shaw, "EEG Technology"-3rd Ed.; Butterworths - London, 1980.
- [2]- H. Krabbe, "A High Performance Monolithic Instrumentation Amplifier"; *ISSCC Dig. Tech. Papers*, Feb. 1971, pp. 186-187.
- [3]- Jacob Millman, Arvin Grabel, "Microelectronics"; McGraw-Hill International Editions, 1988.
- [4]- A. Paul Brokaw, Michael P. Timko, "An Improved Monolithic Instrumentation Amplifier"; *IEEE J. Solid-State Circuits*, vol. SC-10, no.6 - December 1975, pp. 417 - 423.
- [5]- Rudy J. Van De Plassche, "A Wide-Band Monolithic Instrumentation Amplifier"; *IEEE J. Solid-State Circuits*, vol. SC-10, no.6 - December 1975, pp. 424 - 431.
- [6]- Michael S. J. Steyaert, Willy M. C. Sansen and C. Zhongyuan, "A Micropower Low-Noise Monolithic Instrumentation Amplifier for Medical Purposes"; *IEEE J. Solid-State Circuits*, vol. SC-22, no.6 - December 1987, pp. 1163 - 1168.
- [7]- Willy Sansen, "Advanced Analog Design"; *CMOS -VLSI Design Course*, Lausanne - September 1989.
- [8]- Gregorian R., Temes G., "Analogue MOS Integrated Circuits for Signal Processing"; J. Wiley, 1986.
- [9]- J. L. McCreary and P. Gray, "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques-Part 1"; *IEEE J. Solid-State Circuits*, vol. SC-10, no.6 - December 1975, pp. 371-379.
- [10]- D. J. Allstot, "A Precision Variable-Supply CMOS Comparator"; *IEEE J. Solid-State Circuits*, vol. SC-17, no.6 - December 1982, pp. 1079-1080.
- [11]- R. J. Widlar, "New Developments in IC Voltage Regulators"; *IEEE J. Solid-State Circuits*, vol. SC-6, no.1 - February 1971, pp. 2-7.
- [12]- Randall L. Geiger, Phillip E. Allen and Noel R. Strader, "VLSI Design Techniques for Analog and Digital Circuits"; McGraw-Hill International Editions, 1990.

A Simulator Function Library for the SWIFT LAN Manager Prototype

Armando J. Pinho, Fernando M. S. Ramos

Abstract - This paper gives a description of the Simulator part of the SWIFT LAN Manager prototype, developed at INESC, as a contribution to the EURET/SWIFT Project. The editor tool used to manipulate simulator event files and the encoding used to represent actions are also described.

Resumo - Este artigo descreve parte do trabalho desenvolvido no INESC respeitante ao protótipo de um LAN Manager para o projecto EURET/SWIFT, projecto co-financiado pela Comissão da União Europeia. Em particular é descrito o núcleo do simulador assim como o editor dos ficheiros de simulação e o respectivo código usado para a representação de acções.

I. INTRODUCTION

The aim of the EURET/SWIFT (Specifications for Controller Working Positions in Future Air Traffic Control) project is to provide a proposal of detailed specification of the Controller Working Position (CWP) to be used in the development of future Air Traffic Control systems.

One important aspect of the definition of CWPs is the communications environment, and in particular the definition of the LAN Manager architecture and facilities that should be implemented. For the purpose of clarifying the concepts of LAN Management in SWIFT it was specified and developed a prototype of LAN Manager.

The SWIFT LAN Manager prototype includes two basic functional parts: the Human Machine Interface (HMI) and the Simulator. The HMI provides all the interaction with the user, including the support of user inputs and representation of outputs, and the execution of the kernel machine of the prototype. The Simulator part provides the HMI, at its request, with time controlled simulation events previously recorded in a simulator event file. Also, an Editor is provided as the tool to manipulate the simulator event files. The Editor, although working dissociated (i.e., off-line) from the Simulator and HMI, is considered as belonging to the Simulator part.

This paper gives a brief description of the design and implementation of the Simulator part of the prototype (including the Editor) which was developed at INESC, Portugal. The HMI part, that will not be described in this

paper, was developed by ESG, Germany, that is a private company partner of INESC in this SWIFT work package.

All the programming was done using the C language, under the UNIX operating system environment. Although the programs were intended to run on a HP 9000 machine, the code (at least the one used to implement the Simulator part) is easily portable to other architectures (in fact it was developed mostly under the LINUX version of UNIX, running on a 486 machine).

In the following sections we will give an overview of the most important aspects of the Simulator including the respective event file Editor, focusing on the implementation, file format and encoding of actions.

II. THE SIMULATOR KERNEL

The purpose of the simulator kernel that we describe here is the generation of events affecting a global data structure that encodes, in every moment, all the state of the current simulation. This data structure is also the connection between the simulator kernel and the HMI front-end. Briefly, the interaction between the HMI front-end and the simulator kernel is as follows. Periodically (typically every second), the front-end calls a function (the simulator kernel) that, if there is any action to be performed, it will be reflected in the global data structure. Therefore, the coupling between the HMI and the simulator kernel was kept quite low in order to avoid, as much as possible, the inter-dependency of these two modules. This was an important factor since the two modules were developed at distinct institutions, the HMI at ESG in Germany and the simulator kernel at INESC in Portugal.

The events can be of several types including structural and functional changes of the environment, and alarms. Nevertheless, they are treated all the same way by the simulator, i.e. as changes on the global data structure (these changes on the global data structure will be referred from now on as actions).

From the HMI point of view there is only a way to access the simulator kernel which is by calling the `sim_periodic_timer()` function. This function will determine if there is any action to be performed and will execute it if this is the case. The return value of this routine informs the HMI if the call succeeded, if the

simulation ended or if the simulator was not able to access the specified simulator event file.

Analyzing with more detail the simulator kernel we identify an internal timer that is responsible for the correct deliverance of the time programmed events. As we will show shortly, each event has a time tag that represents the time of occurrence of that event during simulation. The data structure that encodes an event is simply:

```
typedef struct {
    int      nActions;
    int      offsetTime;
    int      absoluteTime;
    char     **actions;
} SimEvent;
```

The parameters `offsetTime` and `absoluteTime` hold, respectively, the number of seconds that have to tic since the last event occurred, and the number of seconds since the begin of the simulation (obviously this is redundant information but makes things easier to handle). The parameters `nActions` and `actions` denote, respectively, the number of actions to be performed when that event occurs, and the actions encoded as strings (below we will explain the encoding used to represent actions as strings).

The simulation is controlled by another data structure that is loaded with data from a simulator event file at the start of a simulation:

```
typedef struct {
    FILE      *filePtr;
    char      *fileName;
    char      *creationDate;
    char      *lastChangeDate;
    int      numberOfEvents;
    int      currentEventNumber;
    SimEvent **events;
    int      simStartTime;
} SimEventFile;
```

Parameters `filePtr` and `fileName` contain information related to the simulator event file that is in use, while `creationDate` and `lastChangeDate` indicate the date of file creation and last modification. The other parameters are related with the simulation itself: number of events in file (`numberOfEvents`), indication of the event that is currently being accessed (`currentEventNumber`), a list of events (`events`), and the start time of the simulation obtained through the system clock (`simStartTime`).

When a simulation begins the global data structure is initialized with a default state, which mostly corresponds to null values of the parameters. Therefore, it is expected that the first event (that always occurs at simulation time zero) will fill the parameters accordingly to the desired initial scenario. All the subsequent actions are performed in an incremental fashion, i.e. only the changes are communicated.

The next section is devoted to provide a general overview of the editor tool. It includes also a description

of the simulator event file format and of the encoding used to specify actions.

III. THE EDITOR

The main goal of the editor tool is the manipulation of simulation event files, which contain sequences of sets of actions intended to feed the simulator. Each set of actions is considered a *record* or *event* if it shares the same time tag, corresponding to the time of occurrence of that event. Therefore, each event may consist of several actions that will be performed "simultaneously" (this means that when the time of occurrence of the event arrives all the actions associated will be performed as soon as possible, but at a non-specified order).

When a new file is created the global data structure contains the same initialization values as in the case of a simulation. In every moment of editing the global data structure reflects all past actions that are recorded in the file until the previous event. The current event generates only the actions needed to change the state of the previous event to the state of the current. This means that the editing, as the simulation, is also incremental.

In this section we describe three main items related to the editor tool: its implementation, the format of the simulator event files, and the encoding of actions.

A. The implementation of the editor

The editor was implemented using terminal screen oriented menus. The curses package [1] (system V terminal screen handling and optimization package) was the main source of screen handling routines that we used to develop the editor. This package has the advantage to be independent of the specific terminal used to run the programs, since it uses the terminfo [2] terminal capability data base to look for the appropriate escape sequences.

The main motivation to use a menu based editor was the reasonable complexity of the data structure that has to be handled during editing. This approach offers the advantage of easy move across all the data structure parameters making the changes in an arbitrary sequence. The organization of the menus follows closely the organization of the global data structure. For example, to the sub-structure `alarm_event`:

```
typedef struct alarm_event_type {
    Boolean    alarm_tag;
    id_type    CWP_id;
    id_type    resource_id;
    Int_16     alarm_no;
    char       alarm_message[MAX_mesg];
    Boolean    alarm_criticality;
    Int_8      alarm_type;
    time_type  alarm_log_time;
} alarm_event_type;
```

corresponds the menu items:

Alarm tag
 CWP id
 Resource id
 Alarm number
 Alarm message
 Alarm criticality
 Alarm type
 Alarm log time

The movimentation can be done using the cursor keys and the data is entered using fields with auto-validation. Some of the input fields offer a pre-defined list of choices that can be selected in an easy manner.

B. The simulator event file format

The simulator event files are ASCII files, starting with a header that is twofold. It is used to distinguish simulator event files from other files and also it provides information about the creation time and last change time of the file. Here is an example of the header of a simulator event file:

```
*****
LAN Manager simulator event file
Creation date: Thu Jul 21 12:34:11 1994
Last change : Thu Jul 21 12:38:14 1994
*****
```

Each event begins with a "Begin Event" string and ends with a "End Event" string. Following the "Begin Event" indication, two numbers (on separate lines) represent respectively the offset time (time in seconds since the previous event) and the absolute time (the time in seconds from the begin of the simulation) of the event. All the remaining information until the "End Event" mark is formed of encoded actions (one per line). Next is a short example of one of these files, containing just two events:

```
*****
LAN Manager simulator event file
Creation date: Thu Jul 21 12:34:11 1994
Last change : Thu Jul 21 12:38:14 1994
*****
```

```
Begin Event
0
0
00001
00011
00043825
000903445
000910
02001
020122
02030GATEW
02031012
```

End Event

```
Begin Event
10
10
1001
1011
104 This is an alarm!
1051
1071207
End Event
```

The meaning of the encoded actions are explained in the next section.

C. The encoding of actions

In order to record on file changes of some of the parameters of the global data structure (performed during editing) and read them back during simulation we developed an encoding scheme that we explain briefly. The basic idea was to use the tree organization of the data structure as the mean to access the end parameters. To clarify the idea let us provide an example. Suppose we have the following data structure definition:

```
typedef struct type1 {
    int      P1;                /* 0 */
    char     *P2;                /* 1 */
    double   P3;                /* 2 */
} type1;

typedef struct type2 {
    char     *P4;                /* 0 */
    int      P5;                /* 1 */
} type2;

typedef struct type3 {
    type1    p1;                /* 0 */
    type2    p2[MAX_ELEMS];     /* 1 */
} type3;
```

Suppose also that only type1 and type2 are composed types, i.e. P1, P2, ... are leaves of the data structure tree. Note the commented numbers on each parameter. They number the parameters on an arbitrary but pre-determined manner. Therefore, to indicate that P2 has value "Testing string" we can use the following encoding:

0 1 Testing string

where the first number (0) indicates parameter p1 (of the root tree), the second number (1) indicates parameter P2 of the data structure type1, and finally the value of the parameter is represented. In the same way

1 5 1 1254

indicates that the structure field p2[5].P2 contains value 1254.

As can be easily understood, the code generator needed to encode the actions is straight forward to design although somewhat tedious to implement.

The parser used to interpret the encoded actions is also easy to design since there is no lookahead, i.e. at every node it is always known what data type should be expected.

A simulator event file decoder was implemented as an aid tool associated with the editor. The output generated by this program when applied to the simulator event file given as example in the file format section is the following:

```
* RECORD 1 (Abs. time = 0, Offset time = 0)
Elems.CWP[0].CWP_existing -> TRUE
Elems.CWP[0].station_id -> 1
Elems.CWP[0].login_time -> 03:08:25
```

```
Elems.CWP[0].CWP_secur.Sec_attempt -> 3445
Elems.CWP[0].CWP_secur.Sec_state -> FALSE
Elems.Net_elem[0].Net_elem_existing -> TRUE
Elems.Net_elem[0].Network_No -> 22
Elems.Net_elem[0].node.node_name -> GATEW
Elems.Net_elem[0].node.congestions -> 12
```

```
* RECORD 2 (Abs. time = 10, Offset time = 10)
alarm[0].alarm_tag -> TRUE
alarm[0].CWP_id -> 1
alarm[0].alarm_message -> This is an alarm!
alarm[0].alarm_criticality -> TRUE
alarm[0].alarm_log_time -> 01:20:07
```

REFERENCES

- [1] curses(3X), CRT screen handling and optimization package. Hewlett-Packard Company, HP-UX Release 9.0, August 1992.
- [2] terminfo(4), terminal capability database. Hewlett-Packard Company, HP-UX Release 9.0, August 1992.

Uniformização de Interface de Programação para Aplicações Suportadas em Ambiente RDIS ou TCP/IP

Ilídio Ramalho, Fernando M. S. Ramos, Joaquim Arnaldo Martins

Resumo - Neste artigo é apresentada uma proposta de uma interface que permite uniformizar, do ponto de vista do desenvolvimento de aplicações, a utilização dos ambientes de comunicações RDIS ou TCP/IP.

Abstract - This paper proposes a unified programming application interface for applications based on ISDN or TCP/IP networks.

I. INTRODUÇÃO*

O objectivo deste trabalho foi a criação de uma interface única de acesso a dois dos ambientes de comunicações mais implementados actualmente, a Rede Digital com Integração de Serviços (RDIS) e o protocolo Internet TCP/IP. Com a criação desta Interface, pretende-se garantir a portabilidade de qualquer aplicação a desenvolver sobre a mesma, e que permita o estabelecimento de uma ligação de dados entre dois PC's. Em termos esquemáticos e, tendo em conta o modelo OSI, tem-se:

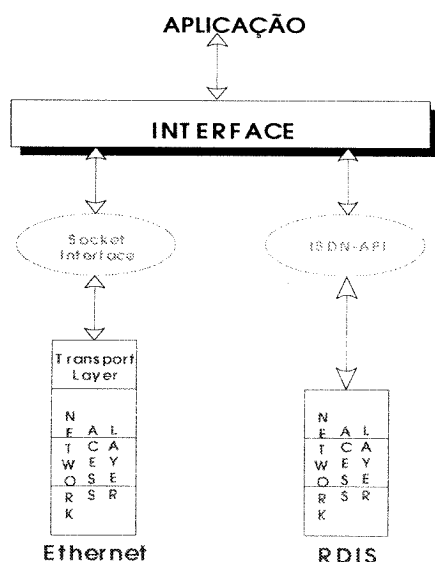


Fig. 1: Uniformização da interface de programação de aplicações para RDIS e TCP/IP.

Este projecto foi desenvolvido na linguagem de programação C em ambiente MS-DOS, tendo sido utilizadas as bibliotecas para Internet (Socket Library) e para RDIS (ISDN-API), existentes no DETUA.

II. DESCRIÇÃO DA INTERFACE

A fase inicial deste projecto consistiu na definição de um conjunto de rotinas básicas que constituirão a interface, e que reflectem no fundo as diferentes etapas a ter em consideração em qualquer processo de comunicação, ou seja, estabelecer e terminar uma ligação, transmitir/receber a informação pretendida e processar os dados recebidos. Para além disto é necessário considerar ainda uma rotina responsável pela inicialização das estruturas de dados a usar na comunicação e que inicie o processo de escuta do canal de comunicação e uma outra, que liberte os eventuais recursos usados pela aplicação para aceder à rede. As rotinas constituintes da interface são:

int init_api();

Esta função deverá ser sempre chamada no início de qualquer aplicação, dado que é responsável por todas as inicializações necessárias ao processo de comunicação em causa, como, por exemplo, a inicialização e alocação de memória para as estruturas suporte dos blocos de dados a enviar ou receber pela rede. Se a inicialização tiver sucesso, esta rotina devolve o valor zero.

*int estabelece_ligacao(char *numero);*

Quando um dos terminais envolvidos na comunicação pretende iniciar uma ligação, chama esta rotina, passando-lhe um ponteiro para o número (ou nome) da máquina remota. A aplicação a desenvolver terá conhecimento do sucesso ou falha desta operação, consoante o valor devolvido (zero ou um valor diferente, respectivamente).

*void (*ligacao_estabelecida)();*

Ponteiro para uma função, que é chamada sempre que os terminais envolvidos na comunicação detectam o início de uma ligação. A sua inicialização e declaração deverá ser sempre efectuada nas aplicações a desenvolver. Por seu

* Trabalho realizado no âmbito da disciplina de Projecto.

lado, o desenvolvimento do código da função *ligacao_estabelecida()* é da responsabilidade do autor da aplicação, podendo assim ter sobre seu controlo o estado actual da comunicação.

int termina_ligacao();

Esta rotina termina a ligação existente, devolvendo zero em caso de sucesso.

*void (*ligacao_terminada)();*

Ponteiro para uma função, chamado sempre que uma ligação é terminada. Tal como no caso do ponteiro *void(*ligacao_estabelecida)()*, a sua inicialização e declaração deverá ser feita na aplicação, assim como o desenvolvimento do código da função em causa.

void end_api();

Rotina responsável pela libertação dos recursos usados pela aplicação, devendo ser chamada apenas no final do programa principal da aplicação.

*int transmit(unsigned char far *buffer,int nbytes);*

Rotina responsável pela transmissão da informação. Quando a aplicação pretender transmitir um pacote de dados, deverá chamar esta função, passando-lhe como parâmetros um ponteiro (*buffer*) para esses dados e o número de bytes a enviar (*nbytes*). O valor de retorno é zero quando a transmissão é feita com sucesso.

*int receive_transmission(unsigned char far*pt);*

Esta função é a responsável pela recepção de todas as mensagens de dados recebidas pela aplicação (no caso da RDIS, para além destas são ainda recebidas mensagens de controlo). Outra função desta rotina é detectar os pedidos de estabelecimento e terminação de ligação efectuadas pelo terminal remoto. Trata-se de uma rotina cujo papel de "escuta" do "canal" de comunicação é fundamental em qualquer processo de comunicação donde, a sua chamada por qualquer aplicação deverá ser efectuada à frequência o mais elevada possível, por forma a obter a máxima eficácia. Detectando uma mensagem de dados, esta função copia-a para o ponteiro *pt* e devolve o número de bytes respectivo, informando assim a aplicação que existem dados a processar. Quando tal não acontece o valor de retorno é menor ou igual do que zero.

*void (*atende_request)();*

Verificou-se que surgia por vezes a necessidade de processar imediatamente as mensagens recebidas antes de enviar novos dados para a rede. Como esse processamento deverá ser efectuado a mais alto nível pela aplicação, foi criado este ponteiro, que é o responsável pela chamada da função posteriormente criada para processamento dos

dados. Portanto, este ponteiro deverá ser inicializado pela aplicação.

São ainda usadas pela interface as seguintes três variáveis globais:

int ligacao: variável que detém a informação sobre o estado da ligação (o seu valor inicial, i.e., quando não existe ligação, deverá ser zero);

int BLOCKSIZE: tamanho máximo da frame de dados a transmitir;

*char *hostname:* ponteiro para o nome (ou número) da máquina remota que estabeleceu a ligação;

Definidas estas rotinas básicas passou-se à sua implementação para cada uma das redes em causa usando as API's respectivas e tendo em conta as especificações protocolares de cada uma delas.

As principais diferenças a nível protocolar e as consequentes implicações em termos da implementação das rotinas atrás mencionadas, serão referidas de forma sumária em seguida.

III. ACESSO À RDIS

A API para a RDIS serve de interface entre o software do utilizador e o software responsável pelo protocolo de acesso à rede (DSS1 - Digital Subscriber Signalling System Nº 1). O acesso físico à RDIS é realizada através da placa PCBIT (desenvolvida pelo INESC) que permite para além da comunicação entre PC's o estabelecimento de chamadas de voz através de um dos canais B.

Através deste pacote de software é então possível escolher quais os protocolos de nível 2 e 3 do plano de utilizador OSI que se pretendem usar em cada caso específico. Este trabalho foi realizado usando a interface de acesso primário (2B+D) estabelecendo uma ligação através de um dos canais B (64Kbits/s) entre os dois terminais remotos. O canal D (16 Kbits/s) funciona como canal de sinalização tendo sido implementados neste plano de controlo os protocolos LAPD e I.451/Q.921 (níveis 2 e 3 do plano OSI, respectivamente) descritos nas recomendações do CCITT.

A API vai portanto estabelecer a ligação entre o protocolo DSS1 e as "aplicações", de modo que estas possam aceder aos canais B e alterar certos parâmetros das mensagens de sinalização para controlar os serviços implementados, definirem compatibilidades e escolherem o endereço do interlocutor desejado.

Estas mensagens trocadas entre a API e o software situado no nível imediatamente superior têm a seguinte estrutura:

HEADER	DADOS da MENSAGEM
--------	-------------------

Por sua vez o HEADER destas mensagens é constituído pelos seguintes campos:

Comprimento: comprimento total da mensagem em bytes
Identificador: Identificador da aplicação (único e

exclusivo)

Comando: Comando a enviar

Nº da mensagem: número sequencial das mensagens segundo o seguinte critério:

0x0000 a 0x7FFF - mensagens provenientes da aplicação

0x8000 a 0xFFFF - mensagens provenientes da rede

Sub-Comando: identifica o tipo de mensagem (_REQ, _CONF, _IND, _RESP)

Este último campo da mensagem mais não é do que a codificação das 4 primitivas de comunicação entre camadas adjacentes do modelo OSI (fig. 2).

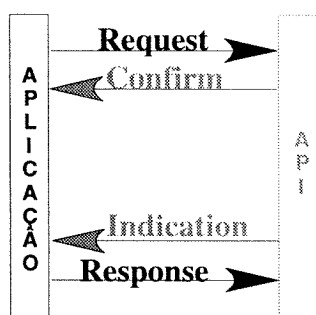


Fig. 2: Primitivas de comunicação entre a interface e a API.

Assim, quando a aplicação (neste caso a camada N+1) pretende requerer um serviço à API (camada N), gera a primitiva REQUEST, obtendo a confirmação da recepção da mesma pela API recebendo uma mensagem do tipo CONFIRM. Por outro lado, quando a camada inferior pretende informar a aplicação do fornecimento dum dado serviço, fá-lo gerando a primitiva INDICATION, confirmando a sua recepção ao receber uma mensagem cujo sub_comando é RESPONSE.

O conjunto de mensagens disponível é numeroso, permitindo aceder a todo o tipo de serviços fornecidos pela RDIS. Neste projecto, no entanto, além de o recurso à maior parte desses serviços não ser necessário, pretende-se que os protocolos envolvidos se tornem transparentes ao futuro desenvolvimento de aplicações sobre a interface aqui definida. Assim, todo o processamento destas primitivas é feito ao nível do ficheiro onde são implementadas as rotinas básicas definidas para a interface criada.

int init_api():

Esta rotina, responsável pela inicialização da API, assegura as seguintes funções:

- Alocação de memória para as mensagens (de controlo e de dados) entre a API e a aplicação;
- Geração do *interrupt* de software usado pela PCBIT;
- Registo da aplicação na API. São aqui definidos, entre outros, o número máximo de ligações de nível 3 a suportar (uma) e o comprimento máximo (em bytes) de um bloco de dados;
- Colocação da API num estado que permita receber e

aceitar uma ligação de dados a qualquer momento. A API é ainda informada do tipo de chamadas que a aplicação aceita. A selecção das mesmas por parte da API será feita pela análise de compatibilidade através da codificação do elemento de informação "Bearer Capability" (BC). Neste caso, os atributos de serviço de suporte definidos foram:

- Possibilidade de Transferência: Informação digital sem restrições;
- Modo de Transferência: Modo circuito;
- Ritmo de Transferência: 64 Kbits/s;

*int estabelece_ligacao(char *numero):*

Esta é a rotina responsável pelo estabelecimento de uma ligação através de um canal B de 64 Kbits/s entre dois terminais RDIS. O estabelecimento de uma ligação deste tipo implica a troca de várias mensagens entre as aplicações existentes nos dois terminais e a API, e entre esta e a rede.

Assim, o terminal que pretende estabelecer a ligação envia uma mensagem CONNECT_REQ para a API. Na estrutura da mensagem enviada salientem-se aqui alguns dos campos que contêm várias informações importantes:

IECallingPartyNumber/IECallingPartySubAddress- Número e sub-endereço do chamador;

IECalledPartyNumber/IECalledPartySubAddress- Número e sub-endereço do chamado;

IEBearerCapability- Codificação do elemento de informação "Bearer Capability";

IEChannelIdentification - Canal de comunicação preferido (neste caso B1);

A aplicação remota recebendo um CONNECT_IND indicando a existência de uma chamada de entrada, inicia a conexão enviando um CONNECT_ACTIVE_REQ. Depois de estabelecida a ligação via canal B, i.e., depois de recebidas as mensagens de CONNECT_ACTIVE_IND por parte do terminal chamador e de CONNECT_ACTIVE_CONF por parte do chamado, são definidos os protocolos de nível 2 e 3 e "link" de nível 2 a usar com o protocolo de nível 3 escolhido (caso algum seja escolhido). A selecção dos protocolos de canal B é feita enviando a mensagem SELECT_PROTOCOL_REQ, cujos campos mais importantes são:

layer2Protocol: Definição do protocolo de nível 2 (neste caso LAPD);

layer2MaxDataSize: Dimensão máxima das tramas de nível 2 (BLOCKSIZE);

layer2END1: SAPI (Service Access Point Identifier) a usar no "link", identificando assim o ponto de acesso do serviço da camada 2 e indicando o tipo de informação transportado na trama.

layer2END2: TEI (Terminal Endpoint Identifier) a usar, identificando assim o terminal.

layer3Protocol: Protocolo de nível 3 a definir entre X.25, ISO 8208 ou transparente. Definiu-se que o acesso pelas aplicações ao canal B será efectuado de modo transparente;

Confirmada a selecção destes protocolos por parte da API (SELECT_PROTOCOL_CONF), é enviada a mensagem ESTABLISH_LAPD_REQ pedindo à API a activação do "link", ou seja, a sua passagem ao estado que permite o envio e recepção de tramas de informação numeradas com confirmação. Em termos de interacção API-Rede (nível 2), é enviada para esta última uma trama não numerada com o comando SABME (Set Asynchronous Balanced Mode Extended), sendo a confirmação da activação do "link" obtida após a recepção da resposta UA (Unnumbered Acknowledge).

Depois de recebida a mensagem ESTABLISH_LAPD_CONF por parte do terminal chamador e enviada pelo terminal remoto um ESTABLISH_LAPD_RESP, é chamado o ponteiro para a função *ligacao_estabelecida()*.

int receive_transmission (unsigned char far pt):*

Esta rotina, sempre que é chamada, pede uma mensagem à API, verificando a validade da mensagem recebida. Neste último caso são analisados os campos de comandos e sub-comandos recebidos. Caso se trate de mensagens de dados enviados pela máquina remota (DATA_LAPD_IND), copia para a memória esses dados e envia para a API um DATA_LAPD_RESP a confirmar a recepção. Neste caso, o valor de retorno desta função é o número de bytes recebidos. Desta forma, sempre que esta última devolve um valor maior do que zero, a aplicação deverá processar imediatamente estes dados existentes no buffer *pt*. Caso a mensagem recebida seja de controlo (ex. CONNECT_IND), o seu processamento é efectuado a este nível, tornando-o assim transparente ao utilizador final desta interface.

int transmit (unsigned char far buffer, int n_bytes):*

Quando uma aplicação pretender transmitir informação através da rede deverá, chamar esta rotina, passando-lhe um ponteiro para os dados e o número de bytes que pretende transmitir.

Devido às limitações impostas pela própria API, o algoritmo usado para implementar esta rotina é relativamente complexo. Quando é enviada uma mensagem para a API, esta é colocada internamente numa fila de espera até que a respectiva confirmação seja obtida.

Este procedimento é idêntico para todas as mensagens, incluindo a usada para enviar dados para a rede (DATA_LAPD_REQ) em que um dos seus campos (*dwData*) contém o ponteiro para o bloco de dados a transmitir. Quer isto dizer que o facto de a mensagem de DATA_LAPD_REQ ser armazenada pela API, não implica o armazenamento dos respectivos dados, mas apenas a informação do seu endereço na memória. Este tipo de funcionamento interno da API leva a que a aplicação não poderá alterar o conteúdo desse ponteiro (e consequentemente enviar novos dados para a rede) até que chegue a confirmação relativa à mensagem previamente enviada. Ora, dado que a velocidade de processamento de

qualquer PC é muito maior do que a taxa de transmissão da rede (64 Kbits/s), esta rotina *transmit* implementa um mecanismo de gestão de ponteiros para os dados a transmitir baseado numa estrutura do tipo FIFO (First In First Out) que permite armazenar até 5 blocos de dados na memória, independentemente da confirmação por parte da API. Assim, sempre que são passados os bytes que se pretendem transmitir (*buffer*) para dentro desta rotina, pela aplicação implementada a um nível superior, são copiados para a área de memória imediatamente a seguir aos enviados anteriormente. O ponteiro para a zona da FIFO onde se encontram esses dados é então passado para a mensagem DATA_LAPD_REQ. O que vai acontecer é que a API vai ter várias mensagens de DATA_LAPD_REQ na fila de espera, cada uma delas com o campo *dwData* a apontar para diferentes endereços de memória.

Por forma a garantir uma gestão correcta da estrutura criada, existe a variável global *b_p_c* (blocos por confirmar), que é incrementada sempre que um bloco de dados é enviado, e decrementada quando uma mensagem DATA_LAPD_CONF, enviada pela rede confirmando a recepção dos dados previamente transmitidos, é detectada.

int termina_ligacao():

Esta rotina deverá ser chamada pelo terminal envolvido na comunicação e que a pretende terminar. É enviada para a API uma mensagem de DISCONNECT_REQ identificando a chamada que se pretende desligar. Quando a API, enviar para a aplicação que iniciou o desligamento da chamada a mensagem DISCONNECT_CONF, ou DISCONNECT_IND para a aplicação remota, as aplicações enviam um LISTEN_REQ, colocando-se novamente em posição de aceitar novas chamadas, e chamam o ponteiro para a função *ligacao_terminada()*.

void end_api():

A aplicação termina a sua interacção com a API, sendo libertada a área de memória entregue por aquela à API quando do registo da aplicação, para gestão por parte desta.

IV. ACESSO A TCP/IP

A *Socket Interface* permite aceder aos níveis 1 a 4 do modelo OSI através de um conjunto de primitivas, independentemente do interface de acesso à rede. Com esta biblioteca de funções é possível estabelecer uma comunicação entre dois PC's recorrendo ao conceito abstracto de *socket*, que é um "*end point*" da comunicação donde é enviada e recebida a informação. Quando da sua criação, um "*socket descriptor*" é associado à *socket* de forma a poder ser referenciada posteriormente pelas aplicações.

Qualquer programa de aplicação interage com esta biblioteca de funções recorrendo a "*system calls*", das quais se destacam as seguintes, usadas neste trabalho:

socket(): criação da *socket*;
 bind(): associa o endereço à *socket* previamente criada;
 listen(): cria um fila de espera para receber pedidos de acesso;
 connect(): inicia a conexão com uma *socket* remota;
 accept(): remove um pedido de conexão da fila iniciando a conexão;
 send(): envia uma mensagem através de uma dada *socket*;
 recv(): recebe uma mensagem numa dada *socket*;
 close(): termina a existência de uma *socket*;

De uma forma geral, as aplicações construídas sobre esta interface baseiam-se no modelo de comunicação *Client-Server* em que o "cliente" requisita um serviço ao *Server* e aguarda a resposta a esse pedido. O "servidor" é um programa de aplicação que oferece um dado serviço que pode ser acedido através da rede. A sequência de chamada das "*system calls*", para estabelecer uma ligação de dados, depende fundamentalmente de duas questões: se se trata de um programa "cliente" ou "servidor" e qual o modo de comunicação (*connection-oriented* ou *connectionless*).

Este tipo de estrutura de comunicação em que apenas uma das partes envolvidas ("cliente") pode iniciar uma comunicação, enquanto a outra tem um papel perfeitamente passivo limitando-se a atender os pedidos de serviço requeridos, é claramente contrária à filosofia imposta inicialmente para as aplicações a desenvolver sobre a interface uniformizada que se pretende criar, e onde se prevê uma completa bidireccionalidade entre os dois programas (no caso presente pretende-se que o programa a correr nas duas máquinas ligadas à rede seja o mesmo, enquanto que usando o modelo acima referido, teremos dois programas distintos nas máquinas em comunicação).

Outra característica fundamental que foi necessário introduzir neste trabalho diz respeito ao modo de funcionamento (Bloqueado ou Não Bloqueado) da própria *socket* criada. Caso se tivesse optado pelo primeiro, ao invocar a *system call* *accept()*, o programa fica bloqueado até que a operação requerida esteja completada, i.e., até chegar um pedido de conexão e esta for aceite. O mesmo acontece aliás com a função *send()* que termina a sua execução só depois de enviar todos os bytes de informação que lhe são passados. Para que isto não aconteça e seja possível que qualquer aplicação a desenvolver sobre a interface criada neste trabalho possa, tanto estabelecer uma comunicação, como aceitar um pedido remoto de conexão, é necessário marcar a *socket* no modo não bloqueado, invocada a *system call* *fcntl()*. Desta forma, as funções atrás descritas terminam o seu processamento após um intervalo de tempo mínimo, independentemente de terem ou não efectuado a sua tarefa específica, retornando o controlo ao programa que as invocou.

int init_api();

Os passos fundamentais para a implementação desta

rotina foram os seguintes:

- Para suporte dos blocos de dados foram criados três ponteiros globais para os quais são alocados dinamicamente (*BLOCKSIZE+2*) bytes. A razão da necessidade destes 2 bytes "extra" prende-se com a especificação de um protocolo implementado a este nível que permite garantir uma correcta separação dos blocos de dados quando da sua recepção.

- Criação da *listen socket*, que será responsável pela recepção de um pedido de conexão remoto, especificando a família de protocolos a usar com a *socket* (*AF_INET* - família *DARPA Internet*), e o tipo de comunicação desejado, escolhendo o tipo de *socket* apropriado (*SOCK_STREAM*).

- Associação de um *port* à *socket* através da chamada da *system call* *bind*. Com este procedimento, o sistema operativo passa a saber para onde dirigir todas as mensagens recebidas da rede, destinadas a esta *socket*.

- Após as propriedades da *socket* serem alteradas impondo-lhe o modo *não bloqueado*, a chamada de *listen*, indica que aquela está preparada para receber qualquer pedido de conexão que lhe seja destinado. Como se pretende que a verificação da existência de algum pedido, seja feita de tempos a tempos, de forma a que o programa não bloqueie neste ponto e possa executar outras tarefas, a sua chamada será efectuada na rotina *receive_transmission*.

*int estabelece_ligacao(char *numero);*

Antes de se passar à descrição da forma como foi implementada esta rotina, convém referir que, ao contrário do que acontece no ambiente de comunicação *RDIS*, o estabelecimento de uma *ligação* numa rede *Ethernet* é puramente lógica. A necessidade de manter uma coerência com a filosofia imposta para a interface definida inicialmente, i.e., para que, do ponto de vista das aplicações a desenvolver sobre a mesma, a *ligação* se comporte como se tivesse um carácter físico, justifica a necessidade da implementação desta rotina sobre a *Socket Interface*.

Após a criação de uma *socket* cuja função é "escutar" os pedidos de conexão, a aplicação que pretende iniciar uma *ligação* chama esta rotina, passando-lhe o nome ou número da máquina remota.

Ao nível desta rotina o que acontece é o seguinte:

- Criação de uma nova *socket* que inicia uma conexão com a *socket* remota efectuando a chamada de *connect*. A chamada desta *system call* implica a associação desta *socket* ao endereço remoto indicado. Em termos locais e, dado que não foi chamada a *system call* *bind()*, a associação é efectuada automaticamente pelo sistema operativo.

- Após o estabelecimento da *ligação*, i.e., depois de o procedimento descrito anteriormente ter sido efectuado sem que nenhuma situação de erro tivesse sido detectada, é chamado o ponteiro para a função *ligacao_estabelecida()*.

Em termos da máquina remota, os procedimentos necessários ao estabelecer desta ligação serão fundamentados quando da descrição da rotina seguinte, já que foram aí implementados.

int receive_transmission (unsigned char far pt):*

Esta rotina começa por verificar se no momento da sua chamada já existe ou não ligação. Neste último caso é da sua responsabilidade verificar se chegou algum pedido de conexão à *listen socket* através da chamada de *accept*. Ao detectar esse pedido, esta *system call* cria uma nova *socket* com as mesmas propriedades da *listen socket* que vai ser de facto a que estará envolvida na futura troca de informação. Note-se aqui a importância de "marcar" a *socket* no modo não bloqueado pois, se tal não tivesse sido feito, o programa ficaria aqui bloqueado até que chegasse um pedido de conexão.

Antes de devolver o controlo ao programa principal, esta rotina chama o ponteiro para *ligacao_estabelecida()*.

No caso de já existir ligação, esta função é responsável pela recepção dos bytes de informação enviados pela máquina remota (invocando a *system call* *recv()*) e posterior cópia dos mesmos para a posição de memória apontada por *pt*. É também da responsabilidade desta rotina, detectar o envio de uma mensagem específica por parte da máquina remota, indicando que pretende terminar a ligação existente.

int transmit (unsigned char far buffer, int n_bytes):*

Foi possível verificar que, se se transmitissem para a rede dois blocos de dados cuja soma fosse menor que *BLOCKSIZE* através da chamada desta rotina, acontecia por vezes que estes chegavam ao terminal receptor num mesmo bloco. Quer isto dizer que este protocolo garante a transmissão integral do número de bytes pretendidos e a sua sequência, mas não a separabilidade dos mesmos, como acontecia com o protocolo LAPD no ambiente de comunicação RDIS. De forma a permitir a uniformização da interface criada, esta rotina implementa um protocolo, completamente transparente para o programador que faça uso da mesma, que garante esse sequenciamento das tramas a enviar para a rede. A forma mais expedita de o fazer é colocar o número de bytes a transmitir (passados como parâmetro para esta rotina) nos dois primeiros bytes do pacote de dados a enviar. Quando da recepção, a análise destes bytes iniciais recebidos permitirá saber o comprimento exacto do pacote que é necessário processar.

O envio da informação para a rede é efectuado pela *system call* *send* que, em caso de sucesso, retorna o número de bytes efectivamente transmitidos. Quando a *socket* funciona no modo não bloqueado, como é o caso, esta função não garante a transmissão de todos os bytes passados para o ponteiro *buffer* donde, houve necessidade de efectuar a sua chamada dentro de um "loop", garantindo assim que a rotina *transmit* só termina o seu processamento depois de todos os bytes terem sido

enviados.

Outra particularidade desta *system call*, directamente relacionada com as especificações do protocolo TCP, é a impossibilidade de enviar novos dados para a rede sem que o terminal remoto tenha enviado o "acknowledge" do pacote enviado anteriormente, o que só acontece depois de, do lado do receptor, ser efectuado um *recv()*. A detecção deste tipo de situação é aqui efectuada verificando a devolução, por parte daquela, da constante de erro *EWouldBlock*. Este facto poderia levar a que, por exemplo, os dois terminais tivessem enviado informação para a rede mais ou menos ao mesmo tempo, ficando os dois à espera da respectiva confirmação, originando um bloqueio simultâneo nas duas máquinas. Por forma a evitar esta situação de todo indesejável, é necessário efectuar a chamada da função *receive_transmission* dentro desta rotina, sempre que *EWouldBlock* é detectada de forma a processar imediatamente os eventuais dados recebidos.

int termina_ligacao():

Tal como foi definido inicialmente pretende-se que, através desta rotina, qualquer das máquinas envolvidas na comunicação possa terminar uma ligação e que a outra tome conhecimento desse facto. Mais uma vez as diferenças entre os protocolos envolvidos nos dois ambientes de comunicação obrigou a uma abordagem diferente para a implementação desta rotina. Enquanto no caso da RDIS o protocolo de nível 3 de controlo de chamadas garantia que, iniciado o processo de desligamento de uma chamada, ambos os terminais envolvidos na comunicação tivessem conhecimento desse facto, neste caso em que a ligação é efectuada apenas ao nível lógico, tal não acontece. Assim, no protocolo TCP, quando um dos terminais efectua o *close()* da sua *socket*, apenas termina a ligação do ponto de vista do seu terminal, não informando o terminal remoto desse facto, i.e., a conexão é *apagada* apenas quando efectuada em ambos os lados. A forma encontrada para resolver este problema foi "forçar" o terminal que pretende desligar a ligação a informar o terminal remoto desse facto, enviando uma mensagem específica (ex. "end"). Só após este procedimento é invocada a *system call* *close()* e actualizado o "estado" da ligação através da chamada do ponteiro para *ligacao_terminada()*.

void end_api():

Esta rotina efectua o *close()* da *listen socket* e liberta a memória alocada para as estruturas usadas para os blocos de dados.

V. CONCLUSÃO

A interface criada foi testada criando uma aplicação que permite a transferência de ficheiros entre dois PC's usando as rotinas constituintes da mesma. O programa criado pode ser usado tanto na rede Ethernet como na

RDIS sem que haja necessidade de efectuar alterações no seu código, bastando apenas recompilá-lo com o ficheiro onde se encontram implementadas as rotinas básicas para a estrutura de comunicação em causa, sobre a respectiva livreria.

Com a implementação desta Interface torna-se possível o desenvolvimento de aplicações para comunicações de dados entre dois PC's independentemente da estrutura de

comunicação que lhe servirá de suporte. Para além desta garantia de compatibilidade com o meio de comunicação, a interface criada permite que o projectista desenvolva o seu software sem ter de se preocupar com os procedimentos de mais baixo nível necessários para efectuar a comunicação, acedendo de uma forma transparente aos protocolos implementados pelas camadas mais baixas do modelo OSI.

Sistema Integrado de Tele-imagiologia

Luis F. C. Figueiredo, Fernando M. S. Ramos, António Sousa Pereira

Resumo - Este artigo pretende, de uma forma resumida, apresentar um trabalho desenvolvido no INESC Aveiro, na área da Tele-medicina. Esse trabalho, que é um Sistema de Tele-imagiologia suportado na RDIS, abrange diversas áreas multidisciplinares, que vão desde o Processamento Digital de Imagem (com especial destaque para as técnicas de Compressão e Descompressão de Imagem), a sistemas de Comunicação de Dados e ao desenvolvimento de Interfaces Gráficas Homem/Máquina.

Abstract - This paper presents a Tele-radiology system supported in the ISDN, and includes discussion on several aspects of the system, such as digital imaging processing, image compression techniques, communications and human machine interfacing.

I. INTRODUÇÃO

O conceito de Teleradiologia ou mais genericamente o de Tele-imagiologia surgiu no início da década de 1980, com o aparecimento dos primeiros sistemas que permitiam a transmissão de imagens médicas digitais, através de uma rede de comunicação de dados, de um local para outro. Desde essa altura esses sistemas não pararam de evoluir, sendo essa evolução devida por um lado aos avanços

tecnológicos que se efectuaram na área dos computadores, cuja relação qualidade/preço cresceu substancialmente ao longo dos anos, bem como ao desenvolvimento de novas infraestruturas de comunicações que permitem cada vez mais a transmissão de grandes quantidades de informação em tempos cada vez menores e a custos mais reduzidos. Por outro lado tem vindo a crescer o interesse por parte dos utilizadores destas novas tecnologias (essencialmente Médicos Especialistas) que, desde o início, reconheceram as enormes vantagens que estes sistemas poderiam trazer na prestação dos cuidados de saúde ao público em geral.

Nos últimos anos foram desenvolvidos sistemas mais complexos que tendem a abranger todas as actividades desenvolvidas num Hospital desde a prestação de cuidados de saúde até à sua gestão e planeamento. Estes sistemas, habitualmente designados por "Image Management and Communication Systems" (IMACS) [2], possuem preços bastante elevados o que constitui um dos principais entraves à sua implementação não sendo no entanto o único: a falta de normalização que leva à existência de alguns *sistemas fechados* que não permitem uma fácil integração de novas aplicações e a inércia dos agentes responsáveis pela implementação destes sistemas, constituem outros dos entraves.

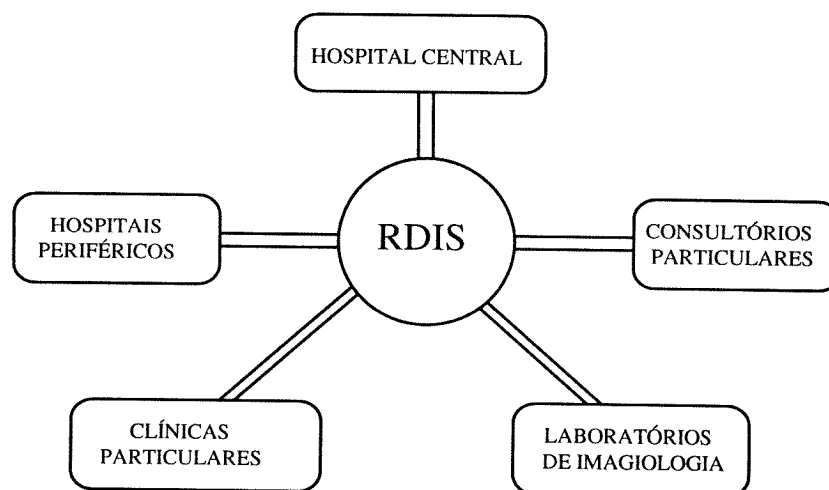


Fig. 1: Diagrama geral do sistema proposto.

II. DESCRIÇÃO DO SISTEMA PROPOSTO

Neste trabalho é proposto um sistema integrado de Tele-imagiologia, que pretende aliar os baixos custos dos equipamentos vulgarmente usados em sistemas de Tele-imagiologia [3], com algumas das facilidades existentes nos sistemas de PACS (Picture Archive and Communication Systems) [1], nomeadamente a possibilidade de acesso a bases de dados onde se encontra

toda a informação relevante, em termos médicos, de cada paciente.

Para tal, além de obedecer ao esquema tradicional dos sistemas de Tele-imagiologia apresentado na figura 1, este sistema pode também funcionar como um sistema interno de Imagiologia, usando como infraestrutura de comunicações uma rede Ethernet e o protocolo TCP/IP, tal como descrito na figura 2.

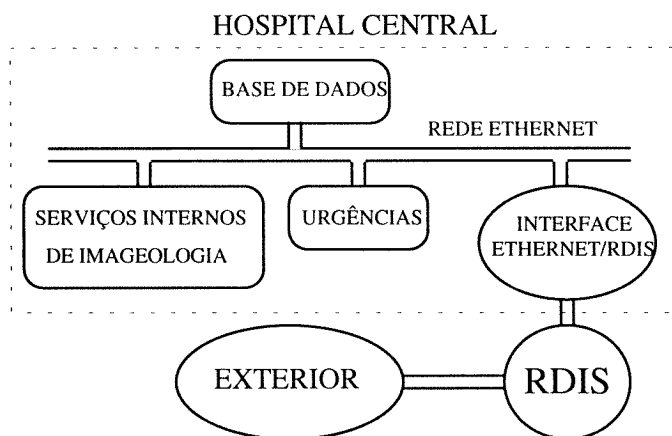


Fig. 2: Diagrama do sistema interno de Imagiologia proposto.

As facilidades oferecidas por este sistema vão desde a manipulação local de imagens médicas com a utilização de várias ferramentas de processamento digital de imagem que permitem aumentar o seu contraste, até à conferência entre especialistas situados em diferentes pontos, havendo aí a possibilidade de qualquer utilizador fazer o ZOOM de uma zona da imagem em *real time* ou utilizar o cursor do rato para apontar um qualquer detalhe. Quando em conferência, qualquer operação feita num lado é imediatamente, e de uma forma automática, reflectida no outro terminal. Permite também a manipulação de uma base de dados onde é armazenada a história clínica de cada um dos pacientes.

III. COMPRESSÃO DE IMAGEM

Nos últimos anos, com o avanço da tecnologia, e a banalização dos meios informáticos, o processamento digital de imagem começou a ser uma ferramenta bastante útil nos mais diversos ramos de actividade. Na área da saúde, em especial, o uso de imagens digitalizadas têm vindo progressivamente a ter mais importância, em detrimento das imagens analógicas. No entanto, o uso de imagens digitais traz consigo alguns problemas: para que não haja uma degradação significativa das imagens, quando se efectua a sua digitalização, esta tem que ter não só uma boa resolução espacial, mas também uma boa resolução tonal. Testes efectuados por médicos

especialistas em imagens radiológicas, apontam para resoluções espaciais da ordem de 2048*2048 pontos e resoluções tonais de 12 bits por pixel, como sendo o ideal. Com estes valores, o número de bytes por cada imagem é demasiado grande, quer para o seu armazenamento quer, sobretudo, para o seu transporte. Surge assim a necessidade de se efectuar a compressão digital de imagem.

A escolha do método de compressão a utilizar numa aplicação concreta, depende de um conjunto de factores, alguns dos quais contraditórios entre si. A grande dificuldade consiste em estabelecer um compromisso aceitável entre esses factores. Esta tarefa é sempre discutível, uma vez que o peso de cada factor depende não só da pessoa que o avalia, como também da altura em que essa avaliação é feita. Isto significa que uma escolha feita neste momento, pode com o passar do tempo e a consequente evolução da tecnologia, deixar de ser a melhor opção. Apresentam-se seguidamente alguns dos factores que deverão ser tidos em conta na escolha do método a usar.

- Compressão com ou sem perda de informação.
- Relação Sinal/Ruído aceitável no caso de compressão com perda de informação.
- Taxas de compressão desejadas.
- Tempo de compressão admissível.
- Utilização ou não de Hardware específico.

No caso concreto do sistema de Tele-imagiologia apresentado neste artigo, optou-se, inicialmente, por desenvolver um método de compressão sem perda de informação, devido ao risco que poderia trazer o facto de se introduzirem erros nas imagens médicas. Numa segunda fase desenvolveram-se também métodos de compressão com perda de informação.

As taxas de compressão desejadas são as maiores possíveis, como se pretende em qualquer método. O tempo de compressão deverá ser o menor possível, uma vez que o sistema de compressão deverá ser usado durante a fase de transmissão de imagens via RDIS [7], tendo em vista a

diminuição do tempo de transmissão com a consequente diminuição dos custos a ela associados. A diminuição do tempo de transmissão só poderá ser conseguido, se o tempo de compressão for inferior ao tempo de transmissão da imagem não comprimida.

Em termos de Hardware e por razões de custo, optou-se por não usar placas de compressão específicas, pelo que todo o software foi implementado numa máquina baseada num processador 80386 ou 80486.

O método de compressão desenvolvido consistiu na utilização de um Estimador, um Quantificador e uma Codificador de códigos de comprimento variável, tal como mostra o esquema da figura 3.

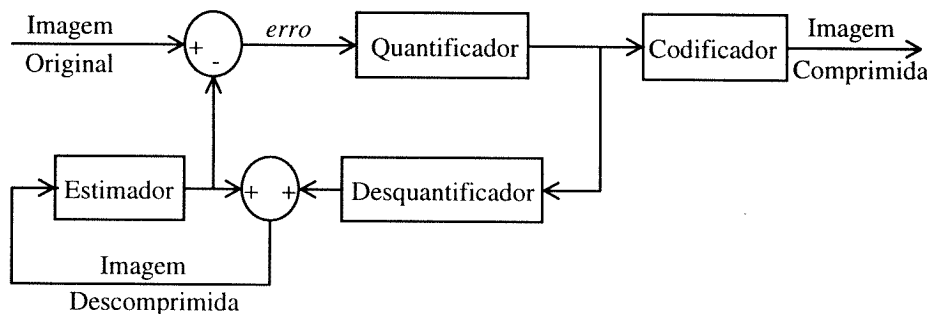


Fig. 3: Diagrama do Compressor de Imagem

O Estimador escolhido mostrou possuir um bom compromisso entre os resultados obtidos e a complexidade de cálculo. Considerando os pixels do esquema da figura 4, a formula do Estimador é a seguinte:

$$P = (A+B+C+1)/3 + ((B-C)+(A-C)+(A-D)+1)/3$$

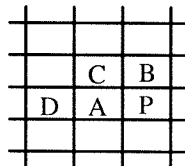


Fig. 4: Distribuição espacial dos pixels usados pelo Estimador

O primeiro termo desta expressão constituído por $(A+B+C+1)/3$ é a média aritmética dos 3 pixels mais próximos do pixel a prever (P). Este termo é justificado pelo facto de existir uma grande correlação entre os pixels vizinhos nas imagens médicas. O segundo termo $((B-C)+(A-C)+(A-D)+1)/3$ foi introduzido por se verificar que os aumentos ou diminuições nos níveis de cinzentos (derivadas) ao longo de uma linha vertical ou horizontal da imagem, eram relativamente constantes. Após a sua simplificação, esta formula resulta na seguinte expressão:

$$P = (3A+2B-C-D+1)/3$$

Este Estimador é usado para cada pixel da imagem que não se encontre nas suas extremidades. No caso dos pixels

a estimar, pertencerem à primeira linha ou à primeira e segunda coluna da imagem, são utilizados estimadores que apenas usam os pixels que se encontram à esquerda ou acima dos referidos pixels. O gráfico da figura 5 mostra o número de ocorrências de cada *erro* (diferença entre o valor verdadeiro e o valor estimado), numa imagem radiológica com 512*512 pixels. O Quantificador permite agrupar vários *erros* num único símbolo. Desta forma pode-se controlar o erro máximo admitido em cada pixel. Para comprimir sem perda de informação, basta fazer com que o majorante do erro por pixel seja zero. O Codificador permite codificar cada símbolo através de uma variante do código de Huffman [6] em que os símbolos mais prováveis são codificados com comprimentos menores e os símbolos menos prováveis são codificados com comprimentos maiores. Nesta variante são apenas codificados os símbolos cujo valor absoluto seja menor ou igual a onze. Os restantes são codificados através de um símbolo especial seguido do verdadeiro valor do pixel a codificar. Na figura 6 são apresentados os resultados comparativos da aplicação de diversos métodos de Compressão sem perda de informação a um conjunto de imagens radiológicas com 512*512 pixels. O primeiro (LZW) é o método de compressão usado nas imagens GIF. O segundo (Planos) consiste na divisão da imagem em vários planos binários (tantos quantos o número de bits por pixel) e aplicação do algoritmo anterior aos planos mais significativos. No terceiro método foi usado um algoritmo proposto pela norma JPEG [4] para a compressão sem perda de informação. Por fim, o algoritmo proprietário utiliza as técnicas já descritas anteriormente.

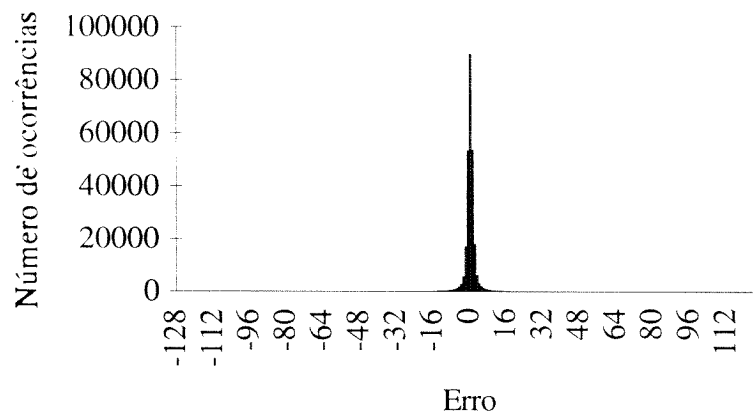
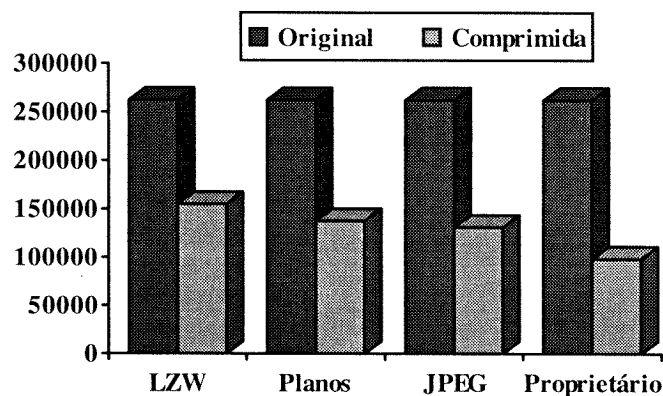


Fig. 5: Número de ocorrências de cada erro.



	LZW	Planos	JPEG	Proprietário
Original	262166	262166	262166	262166
Comprimida	154859	137500	131029	98493
TC	1.69	1.91	2.00	2.66

Fig. 6: Comparação dos diferentes métodos de compressão sem perda de informação.

Em termos de compressão com perda de informação, o grande objectivo foi a obtenção de taxas de compressão tanto quanto possível próximas das obtidas pelo uso da norma JPEG [5], com relações Sinal/Ruído semelhantes, mas com tempos de compressão e descompressão bastante inferiores. O método descrito anteriormente, mesmo com majorantes de erro por pixel de 3, tinha taxas de compressão da ordem dos 5 ou 6 que comparativamente com as obtidas pela aplicação da norma JPEG, que variam entre 30 e 33, eram bastante baixas. Desenvolveu-se uma

outra técnica, de implementação simples e execução bastante rápida, que quando aplicada em série com a anterior aumenta bastante a taxa de compressão à custa de uma ligeira descida na relação Sinal/Ruído. Esta técnica consiste na substituição de cada grupo de 2*2 pixels pela sua média aritmética. Por sua vez, na descompressão, além do processo inverso que consiste na geração de um bloco de 2*2 pixels por cada pixel da imagem comprimida, é feita uma filtragem espacial do tipo *Passa Baixo*, seguido de um ajustamento do valor dos 4 pixels de forma a que a

sua média seja igual à média dos pixels originais. A utilização desta filtragem e ajustamento permite aumentar a relação Sinal/Ruído em cerca de 3dB o que, em termos práticos, corresponde à diminuição do erro médio de cada pixel de 3 para 2, isto relativamente a uma imagem em que estes dois processamentos não sejam efectuados.

A conjugação deste método de compressão com o anterior permite criar diferentes modos de compressão que

vão desde a compressão sem perda de informação com taxas da ordem dos 2.66 (Modo 0), até taxas de 19.4 com relações Sinal/Ruído de 40.38dB (Modo 5).

A figura 7 mostra, de uma forma resumida, os resultados médios da utilização de cada um destes modos de compressão, bem como os resultados obtidos com o uso da norma JPEG.

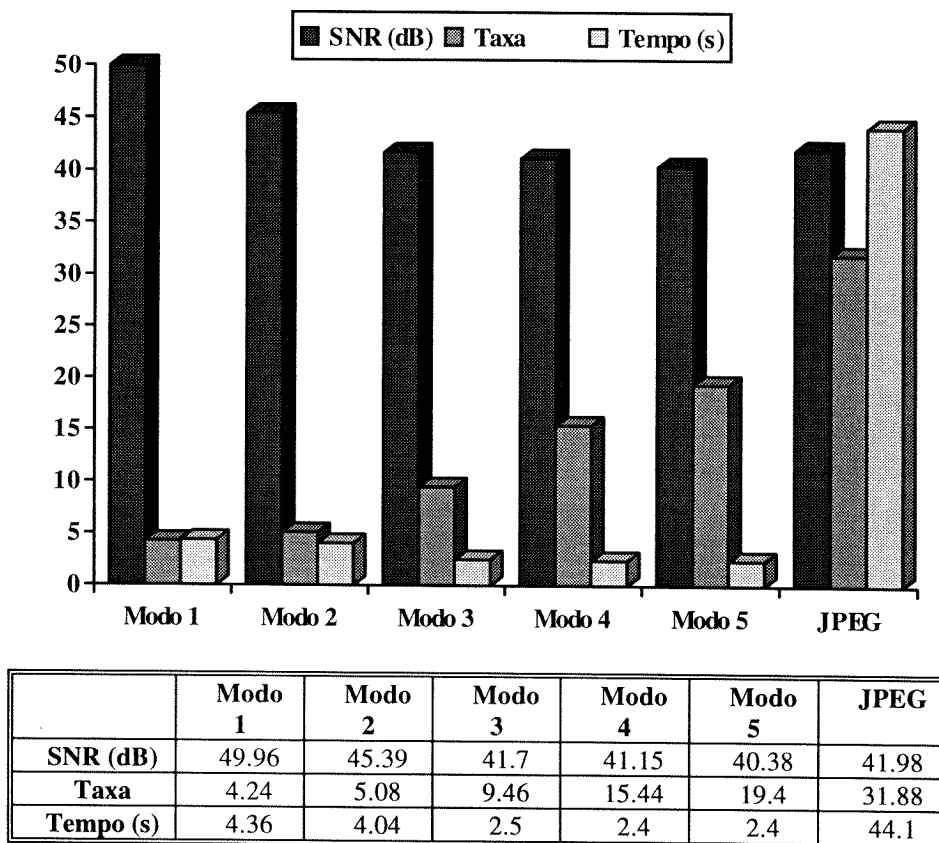


Fig. 7: Relação sinal/ruído, Taxa de Compressão e Tempo de Compressão dos diferentes Modos desenvolvidos.

A análise destes resultados em termos de relação sinal/ruído, mostra que os modos 1 e 2 possuem valores superiores a qualquer um dos outros modos, incluindo o algoritmo JPEG. Este facto, permite o uso de qualquer um destes métodos em aplicações que seja exigida uma qualidade de imagem superior à oferecida pela norma JPEG. Os Modos 4 e 5, possuem relações Sinal/Ruído ligeiramente inferiores em relação às obtidas pelo algoritmo JPEG. Em termos práticos, a diferença entre a qualidade das imagens resultantes destes dois modos de compressão, e das imagens resultantes do algoritmo JPEG é muito pequena, sendo muito difícil diferenciá-las.

A análise das taxas de compressão obtidas permite concluir que a norma JPEG é, sem dúvida alguma, a que possui melhores resultados.

Ao analisar o factor tempo verifica-se que os modos 1 a 5 possuem tempos de compressão substancialmente mais baixos do que o tempo obtido pelo algoritmo da norma JPEG.

IV. SUB-SISTEMA DE COMUNICAÇÕES

A estrutura de comunicações desenvolvida teve em conta a necessidade de se fazerem alterações frequentes quer ao nível lógico da rede uma vez que se estava a trabalhar com uma versão experimental da RDIS, quer ao nível físico dado que também se pretendia desenvolver uma aplicação usando a rede Ethernet como infraestrutura de comunicação. Desta forma definiu-se uma interface entre a aplicação e os drivers de rede tal como mostra a figura 8.

Esta interface permitiu desenvolver toda a aplicação de uma forma independente da estrutura lógica ou física da rede utilizada, aumentando assim a versatilidade da aplicação. A passagem da aplicação para a rede Ethernet resumiu-se à implementação das rotinas de interface Aplicação/Rede utilizando a Socket Library.

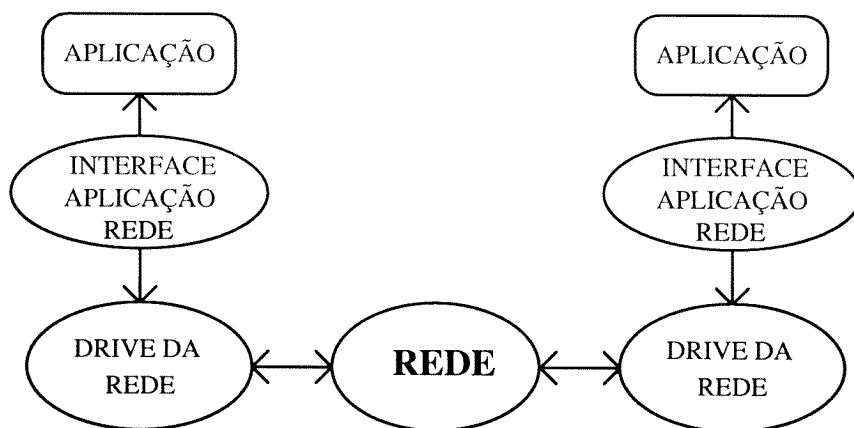
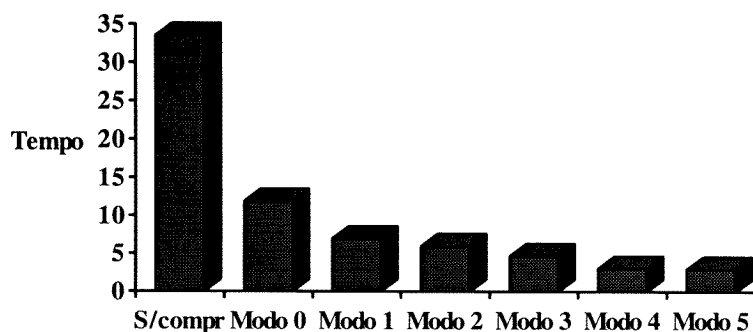


Fig. 8: Diagrama geral da estrutura de comunicações desenvolvida.

A velocidade de transmissão de imagens através da RDIS foi substancialmente aumentada pela integração do software de compressão com o software de comunicação. Esta integração consistiu na paralelização das tarefas de compressão e transmissão do lado do emissor e das tarefas de recepção e descompressão do lado de receptor. Como

resultado foi possível otimizar ao máximo o tempo de processamento do CPU.

O gráfico da figura 9 mostra as médias dos tempos de transmissão de imagens na RDIS, quando da aplicação dos diferentes métodos de compressão desenvolvidos.



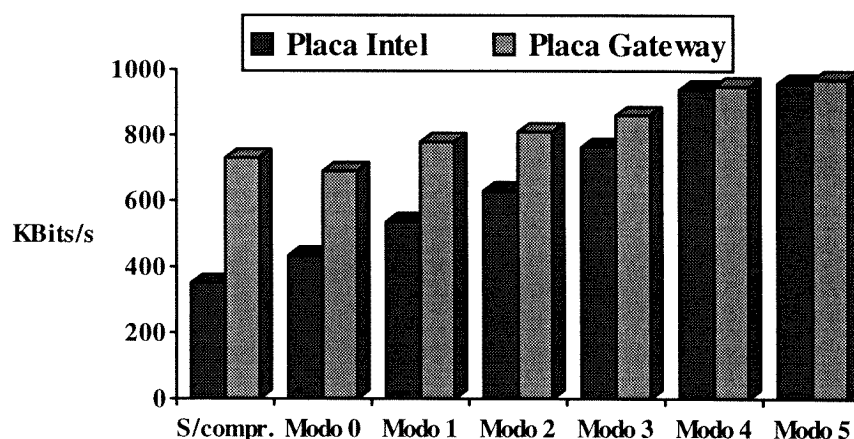
	S/compr	Modo 0	Modo 1	Modo 2	Modo 3	Modo 4	Modo 5
Tempo (s)	33.5	11.9	7	6	4.7	3.1	3

Fig. 9: Tempos de transmissão de imagens em função do Modo de compressão usado.

A rede Ethernet possui características bastante diferentes da RDIS. O facto da sua largura de banda ser substancialmente superior e a maior utilização do CPU na implementação das camadas protocolares, impede a utilização do esquema proposto de *paralelização* das tarefas de Compressão/Transmissão e Recepção/Descompressão uma vez que o CPU não tem tempo disponível entre o envio de dois pacotes. Apesar disto os resultados obtidos na utilização da compressão de

imagem durante a sua transmissão na rede Ethernet foram relativamente bons tal como mostra a figura 10.

Estes resultados foram obtidos numa rede com tráfego real, em que a taxa média de ocupação era inferior a 10%. Como se pode verificar a taxa de transmissão é dependente do tipo de placa de interface à rede usada. O uso destes algoritmos de compressão na transmissão de imagens, possui ainda a vantagem de diminuir o tráfego médio na rede, permitindo assim a obtenção de melhores performances globais da rede.



	S/compr.	Modo 0	Modo 1	Modo 2	Modo 3	Modo 4	Modo 5
Intel	350	434	536	630	763	940	960
Gateway	730	690	780	810	860	950	970

Fig. 10: Taxas de transmissão obtidas na rede ETHERNET em função da placa de interface e dos modos de compressão.

V. INTERFACE GRÁFICA

A interface gráfica desenvolvida veio dar resposta à necessidade de utilização de um sistema gráfico de alta resolução (1024*768 com 256 cores) em ambiente MS-DOS, uma vez que na altura ainda não estava disponível a API da RDIS para MS-WINDOWS. Procurou-se criar uma interface com o utilizador simples, prática e agradável, de forma a poder ser utilizada facilmente por pessoas pouco familiarizadas com os meios informáticos.

Foi desenvolvido um algoritmo que permite fazer uma gestão inteligente do espaço de visualização de imagens. Esse algoritmo distribui as imagens automaticamente pelo espaço disponível, procurando não utilizar o espaço ocupado pelas últimas imagens visualizadas. Desta forma, o utilizador fica liberto da tarefa de escolher o local de visualização das imagens. A figura 11 é um exemplo da forma como esse algoritmo distribuiu várias imagens de tamanhos diferentes. Note-se que esta distribuição é dependente da ordem pela qual se efectua a visualização das imagens.

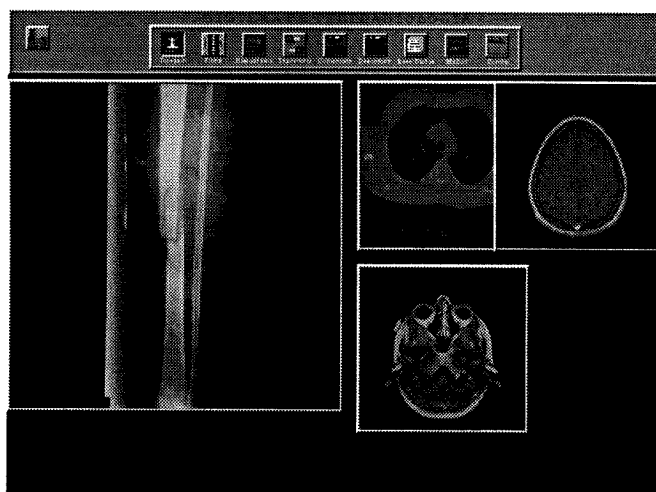


Fig. 11: Aspecto da distribuição automática de várias imagens com tamanhos diferentes.

O utilizador pode ainda escolher uma de várias paletes de cores disponíveis, abrindo assim a possibilidade de utilização de pseudo-cor como ajuda no seu diagnóstico.

VI. CONCLUSÕES

Este trabalho, tendo sido desenvolvido a pensar na utilização da RDIS como infraestrutura de comunicação, serviu para mostrar as suas potencialidades em aplicações que exijam um ambiente integrado de comunicações de voz e dados.

A possibilidade de interacção entre os sistemas internos de Imagiologia, através do sistema de Tele-imagiologia, permite que cada hospital deixe de ser uma *ilha informática*, passando a estar ligada em rede a todos os centros que possuam o mesmo sistema.

A utilização da compressão de imagem, com os diversos modos desenvolvidos permite que o utilizador possa obter diferentes taxas de compressão em função da relação sinal/ruído pretendida. As taxas e sobretudo os tempos de compressão obtidos pelos algoritmos proprietários, justificam plenamente a sua utilização, isto em detrimento de algoritmos mais comerciais.

A forma como se integraram os algoritmos de compressão com os protocolos usados na transmissão de imagens, através da *paralelização* das tarefas de compressão, transmissão, recepção e descompressão, permitiu otimizar bastante os recursos das redes usadas com especial destaque para a RDIS cujo custo de exploração é mais elevado, permitindo assim diminuir

bastante os tempos de transmissão.

A utilização destes algoritmos de compressão na transmissão de imagens através da Rede ETHERNET, cujas características são substancialmente diferentes da RDIS, mostrou que, de uma forma geral, se obtém ainda uma diminuição do tempo de transmissão.

Em resumo, pode-se afirmar que a utilização de compressão de imagem permitiu diminuir o custo de manutenção e exploração deste sistema, quer pela diminuição do espaço necessário ao armazenamento das imagens clínicas quer, sobre tudo, pela diminuição dos custos de transmissão dessas mesmas imagens.

REFERÊNCIAS

- [1] "Impact of PACS on Radiologists Daily Work in Western Countries" IEEE Journal on Selected Areas in Communications, vol 10, no 7 Setembro 1992.
- [2] "Multimedia Communications in Medical Imaging" IEEE Journal on Selected Areas in Communications, Vol 10, Nº 7, Setembro de 1992.
- [3] "Computers for Clinical Practice and Education in Radiology", INFORAD, Maio de 1993.
- [4] "JPEG-9-R6: Working Draft for Development of JPEG CD", Janeiro de 1991.
- [5] Rui Rocha Ribeiro, "Implementação de um Algoritmo de Compressão de Imagem Baseado em JPEG", relatório da disciplina de Redes de Computadores do Mestrado de Electrónica e Telecomunicações da Universidade de Aveiro, Setembro de 1992.
- [6] Rafael C. Gonzalez, "Digital Image Processing", ed. Addison-Wesley Publishing Company, Junho de 1992.
- [7] Mário Serafim Nunes, Augusto Júlio Casaca, "Redes Digitais com Integração de Serviços", Editorial Presença, 1992.

Sistema de Televisilância suportado na RDIS

José P.O. Santos, Fernando M. S. Ramos, Osvaldo A. Santos

Resumo - Este artigo apresenta uma proposta de um sistema de televisilância suportado pela RDIS.

O artigo encontra-se organizado em várias secções contemplando cada uma delas os seguintes aspectos: arquitectura do sistema e serviços RDIS, descrição funcional, análise e projecto estruturado, análise de desempenho do sistema proposto, e ainda conclusões sobre cada uma das secções anteriores.

Este sistema é composto por uma estação central (EC) de televisilância e por uma ou várias estações remotas (ERs) que comunicam através da RDIS. O sistema permite a transferência de imagens fixas ou móveis das ERs para a EC, sendo a iniciativa do estabelecimento de cada comunicação da responsabilidade do operador da EC, ou da ER; neste último caso a comunicação é estabelecida sempre que haja uma situação anómala correspondente à activação de um dos sensores da ER.

Como veremos a análise de desempenho do sistema permite concluir que o seu desempenho é adequado para a aplicação para que foi planeado e que as características da RDIS são apropriadas para o suporte deste tipo de aplicação.

Abstract - This article is a proposal of a telesurveillance system based on the ISDN - Integrated Services Digital Network.

The article is structured in several sections covering the following subjects: system architecture and ISDN services, functional description, structured analysis and project, and performance evaluation of implemented system.

The basic components of the system are a central telesurveillance station and one or several remote stations, that communicate through the ISDN. The purpose of the system is to allow the transfer of images (still or moving pictures) from the remote station whenever it is requested by the operator of the central station or when an abnormal situation occurs in a remote station; in this latter case this is related with the activation of one of the sensing devices installed in the remote station. The performance evaluation of the system leads to the conclusion that its behaviour is appropriate to its purpose, and also that ISDN provides a good support, for this type of application.

I. ARQUITECTURA DO SISTEMA E SERVIÇOS RDIS

O sistema proposto é constituído por um conjunto de estações remotas de vigilância equipadas com câmaras de vídeo, que permitem a transferência de imagens para uma

estação central a pedido do seu operador ou por iniciativa da própria estação remota no caso de ocorrer um alarme. A comunicação entre a estação central e as estações remotas é efectuada através da RDIS, utilizando para este efeito acessos básicos RDIS (2B+D), estando por isso cada estação equipada com uma placa de rede [1] PCBIT.

Na EC pretende-se basicamente visualizar imagens obtidas pelas câmaras de uma das ERs existentes.

Em vigilância de tipo convencional, os vigilantes no local podem dar o alarme ou tomar algumas acções quando uma situação anormal ocorre. Para compensar, com vantagem, a ausência de vigilantes no local este sistema deve possuir sensores e actuadores nas ERs.

Pretende-se que as ERs tenham sensores (ex. sensores de presença, fogo,...) para detectar anomalias nas instalações protegidas e poderem dessa forma tomar a iniciativa de ligar para a central a avisar do sucedido. Nessa altura a EC deve (se for conveniente) pedir imagens dessa ER.

A partir da EC o vigilante deve poder accionar automatismos (tais como: Sirenes, Trincos eléctricos, máquinas...) na instalação a proteger tal como o poderia fazer se lá estivesse presente.

TABELA 1
PARÂMETROS QUE CARACTERIZAM O SUPORTE DE COMUNICAÇÃO RDIS USADO PELO SISTEMA.

Parâmetros RDIS	serviços de comunicação RDIS, usadas pelo sistema proposto
capacidade de transferência	Voz, Informação digital não restrita.
Modo de transferência	Modo circuito
Taxa de transferência	64 Kbit/s
Configuração da estação do utilizador	Ponto a Ponto / Multiponto
Estabelecimento de ligação	pedido circuito permanente
Simetria de transmissão	Bidireccional
	Protocolos utilizados na pilha protocolar do utilizador
camada 1	Recomendações do CCITT I430/1 algoritmo de codificação de voz
camada 2	Recomendações do CCITT I440/1 (LAPD)
camada 3	não é usada

Este sistema deverá permitir vigiar instalações, dispersas geograficamente, de uma forma centralizada com redução de mão de obra e respectivos custos. Deverá também permitir um controlo de acessos a instalações com a visualização dos utilizadores.

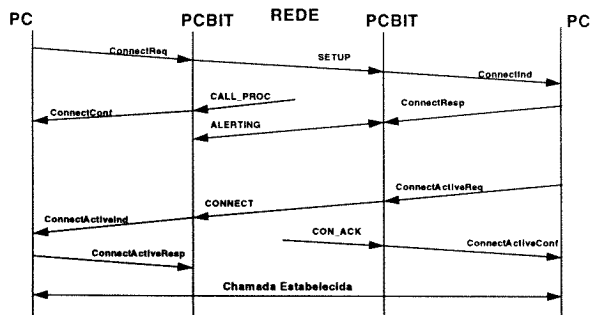


Fig. 1: Sinalização (canal D) necessária para o estabelecimento de uma ligação em comutação de circuitos no canal B.

Dado que a RDIS pode fornecer vários tipos de suportes de comunicação, nomeadamente comutação de pacotes e comutação de circuitos, com diferentes taxas de transmissão, são apresentadas na tabela seguinte os principais parâmetros suportados pela RDIS e utilizados pelo sistema.

Uma aplicação que queira aceder aos protocolos de transmissão sobre os canais B utilizando a placa de rede PCBIT, como é o caso do sistema proposto, deve começar por estabelecer a chamada RDIS como se pode ver na figura 1.

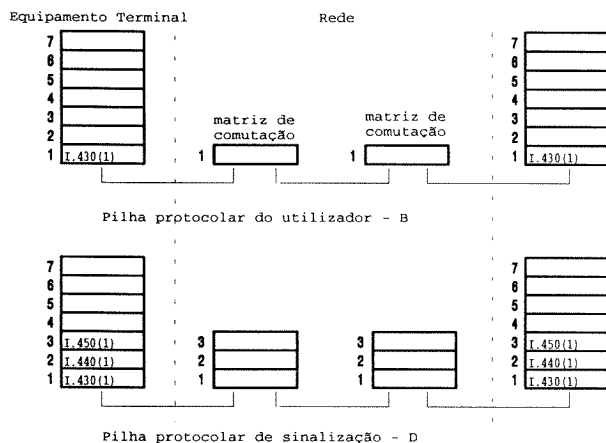


Fig. 2: Arquitectura do Protocolo para Ligação em Comutação de Circuitos

Após esta troca de mensagens de sinalização com sucesso entre o TE e rede através da pilha de protocolos de sinalização [2], a rede pode disponibilizar ao utilizador um canal de transmissão através da pilha de protocolos do utilizador ou através da própria pilha de protocolos de sinalização (fig. 2). Existem três tipos de ligações possíveis: Ligação em modo circuito nos canais de dados, ligação em modo pacote nos canais de dados e ainda

ligação em modo pacote no canal D. O sistema proposto utiliza apenas a ligação por comutação de circuitos com os protocolos indicados. Na RDIS e neste tipo de ligação a sinalização é feita fora da banda, esta situação tem a vantagem de permitir a notificação de uma nova chamada mesmo que já esteja a decorrer outra, o que é particularmente útil quando já existe uma ligação estabelecida entre a EC e uma ER, e ocorre um alarme noutra ER. Nessa altura o serviço suplementar *User to User Signalling* - UUS é usado para notificar a EC da ocorrência da anomalia.

Após o estabelecimento de ligação estar concluído, a aplicação deve proceder à selecção do protocolo (fig. 3) que quer usar na pilha de protocolos do utilizador B recorrendo à mensagem *SelectProtocolReq*, que informa a placa PCBIT local dos protocolos pretendidos. Se a placa puder aceitar essa selecção responde com a mensagem *SelectProtocolConf*; remotamente a(s) outra(s) aplicação(ões) deverão fazer o mesmo nos seus PCBIT respectivos.

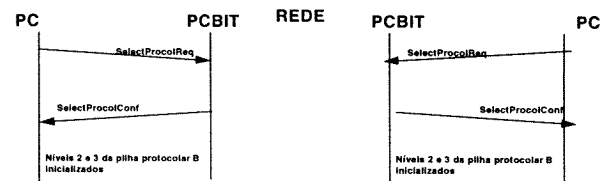


Fig. 3: Selecção do protocolo dos níveis 2 e 3 da pilha de protocolos do utilizador.

As aplicações que queiram usar transparentemente o nível 3 da pilha de protocolos do utilizador, têm à sua disposição um conjunto de mensagens de acesso directo ao nível 2 exclusivamente para o protocolo LAPD, nesse caso deverão na mensagem *SelectProtocolReq*, seleccionar o protocolo LAPD (fig. 2) para o nível 2. Depois de seleccionado este protocolo é necessário enviar para a API e desta para a RDIS pelo canal B as seguintes mensagens:

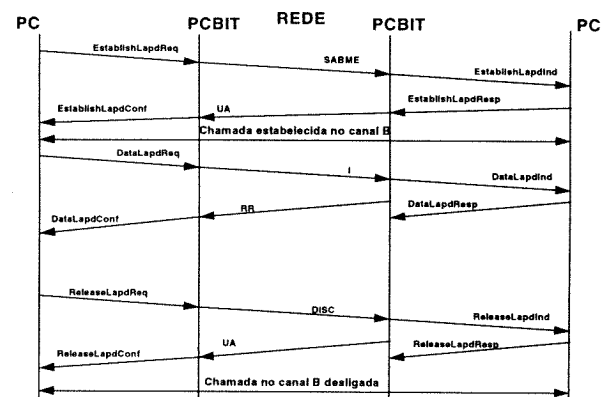


Fig. 4: Mensagens (canal B) necessárias para o estabelecimento de uma ligação LAPD no canal B.

II. DESCRIÇÃO FUNCIONAL

O sistema proposto [7], sistema de televigilância suportado pela RDIS (fig. 5), é constituído por um conjunto de ERs equipadas com câmaras vídeo que permitem a transferência de imagens para uma EC, a pedido do operador da EC ou por iniciativa das ERs no caso de ocorrer uma situação anómala.

A comunicação entre a EC e as ERs é efectuada através da RDIS, com base numa interface de acesso básico (2B+D).

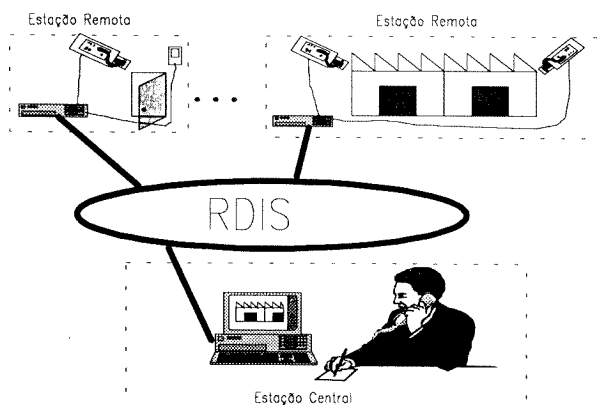


Fig. 5: Arquitectura do sistema de televigilância suportado pela RDIS

O sistema proposto é baseado em dois tipos de estações, a estação central (EC) e a estação remota (ER). A EC é baseada num computador do tipo PC compatível (de preferência com processador 486/66Mhz), equipado com:

- Carta de comunicações PCBIT, esta carta [1] dispõe de um interface com a RDIS do tipo 2B+D (acesso básico), e de um interface com a aplicação definido na API.
- Carta de processamento de imagens, esta carta é baseada no processador de sinal C30 e destina-se a suportar os algoritmos de compressão/descompressão de imagens.
- Carta gráfica SVGA (800*600 pixel) com 32000 cores
- Monitor policromático SVGA

Cada ER é igualmente baseada num computador de tipo PC compatível (de preferência com processador 486/33 Mhz ou superior), equipado com:

- Carta de comunicações PCBIT
- Carta de processamento de imagens, esta carta é idêntica à utilizada na EC, mas dispõe adicionalmente, de um interface baseado em ADCs para a aquisição e digitalização das imagens provenientes das câmaras.
- Carta de sensores e actuadores, esta carta permite a ligação de sensores e actuadores do tipo *on/off*

O sistema permite a aquisição e transferência de imagens a preto e branco com resolução espacial de 144*180 *pixel* e resolução tonal de 256 níveis de cinzento com um ritmo

de transferência de 8 imagens/segundo, e ainda a aquisição e transferência de imagens a cores com 288*360 *pixel* e 32000 cores. Em ambos os casos as imagens são previamente comprimidas na ER antes de serem enviadas para a EC; no caso das imagens a preto e branco é usado um algoritmo proprietário [3] baseado no cálculo de imagens diferença, e no caso das imagens a cores é usado o algoritmo [4] JPEG.

A carta de aquisição e processamento de imagens utilizada dispõe de seis entradas independentes de canais de vídeo, o que permite a ligação de um máximo de seis câmaras a preto e branco, ou de duas câmaras a cores RGB, ou ainda a ligação de uma câmara RGB e três a preto e branco.

Como vimos o estabelecimento de uma ligação entre a EC e as ERs pode ser da iniciativa de qualquer uma delas. No caso da iniciativa ser da EC tal resulta de uma acção nesse sentido por parte do respectivo operador; no caso da iniciativa partir da ER tal resulta de se ter verificado uma situação anómala a que corresponde a activação de um sensor. Dado que numa ER podem existir diversas câmaras e sensores, o sistema permite, para cada ER, definir uma correspondência entre câmaras e sensores, de tal forma que quando um dado sensor é activado a ER transmitirá imagens provenientes da câmara correlacionada com o sensor activado.

A EC permite a visualização simultânea de uma imagem fixa e outra móvel. As imagens a preto e branco podem ser visualizadas no formato original (144*180 *pixel*), ou expandidas para uma área quádrupla (288*360 *pixel*).

Numa ER é possível que, mesmo sem existir comunicação estabelecida com a EC, esteja a ser efectuada a aquisição de imagens de alta resolução, as quais são armazenadas num *buffer* circular com capacidade para armazenar 4 imagens. Esta facilidade permite que, em caso de activação de um sensor, sejam guardadas na ER as duas imagens adquiridas imediatamente antes e as duas imagens adquiridas imediatamente após o instante do alarme. Esta sequência de quatro imagens, designada por pré-imagens, é guardada na ER podendo por iniciativa do operador da EC, virem a ser transferidas para visualização e armazenamento na EC.

O sistema permite ainda, que por iniciativa do operador da EC e na situação de estar estabelecida uma comunicação com transferência de imagem móvel, suspender temporariamente este modo de operação para ser obtida uma imagem de alta resolução (288*360 *pixel*) que poderá ser a cores ou a preto e branco consoante o tipo da câmara seleccionada. Esta facilidade pode ser usada sempre que a resolução limitada da imagem móvel não seja suficiente para caracterizar um dado pormenor.

A utilização da RDIS como rede de suporte neste sistema resulta desta rede ter algumas características particulares, algumas das quais não disponíveis noutras redes, nomeadamente:

- Canais de transferência de informação em modo circuito a 64 Kbits/s

- Disponibilidade do serviço UUS (User to User Signaling) usado para a transferência da ER para a EC das informações sobre os sensores.
- Tempo de estabelecimento de ligação em modo circuito da ordem dos 4 segundos
- Possibilidade de ligar até oito ERs no mesmo acesso básico.
- Tendência para a disponibilidade universal de acessos.

A. Facilidades da estação central

A EC permite ao operador o controlo da configuração e operação do sistema, quer no que respeita à EC quer no que respeita às ERs.

Estas facilidades estão disponíveis no écran principal da EC, e estão agrupadas nos seguintes menus:

- Configuração
- Configuração da EC

- Configuração da ER
- Operação
- Alteração
- Visualização

Nos parágrafos seguintes são detalhadas as facilidades disponíveis em cada um destes menus.

A.1. Configuração da estação central:

Esta opção (fig. 6) permite ao operador definir os atributos da EC: nome, endereço e sub endereço RDIS e chave de acesso. A chave de acesso é fundamental para que alterações futuras da própria configuração da EC sejam aceites.

Estes atributos (ex. endereço da EC) são comunicados à ER quando da sua configuração, para que esta possa contactar a EC no caso de ocorrer um alarme.

Fig. 6: Configuração da estação central.

A.2. Configuração da estação remota

Esta opção (fig. 7) permite ao operador gerir a configuração das ERs instaladas. Para este efeito o operador pode *Inserir*, *Alterar* ou *Remover* ERs.

Cada ER é caracterizada por um conjunto de atributos: nome, endereço RDIS, sub endereço RDIS, chave de acesso, câmaras existentes e sensores. Estes atributos, definidos durante a operação de inserção ou alteração de uma ER, são comunicados pela EC à ER, só sendo

considerados atribuídos depois de aceites pela ER. Os atributos: sensores e câmaras podem ser correlacionados por forma que a ocorrência de um alarme e o consequente accionamento de um sensor (1 a 8) determine que sejam vistas imagens da câmara indicada (F1 a F8) na EC.

Quando uma ER é inserida a EC fica com uma cópia desses atributos. Quando a ER é configurada recebe não só os seus atributos, mas também o endereço e sub endereço da EC. Depois de uma ER ter sido inserida só é possível alterar os seus atributos ou removê-la conhecendo

a sua chave de acesso. A remoção de uma ER tem duas consequências: a eliminação dos atributos da EC existente na ER e a eliminação dos atributos dessa ER da lista das ERs, existente na EC. Por isso, após a remoção de uma ER

esta fica impossibilitada de estabelecer ligação para a EC mesmo quando um dos seus sensores é accionado, e a EC deixa de a reconhecer não sendo permitido ao operador seleccioná-la.

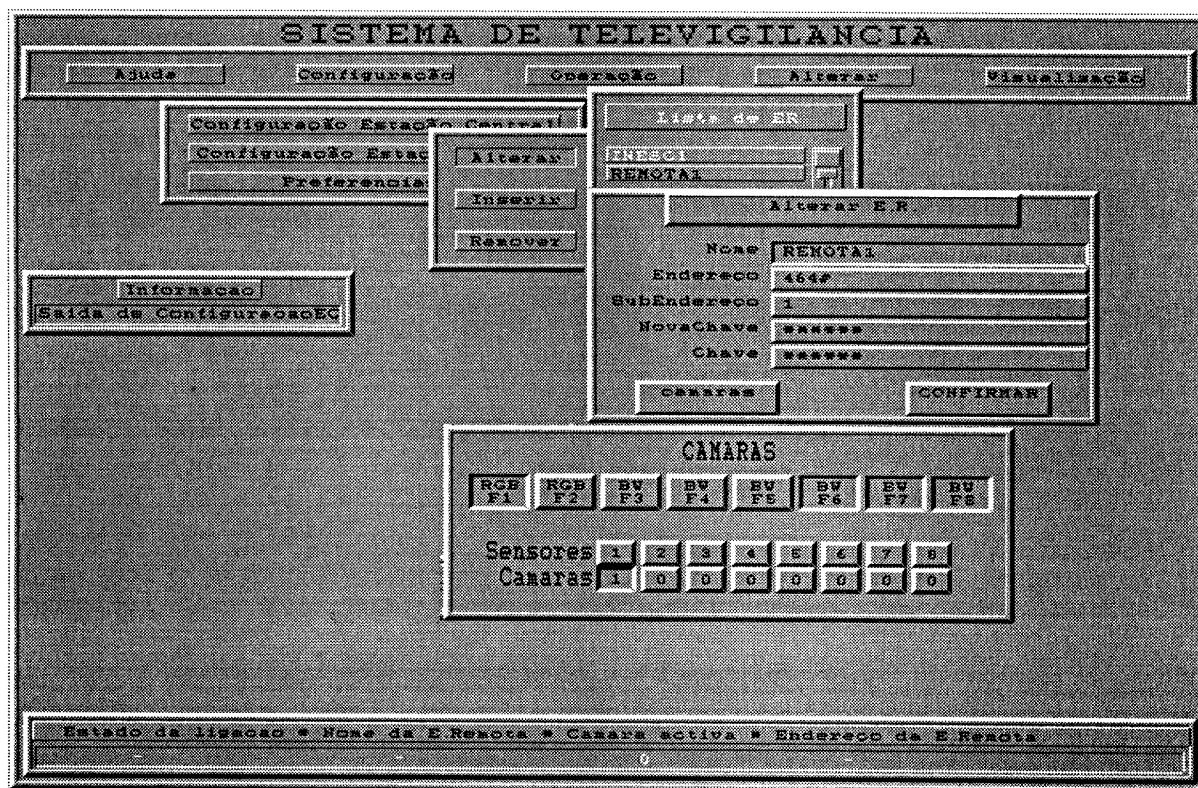


Fig. 7: Configuração da estação remota

A.3. Operação

Esta opção (fig. 8) permite ao operador da EC activar ou terminar a recepção de imagens provenientes uma câmara remota.

Quando o operador selecciona esta opção é-lhe apresentada uma lista das ERs configuradas (fig. 8). Depois de seleccionada uma das ERs são-lhe apresentadas as câmaras disponíveis nessa ER. Logo que o operador escolhe a câmara de onde pretende obter imagens é iniciado o processo de estabelecimento da ligação de rede com a ER seleccionada. Quando a ligação se encontrar estabelecida é transmitida para a ER a ordem para começar a adquirir e enviar imagens da câmara escolhida para a EC.

Ao escolher a câmara o operador selecciona implicitamente o tipo de imagem que pretende obter, imagem fixa (a cores ou a preto e branco) ou imagem móvel a preto e branco. No caso do operador seleccionar uma imagem fixa pode definir qual a qualidade de imagem visual que pretende alterando as tabelas de quantificação: *Muito Boa*, *Boa*, ou *Razoável*; convém notar que uma

qualidade elevada implica maior demora na compressão e transmissão da imagem fixa.

Por *Terminar* entende-se o fim da ligação com a ER e consequentemente o fim da recepção de imagens.

A.4. Alteração

Esta opção (fig. 9) permite ao operador, seleccionar outra *Câmara*, *Parar*, *Continuar* ou *Gravar*.

Por seleccionar outra *câmara* entende-se o envio de uma mensagem para a ER com a indicação da nova câmara de onde se pretende obter imagens.

Por *Parar* entende-se a suspensão do envio de imagens móveis continuando a ligação com a ER activa.

Por *Continuar* entende-se a continuação da recepção de imagens móveis vindas da ER.

Por *Gravar* entende-se a gravação em disco na EC da imagem a cores presente no écran. A imagem fixa a cores será gravada com a indicação da câmara, do dia e da hora em que foi obtida, num ficheiro com um nome escolhido pelo operador.

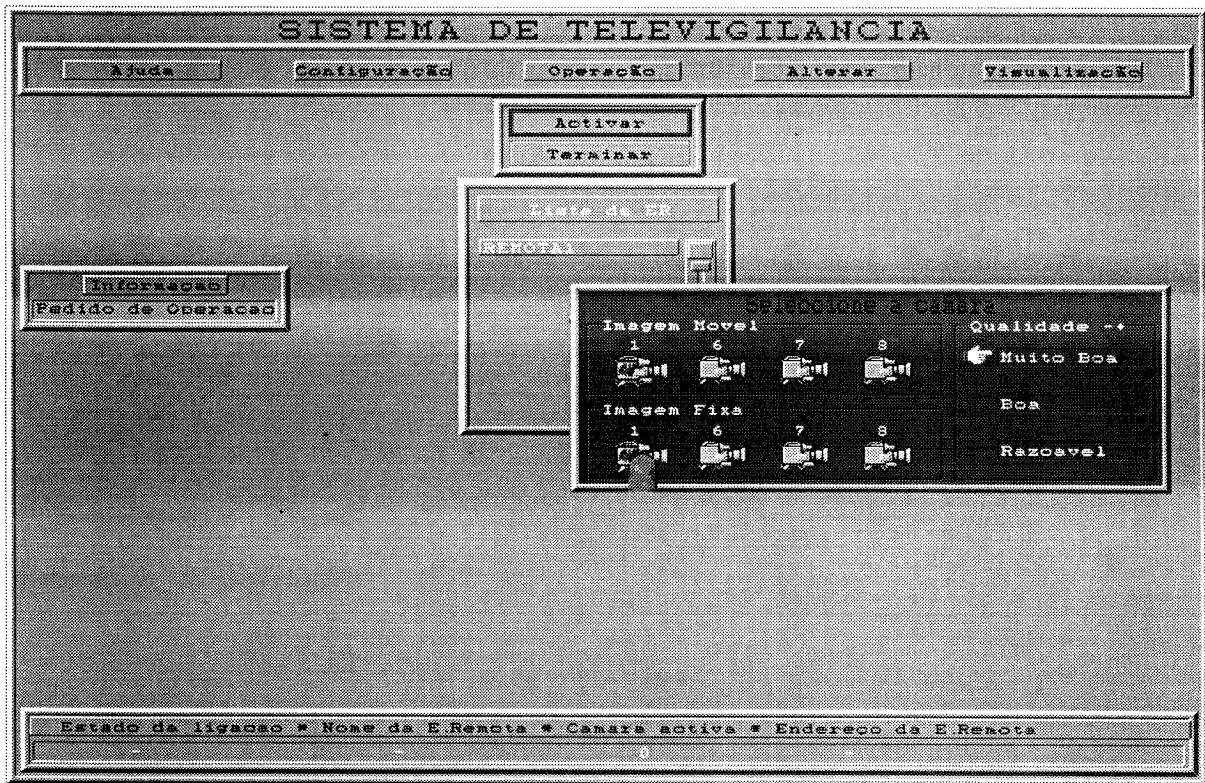


Fig. 8: Activar a recepção de imagens - Selecção da ER



Fig. 9: Alterar a câmara de onde se obtêm imagens.

III. ANÁLISE E PROJECTO ESTRUTURADO

Para o desenvolvimento deste sistema, e antes do mesmo começar a ser implementado, foram usadas técnicas de análise e projecto estruturado.

Para a análise estruturada foi seguida a metodologia proposta por Yourdon [5]. A figura 10 apresenta o modelo Ambiente do sistema em análise.

Na análise estruturada, utilizando diversas ferramentas como os diagramas de fluxos de dados - DFD, dicionário de dados - DD e especificação de processos - EP, foram descritos os fluxos de dados, os armazéns, e ainda as diversas operações que o sistema a desenvolver deverá, respectivamente, tratar, possuir e efectuar para que os requisitos inicialmente especificados sejam cumpridos [7].

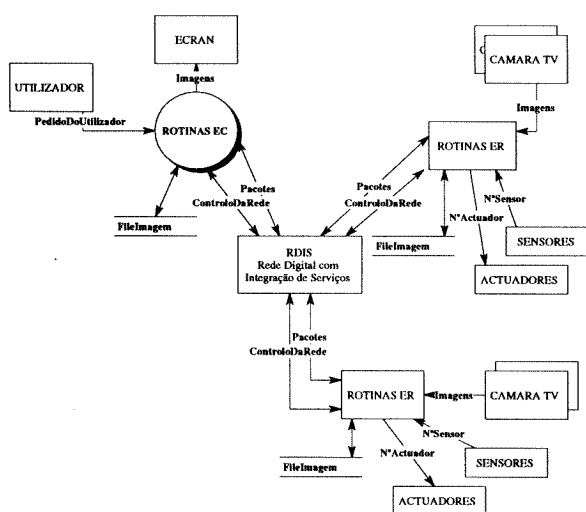


Fig. 10 : Modelo ambiente do sistema de televigilância

A figura 11 apresenta o diagrama de fluxo de dados do modelo de comportamento do sistema em análise.

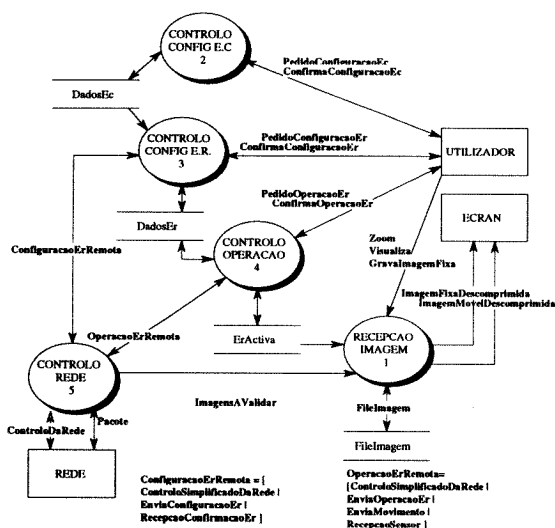


Fig. 11 - DFD 0: Rotinas EC

Usando as especificações criadas pela análise estruturada poderíamos começar logo a implementar o sistema, no entanto antes de o fazermos é conveniente considerar um conjunto de aspectos relacionados com o ambiente e com o tipo de tecnologia sobre a qual se irá implementar o sistema - projecto estruturado.

Com este objectivo Yourdon [5 - CAP22] propõe a criação de um diagrama de estrutura inicial (fig. 12), através da aplicação de técnicas de análise transformacional ou de análise transaccional [5-Cap22] [6] ao modelo de comportamento do sistema.

Neste caso como existem dois centros de transacção de fluxos, dado que os fluxos provenientes do *Utilizador* e os fluxos provenientes da *Rede* consoante o seu tipo são encaminhados para processos diferentes permitindo completar o seu processamento [6], optou-se pela análise transaccional, aliás Yourdon e Constantine referem que este tipo de análise se costuma adaptar particularmente bem a sistemas de tempo real com os processos de controlo e aquisição de dados.

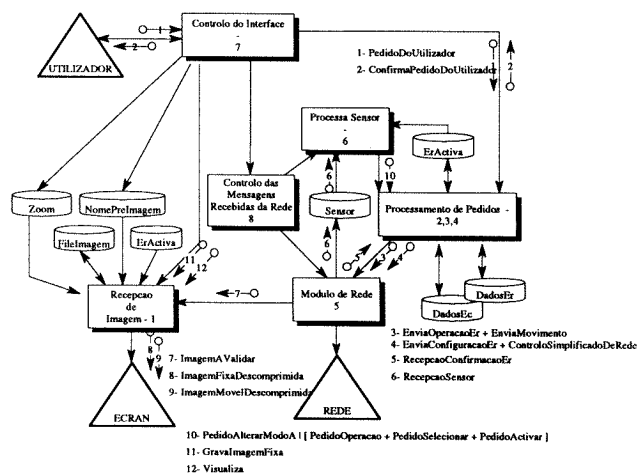


Fig. 12: Diagrama de estrutura inicial

Com base no diagrama de estrutura desenvolvido no Projecto estruturado, mais exactamente, com base nos módulos especificado (correspondendo cada módulo a uma ou várias rotinas) foi então implementadas o sistema.

IV. ANÁLISE DE DESEMPENHO

Esta secção tem como objectivo determinar o desempenho do sistema proposto [7], sob duas perspectivas: tempo de resposta do sistema a pedidos provenientes do operador ou da ER quando esta detectar situações anómalas.

Para o desenvolvimento do *software* da EC e da ER e para as medições efectuadas nesta secção foram usados: computadores baseados no processador Intel486 com o sistema operativo DOS e a ferramenta de *software* BorlandC, tendo o *software* sido desenvolvido em linguagem C++.

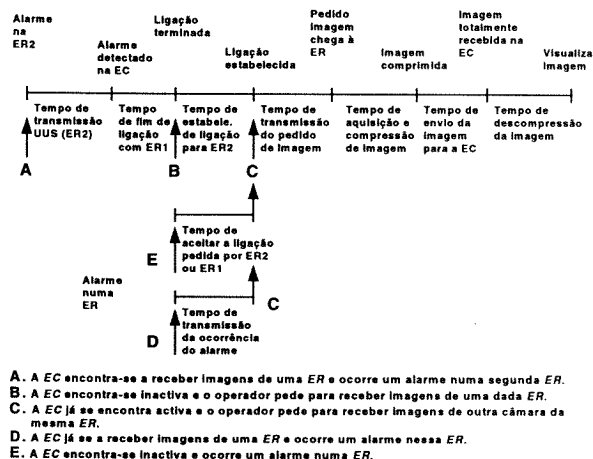
A. Identificação dos parâmetros com influência no desempenho

Para identificar os parâmetros com influência no desempenho deste sistema devemos analisá-lo numa perspectiva extremo a extremo, ou seja interessa considerar o tempo despendido desde a ocorrência de um pedido até à conclusão do seu processamento. O tempo despendido entre a ocorrência de um alarme ou a ocorrência de um pedido do utilizador e a visualização da primeira imagem relativa à câmara correspondente, é um dos parâmetros mais importantes. Na figura seguinte são apresentadas cinco situações possíveis (A-E), dependendo dos pedidos que o sistema recebe e do estado em que a EC se encontra.

TABELA 2
IDENTIFICAÇÃO DAS DIFERENTES SITUAÇÕES EM QUE O DESEMPENHO TEMPORAL DO SISTEMA É CRÍTICO, EM FUNÇÃO DO ESTADO DA EC E DO TIPO DE OCORRÊNCIA.

Estado da EC	Tipo de ocorrência			
	Alarme na ER1	Alarme na ER2	Activar imagem	Alterar a câmara
ligação terminada	E	E	B	-
ligação estab. c/ ER1	D	A	-	C

Na figura 13 (desde o ponto A) podemos observar os procedimentos necessários para visualizar na EC imagens obtidas numa ER relativas a uma situação anómala, quando a EC se encontrar a receber imagens vindas de outra ER (ER2).



nota: neste figura apenas se pretende mostrar a sequência temporal de processamento de ocorrências; o espaçamento gráfico entre processamentos não tem significado temporal.

Fig. 13: Procedimentos necessários à visualização de imagens na EC.

Esta situação foi escolhida dado ser a situação mais complexa e que envolve o maior número de procedimentos, permitindo assim determinar o tempo no maior número de procedimentos possível, dado que esses procedimentos ou parte deles são comuns ao tratamento de

outras situações por parte do sistema; aliás na mesma figura podemos também observar os procedimentos necessários a outras situações, como por exemplo:

- Quando o operador (observar a fig. 13 desde o ponto B até *Visualiza Imagem*) toma iniciativa de pedir para receber imagens de uma ER, estando a EC inactiva.
- Quando o operador (observar a fig. 13 desde o ponto C até *Visualiza Imagem*) toma iniciativa de pedir para receber imagens de outra câmara, da ER que já se encontra activa.

Para que seja possível detectar e visualizar na EC com rapidez situações anómalas que ocorram na ER pode-se concluir que é necessário que o sub sistema de aquisição/compressão de imagens, o sub sistema de comunicações e o sub sistema de descompressão tenham um elevado desempenho. É de salientar ainda, a importância dos sub sistemas de aquisição/compressão e de descompressão para que a relação "taxa de compressão - qualidade de imagem" seja elevada, dado que quanto menos *bytes* ocuparem as imagens comprimidas maior a sua taxa de transferência; por outro lado, quanto maior a taxa de compressão mais tempo demora a compressão a ser efectuada ou menor é a qualidade da imagem comprimida.

B. Quantificação dos parâmetros com influência no desempenho

Existem basicamente dois tipos de parâmetros a quantificar, tempos de transmissão e tempos de processamento em cada estação. Este último é o mais fácil de quantificar, graças à leitura de uma variável da RAM BIOS incrementada 18.2 vezes pelo relógio interno do computador pessoal permitindo, por isso, uma resolução máxima de 55 ms. Os tempos de transmissão são mais difíceis de quantificar mas podem ser determinados se os relógios da EC e da ER forem previamente sincronizados (*reset* das variáveis da RAM BIOS no mesmo instante) e se nos pacotes transmitidos constar o instante de envio (*T* envio). Neste cenário o tempo de transmissão pode ser obtido através da diferença entre o instante de recepção (*T* recepção) e o instante de envio (*T* envio):

$$T_{\text{transmissão}} = T_{\text{recepção}} - T_{\text{envio}}$$

A sincronização dos relógios da EC e da ER foi obtida através da inicialização simultânea das respectivas variáveis da RAM BIOS. Esta inicialização foi efectuada em cada máquina através de um programa que monitoriza o estado lógico do pino 13 da interface paralela (printer OnLine), sendo forçada a ocorrência simultânea do estado "0" através de uma ligação à massa. A simultaneidade é garantida através do *shunt* de ambos os pinos das duas máquinas.

Os valores das medições efectuadas foram registados no ficheiro de LOG do módulo de *Rede* e no ficheiro de LOG do módulo de *Recepção Imagem*.

B.1. Sub sistema de comunicações

Quando uma anomalia é detectada na ER o tempo de estabelecimento da ligação é crítico, dado que se pretende visualizar na EC em tempo real e tão depressa quanto possível o sucedido na ER.

Depois da ligação com uma ER estar concluída, seja por iniciativa da ER ou do operador da EC, o tempo de transmissão das imagens, ou o tempo de transmissão de uma anomalia que ocorra nessa ou noutra ER, é importante para um bom desempenho do sistema.

O tempo de fim de ligação pode também ser crítico, sobretudo quando a EC se encontra a monitorar uma dada ER e é notificada da ocorrência de uma anomalia noutra ER: neste caso é necessário terminar a ligação existente e estabelecer uma nova.

Em resumo, para analisar o desempenho do sub sistema de comunicações devemos ter em conta quatro parâmetros: tempo de estabelecimento de ligação, tempo sinalização da ocorrência de uma situação anómala na ER, tempo de transferência de dados, e ainda o tempo necessário para terminar uma ligação.

O tempo de estabelecimento da ligação corresponde ao tempo de execução da função *UserConnectReq* dado que o *overhead* introduzido por esta função é negligenciável, limitando-se a evocar um conjunto de funções da API, de uma forma ordenada. Este tempo bem como os restantes tempos medidos neste capítulo, são medidos através da leitura de uma variável da *RAM BIOS*. A tabela 3 apresenta os valores característicos do tempo de estabelecimento de uma ligação, pedido pela ER quando detecta uma anomalia, e pela EC.

Quando a ligação de rede entre a ER e a EC não se encontra estabelecida e ocorre um alarme numa ER, esta tenta estabelecer ligação com a EC para lhe comunicar o sucedido. Para isso envia-lhe a mensagem *REDIS - SETUP* que contém entre outras informações o número do sensor que detectou a anomalia e o nome da ER que a envia, ambos definidos no elemento de informação *User to User Signaling -UUS*.

TABELA 3
TEMPO DE ESTABELECIMENTO DE LIGAÇÃO

Iniciativa	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
ER -> EC	4.4	4.6	4.8
EC -> ER	4	4.3	4.5

Mesmo que a EC não queira atender imediatamente esse pedido de ligação, toma de imediato conhecimento da ocorrência na EC, podendo mais tarde activá-la. Nesta situação o tempo de transmissão da mensagem de *SETUP* corresponde ao tempo que a EC demora a detectar a ocorrência de um alarme. A tabela 4 apresenta os valores

característicos da transmissão da mensagem de *SETUP* desde o instante em que o alarme é detectado até ao instante em que a EC toma conhecimento.

TABELA 4
TEMPO DE TRANSMISSÃO UUS.

Iniciativa	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
ER -> EC	2.3	2.4	3.2

Quando a ligação de rede entre a EC e a ER já se encontra estabelecida esta última pode enviar para a EC três tipos de dados: imagem móvel, imagem fixa e informação sobre a ocorrência de uma anomalia. Por seu lado a EC pode enviar para a ER pedidos de configuração ou operação. Para analisar o desempenho do sistema é necessário conhecer os tempos de transmissão (tabela 5) de cada um destes dados, desde o instante que a função *UserDataReq* é evocada para enviar dados pela rede até ao instante em que esses dados são totalmente recebidos no destino.

TABELA 5
TEMPO DE TRANSMISSÃO DOS DADOS.

ER -> EC	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
Imagem móvel	0.16	0.22	0.27
Imagem Fixa	0.8		1.9
Sensor	0.22	0.27	0.33

Pedidos feitos pela EC à ER	< 0.055	< 0.055	< 0.055
-----------------------------	---------	---------	---------

O tempo de fim de ligação (tabela 6) corresponde ao tempo de execução da função *UserDisconnectReq* dado que esta função se limita a evocar um conjunto de funções da API de uma forma ordenada. Os valores apresentados referem-se apenas a pedidos da EC, dado que em nenhum caso a ER pode solicitar esta acção.

TABELA 6
TEMPO DE FIM DE LIGAÇÃO

Iniciativa	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
ER -> EC	-	-	-
EC -> ER	1.8	2	2.4

B.2. Sub sistema de processamento de imagem

Como já vimos é de esperar que o atraso na obtenção de imagens e a sua qualidade seja em grande medida condicionada pelos sub sistemas de processamento de imagem. Para que seja possível determinar objectivamente

o desempenho destes sub sistemas é necessário conhecer os tempos de aquisição e de compressão e descompressão de imagens móveis e imagens fixas.

A tabela 7 apresenta os tempos de aquisição e compressão de uma imagem, móvel ou fixa. Estes tempos foram medidos como um todo, dado que ao sistema apenas interessa saber o tempo que decorrido entre o momento em que a ordem de aquisição foi dada à placa de imagem e o momento em que esta possui um *array* com a imagem já comprimida.

TABELA 7
VARIAÇÃO DO TEMPO DE AQUISIÇÃO/COMPRESSÃO

Aqui/Compressão	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
Imagem Móvel	0.1	0.12	0.16
Imagem Fixa	0.9	1	1.2

Quando a EC recebe da rede uma imagem comprimida vários procedimentos são realizados: primeiro a imagem é armazenada na memória da placa de imagem, em seguida é dada ordem à placa para a descomprimir, depois a imagem descomprimida é transferida da memória da placa para a memória da EC e finalmente é visualizada. Destes procedimentos interessa sobretudo determinar o tempo de descompressão dado que este é seguramente o mais significativo.

A tabela 8 apresenta o tempo de descompressão de uma imagem móvel ou fixa, desde o momento em que é dada a ordem à placa de imagem armazenada na sua memória até ao momento em que esta possui um *array* com a imagem descomprimida.

TABELA 8
VARIAÇÃO DO TEMPO DE DESCOMPRESSÃO DE UMA IMAGEM

Descompressão	Valor mínimo (s)	Valor médio (s)	Valor máximo (s)
Imagem Móvel	< 55 ms	< 55 ms	< 55 ms
Imagem Fixa	0.88	0.93	0.99

V. CONCLUSÕES

A. Implementação e Análise de Desempenho

A.1. Organização do software

Tendo por base o objectivo de escrever código de uma forma estruturada por forma a facilitar a sua reutilização, manutenção e compreensão, optou-se por organizar as diferentes funções em três tipos principais: rotinas de *Interface*, rotinas de *Processamento* e rotinas de *Comunicação*. Para melhor se compreender as vantagens desta organização convém analisar o funcionamento de uma aplicação com comunicações e o papel de cada um destes tipos de rotinas.

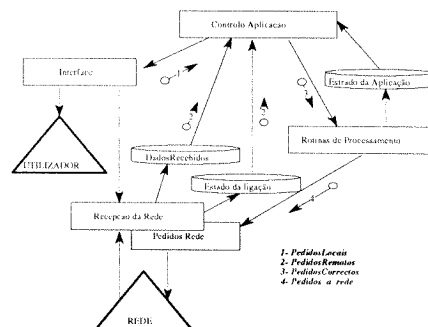


Fig. 14: Organização do software.

Na fig. 14 podemos observar que a rotina de *Controlo da Aplicação* chama as rotinas de *Interface* com o *Utilizador* para lhe permitir seleccionar opções (*PedidosLocais*). Logo devem existir tantas rotinas de *Interface* como opções, janelas e menus disponíveis para o *Utilizador*. Por exemplo, a rotina *IntroduzChaveAcesso(char * Chave)* deve, no caso de um interface gráfico, criar uma janela que permita ao utilizador escrever a chave de acesso. Como as rotinas de *Interface* podem manter a execução da execução do programa por muito tempo (ex. introdução de uma chave de acesso, feita pelo utilizador), para que a rede possa continuar a ser monitorada as rotinas de *Recepção da Rede* (que fazem parte das rotinas de *Comunicação*) deverão continuar a ser chamadas. As rotinas de *Recepção da Rede* podem também ser chamadas pelas rotinas de *Processamento* se estas mantiverem por muito tempo o controlo da execução do programa ou se necessitarem de monitorar a rede. Em qualquer dos casos os dados recebidos da rede são armazenados num *buffer* de recepção. O *Controlo da Aplicação* lê o *buffer* de recepção (*PedidosRemotos*) e consoante o tipo de dado recebido chama a rotina de processamento adequada.

Para se obter uma melhor organização do software cada tipo de rotinas deve ser implementada no seu directório respectivo.

Além disso, e apesar de em cada directório só existirem rotinas de um tipo, devem mesmo assim estar organizadas em vários ficheiros possuindo um deles todas as rotinas de mais alto nível usadas pelos outros módulos, sendo as restantes rotinas (auxiliares, de processamento intermédio, etc.) desse tipo agrupadas noutros ficheiros nesse mesmo directório. Por exemplo, no caso do módulo de processamento, devem existir tantas rotinas de processamento de alto nível nesse ficheiro como pedidos feitos a este módulo e o nome de cada rotina deve ser igual ao pedido correspondente, tendo essa rotina tantos parâmetros como os campos do pedido correspondente.

As rotinas de processamento podem enviar para a rede dados (*Pedidos_a_Rede*). Se os pedidos feitos à estação remota forem bem sucedidos ou se os pedidos recebidos poderem ser satisfeitos o estado da aplicação é alterado. O estado da aplicação é importante porque permite ao

controlo de aplicação e às rotinas de processamento, validar (*PedidosCorrectos*) ou tratar de forma diferente os pedidos recebidos.

A separação entre os três tipos base de rotinas, rotinas de *Interface*, rotinas de *Comunicação* e rotinas de *Processamento*, permitem também um desenvolvimento em paralelo da aplicação por dois ou três programadores, facilitando a integração e os testes finais uma vez que é possível realizar testes individuais. A própria integração é gradual uma vez que cada módulo quando atinge alguma maturidade é passado aos outros programadores.

Em conclusão e pelas razões acima expostas esta separação entre rotinas tem as seguintes vantagens:

- Maior rapidez de desenvolvimento de aplicações.
- Melhor organização do software.
- Permite uma identificação e uma correcção mais fácil dos erros existentes
- A manutenção do software torna-se mais fácil.
- Alterações do interface, das rotinas de rede ou do processamento de pedidos são independentes do resto do programa. Apenas o módulo alterado necessita de ser recompilado sendo depois *linkado* com os restantes.

A.2. Desempenho do Sistema

Para ser possível determinar se os parâmetros identificados são realmente significativos, isto é, se são na realidade os principais responsáveis pelo tempo de resposta do sistema, é necessário comparar o tempo total gasto por eles, para cada uma das cinco situações consideradas, com o tempo de resposta total do sistema.

Na Tabela 9 são apresentados no seu conjunto e para cada uma das cinco situações consideradas os tempos médios de processamento medidos e o tempo total de cada uma delas.

Convém ter em conta que os valores medidos tem uma resolução de 0.055 segundos e podem, no caso das imagens fixas, ser significativamente influenciados pelo tipo de imagem adquirida, na medida em que o tempo de processamento e transmissão pode variar no seu conjunto de mais ou menos 1 a 2 segundos; no entanto, e apesar disso, podemos a partir destes valores concluir que os parâmetros identificados são os que contribuem de forma significativa para o desempenho do sistema.

Existe no entanto, na situação C, uma diferença significativa entre o tempo esperado e o tempo real de visualização da primeira imagem móvel, devido ao tempo de reinicialização da placa de processamento de imagem da ER sempre que se altera o seu modo de funcionamento (pré-imagens, imagem fixa, ou imagem móvel) que não foi considerado.

Com base nestes valores podemos concluir que só é possível melhorar o desempenho do sistema se for possível diminuir os tempos de processamento, de estabelecimento de ligação ou de transmissão, ou ainda se for possível paralelizar estas duas actividades. Esta

solução já foi implementada para a imagem móvel, e só por isso é possível receber oito imagens por segundo ao contrário das 4 imagens por segundo que os tempos medidos para a aquisição/compressão, transmissão e descompressão da primeira imagem móvel visualizada na EC permitiriam concluir. A mesma solução pode e deve ser implementada para a imagem fixa, devendo por isso, surgir um novo modo de funcionamento (refrescamento periódico de imagem fixa) que permitiria receber uma imagem de qualidade cada 2 ou 3 segundos.

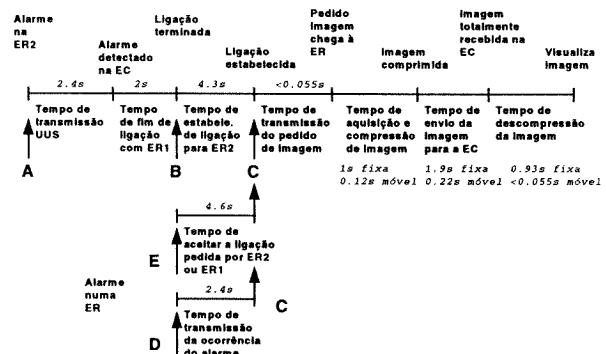


Fig. 15: Tempos considerados significativos para o tempo de resposta total do sistema.

Tabela 9

Comparação entre os tempos de resposta totais e o somatório dos tempos relativos aos parâmetros considerados significativos.

	A(s)	B(s)	C(s)	D(s)	E(s)
tempo considerados	9	4.64 (imag. móvel) 8.17 (imag. fixa)	0.34 (imag. móvel) 3.87 (imag. fixa)	2.74	4.94
tempo total	10	5 (imag. móvel) 8 (imag. fixa)	0.8 (imag. móvel) 4 (imag. fixa)	3	5.5

Existe ainda outra solução, à semelhança do que é feito para a imagem móvel, que permitiria obter um maior número de imagens por segundo com uma qualidade intermédia; esta solução consiste em aplicar o algoritmo das diferenças entre imagens consecutivas às imagens fixas, e posteriormente aplicar o algoritmo JPEG às diferenças, passando o sistema a disponibilizar imagens a cores com uma cadência inferior às imagens móveis mas com uma qualidade superior a estas.

REFERÊNCIAS

- [1] Augusto Casaca, João Lourenço, M. Serafim Nunes: "Personal Computer Based ISDN Terminal - PCBIT", *Telemática 90*, IFIP TC6, Portalegre, Brasil, Set. 1990.

- [2] Peter Bocher , "The Integrated Service Digital Network", ed. Springer Verlag, 1988.
- [3] L. Corte-Real, A. Pimenta Alves: "Vector Quantization of Images Sequences Using Variable Size And Variable Shape Blocks", *Electronics Letters* 30th August 1990, vol. 26 Nº 18 pp. 1483-1484.
- [4] ISO/IEC JTC1/SC2/WG10/CCITT SGV VIII, "Working Draft for Development of JPEG CD", *JPEG - 9 - R6*, January 1991.
- [5] E. Yourdon, "Modern Structured Analysis", ed. Prentice-Hall, Inc, Englewood Cliffs, N.J.,U.S.A.,1989.
- [6] Edward Yourdon and Larry Constantine, "Structured Design", Englewood Cliffs, NJ: Yourdon Press/Prentice-Hall, 1979.
- [7] J. Santos, "Especificação, Análise, Projecto e Implementação de um Sistema de Televisão suportado pela RDIS", Tese de Mestrado, Universidade de Coimbra, 1994.

Voice Integration in a TCP/IP/Ethernet based LAN

Joaquim Azevedo, Joaquim Arnaldo Martins, Pedro Guedes de Oliveira

Resumo- Neste trabalho é feito um estudo experimental de um sistema de comunicação de voz a 64 kbit/s, com base em computadores pessoais, sobre a rede local Ethernet. A integração de voz e dados é alcançada sem necessidade de recorreremos a qualquer novo protocolo para comutação de pacotes e sem quaisquer alterações às especificações do protocolo CSMA/CD. O desempenho do sistema é avaliado segundo o atraso médio de transmissão e percentagem de pacotes perdidos em função da carga na rede, do tamanho dos pacotes, do tamanho dos *buffers* e das características do tráfego. Apresentamos, ainda, uma estimativa do número máximo de canais de voz, sem degradação significativa da inteligibilidade do sinal reconstruído. É feita uma referência ao problema da sincronização em sistemas de voz com comutação de pacotes sendo sugerido um método para eliminar o *clock drift*. É proposto um algoritmo de reconstrução sem perdas para um sistema sem detecção de silêncio. Finalmente, é apresentada uma implementação de um canal de voz sobre o protocolo TCP/IP.

Abstract- This work presents an experimental study on a personal computer based implementation of a 64 kbit/s voice communication system over the Ethernet local network. Voice/data integration is achieved without resorting to any new packet switching protocol and without any changes to the CSMA/CD protocol specifications. System performance is evaluated according to its average transmission delay and percentage of packet loss as a function of network load, packet size, buffers size, and traffic characteristics. An estimate of the maximum number of voice channels that can be supported without significant degradation of the reconstructed signal is also presented. A reference is made to synchronisation problems in packet voice systems and a method to eliminate clock drift is suggested. A non packet discarding reconstruction algorithm for a system with no silence detection is proposed. Finally, a voice channel implementation over TCP/IP protocols is presented.

I. INTRODUCTION

Lately, great efforts have been made in order to integrate voice and video into packet switching networks. Many of these are intended to integrate real-time voice and data in IEEE 802.3-5 local computer networks, some of which involve simulations [1-6] and others experimental implementations [7-9].

Future trend is to integrate voice, data, and video into a single workstation. Broadband networks (ex: FDDI,

B-ISDN) are expected to support these multimedia applications, however, the idea of using Ethernet networks as transmission medium should not be put aside. They represent a low error with reasonable bandwidth, cheap and widely spread transmission medium for communications within buildings and other institutions (ex: universities, research centres, industrial manufacturing sites, etc). On the other hand, some of these networks, which today handle only data, have a low occupation rate. Finally, with Ethernet's migration from 10 to 100 Mbit/s [10], it is expected that a greater number and more bandwidth demanding applications can be supported.

In the sections which follow, we review voice traffic characteristics and the Ethernet local network, describe an experimental prototype design and implementation, measure system performance when only voice traffic is present, and then measure its behaviour in an integrated voice/data environment.

II. TRANSMITTING REAL TIME VOICE ON ETHERNET

A. Voice traffic characteristics

Digital speech, with telephone quality at 64 kbit/s, is achieved by sampling voice signal at 8 kHz, 8 bits per sample. Speech consists of alternating segments of speech followed by silence, with each speaker, in a typical conversation, talking only about 40% of the time and being silent for the rest [1]. This characteristic can be exploited in order to achieve greater system performance by silence suppression. The primary requirement for voice traffic is its real-time treatment. Voice samples which are not delivered within some previously well established period are rejected, resulting in speech loss. Owing to speech redundancy and attending to human audition characteristics, losses of 1 or 2% are acceptable without incurring significant quality degradation [2, 11].

Transmission/reception of digital voice through a packet switched network is simple, but leads to inevitable delays. This delay has a fixed and a variable component. Packet generation time, time spent in reception queues, and other minor fixed delays (ex: transmission time) contribute for the fixed component. The variable component is essentially due to the delay in obtaining the transmission medium and is highly dependent on the access protocol. Various efforts have been made to establish a limit for the

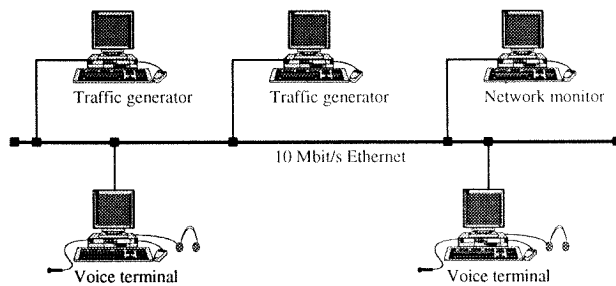


Fig. 1 - System configuration used during the experimental study.

maximum acceptable delay but, due to the subjective nature of the matter, there is no unanimous opinion globally accepted. For conversations within a LAN, delays between 100 and 200 ms are tolerable [12].

Another important issue in packet voice systems is the amount of speech present in each packet. Large packets increase effective bandwidth by reducing the probability of collisions in CSMA/CD based systems. However, it increases the delay's fixed component and the amount of speech lost per packet may become unacceptable. Packet length must result from a compromise between overall delay, system performance, and the amount of speech lost per packet.

B. Review of Ethernet's principles of operation

Ethernet uses *carrier sense multiple access with collision detection* (CSMA/CD) to allow multiple stations to share a single broadcast bus. If a transmitting station detects a collisions, it immediately aborts transmission, sends a jam sequence, and then reschedules its packet for transmission at a random time calculated according to a truncated binary exponential backoff algorithm. Detailed descriptions of these protocols may be found in [13, 14]. CSMA/CD based networks are known to give equal access opportunities to all stations. In environments where voice and data coexist, this is obviously not the best approach. In the presence of data traffic, voice stations should have priority over data stations.

III. PROTOTYPE DESIGN AND IMPLEMENTATION

A. System configuration

In order to further understand the problems involving a synchronous packet voice communication system over a non deterministic multiple access network, a prototype, that implements a 64 kbit/s full-duplex voice channel, was developed. It currently consists of two voice terminals, two traffic generators, and a network monitor as illustrated in Fig. 1.

Each subsystem is individually implemented on a PC and, in each, the network interface board used was the WD8003E [15]. This board contains an 8 Kbyte on-board PC-addressable memory buffer, and is compatible with IEEE 802.3/Ethernet standards. The voice board used was

the Sound Blaster Pro version 2.0. The Traffic Generator as well as the Network Monitor, used exhaustively during this experimental study, belong to a set of network traffic analysis tools developed at our University [16, 17].

B. System operation

The functional basis of a full-duplex communication system are elementary. A conversation consists of two independent and identical simplex voice channels, each of which contains a packet voice transmitter and a packet voice receiver. In the remainder of this section, we describe the principles of operation of each of these subsystems. We also discuss briefly the data traffic generator and network monitor. Finally, some attention to the problem of clock drift shall be given.

1) *Packet Voice Transmitter*: Speech is digitized at a rate of 8 kHz using 8 bits/sample. Samples are gathered into an intermediate transmission buffer until a packet is filled, after which it is transferred to the network interface buffer and submitted for transmission. Although Ethernet specifies 16 as the maximum number of collisions a packet may suffer, any packet submitted for transmission, that experiences multiple collisions, is continuously rescheduled until it is either successfully transmitted or a new packet of voice samples is collected. If the latter occurs, the older packet is discarded and the new one is transmitted.

The intermediate transmission buffer is implemented with a circular FIFO (First In First Out) buffer with capacity for two voice packets. This buffer is handled by two pointers. A write pointer points to where the next packet to be acquired is to be put and the other, a read pointer, keeps track of the next packet to be sent to the network interface buffer.

At this stage, silence suppression has not yet been implemented. No error recovery mechanisms were contemplated due to real-time constraints. Suppose the receiving station detects that something went wrong (ex: a voice packet is missing); it would then be too late to formulate a request for packet retransmission. We rely on the knowledge of the good transmission characteristics of the Ethernet to achieve good results. The only additional information a voice packet contains is its sequence number. It is used, by the receiver, for statistical purposes and as a means for detecting lost packets.

2) *Packet Voice Receiver*: Packets whose destination address is that of the receiving station, and are identified as voice packets, are transferred from the network interface buffer to an internal receive buffer. Voice samples are then taken from the internal receive buffer and played back through the voice board's DAC (digital to analog converter). Voice packets are generated and submitted for transmission at regular intervals. The mechanism responsible for speech reconstruction, at the receiver, must compensate for variable delay by inserting a certain delay D before playing back each packet. This delay is limited by the maximum end-to-end delay, D_{max} ,

and by the maximum percentage of packets that can be lost at the transmitter or rejected by the receiver.

There are essentially two methods for speech reconstruction in a packet voice system [18, 19]. The first, and most simple, makes use of no timing information to determine variable packet delay as it traverses the network. The receiver introduces a fixed artificial delay D before the reproduction of the first packet of a conversation, if silence detection is not implemented, or before the first packet of each talkspurt, if silence detection is implemented, and the following packets are played back at intervals of D_g (packet generation time) after the first. If a packet is not present when it is scheduled for playback, it is discarded and a silence packet is played back instead, as shown in Fig. 2 a). The choice of D results from a trade-off between the percentage of lost packets and the maximum allowable end-to-end delay. Increasing D reduces packet loss but increases the overall delay as well as the receiver's intermediate buffer size. This method is suitable for environments in which the variance of the delay's variable component is small.

The second approach makes use of timing information, in the form of timestamps, to determine each packet's delay through the network. Montgomery [18] suggests that the first method is more appropriate for intra LANs applications where variable delay as well as other fixed delays are relatively small. For inter LANs applications, where variable delays may be rather significant, this method no longer produces satisfactory results and a method with more timing information is needed.

The reconstruction process, here presented, is based on the first method with some slight modifications. As before, the first packet of each conversation is delayed by D and the others are played back at intervals D_g after the first. D is here implemented by starting the speech reproduction mechanism after the arrival of the second, third, ..., n^{th} packet (depending on the delay D to be introduced). In opposition to the first method, if a packet arrives at the receiver after its scheduled playback time it is accepted and is immediately sent for reproduction, Fig. 2 b). We believe that for small mean variable delays with equally small variance, it is best to accept the late coming packets for playback than to reject them, this because the silence interval introduced due to the receiver's intermediate

buffer underflow is significantly smaller than D_g . The advantage of this method is that, in many cases, the listener will not become aware of speech interruption. It is Musser's opinion [6] that a long silence gap is very disconcerting to the listener and that absolute silence causes the listener to immediately detect that something is missing.

A second variant of this non discarding reconstruction algorithm was implemented: a late arriving packet is scheduled for playback D_g after receiver's intermediate buffer underflow, as can be seen in Fig. 2 c). This method has the advantage of adjusting D every time the receiver's intermediate buffer becomes empty. It acts as though the reproduction mechanism is restarted with D' as the new artificial delay. The interest in restarting the reproduction mechanism is related with the fact that being D an estimate of the delay encountered by the packets before speech reproduction is initiated, it could be a worst case assumption. This reconstruction method allows regular revisions of this delay and proved to be more robust than the previous method. All results presented were obtained using the second method and speech reproduction initiated after the arrival of the second packet.

3) *Clock Drift*: In any CSMA/CD based communication system there are two unavoidable causes responsible for variable packet spacing: transmission delay and clock drift. Variable transmission delay, caused by CSMA/CD access protocol, is intimately related to the number of collisions a packet suffers before it is correctly transmitted. Only for relatively high loads does transmission delay become dominant.

Clock drift is a less obvious cause for losing speech samples and is always present, no matter what the load on the network. If the receiver's clock runs slightly slower than that of the transmitter, packets will accumulate at the receiver's buffer and it will eventually overflow. In the opposite case, if the receiver's clock runs slightly faster than that of the transmitter, the receiver's buffer will underflow and breaks will be introduced in the conversation. To bypass this problem, we introduced feedback into the receiver's reproduction frequency. It consists on varying the reproduction frequency between a high and a low threshold. As an example, suppose that speech reproduction is initiated after the arrival of the second packet, the lower threshold (NPktMin) is 1, the higher threshold (NPktMax) is 3, and the initial reproduction frequency is f_{si} . Every time a new packet is scheduled for playback, the number of packets present in the receiver's buffer is compared to the lower and higher thresholds. If it is smaller than or equal to NPktMin the new reproduction frequency is set to:

$$f_{sn} = f_{si} - Df, Df \in [0, Df_{\max}]$$

On the other hand, if it is greater than NPktMax the new reproduction frequency is set to:

$$f_{sn} = f_{si} + Df, Df \in [0, Df_{\max}]$$

This approach, besides solving the clock drift problem, introduces, as side effect, another interesting behaviour. If there is a sudden raise of network traffic, packet

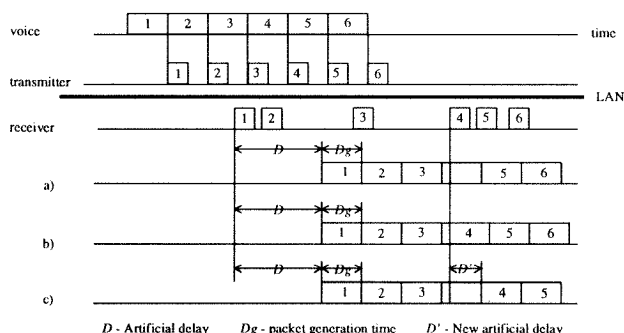


Fig. 2 - Speech reconstruction methods using no timing information.

transmission delay increases. This results in the receiver's buffer being gradually emptied. However, as soon as the number of packets in the buffer goes below NPktMin the reproduction frequency also decreases, thus diminishing the increased packet arrival delay.

The above procedure did indeed eliminate the clock drift problem but gave place to the following questions: What is the maximum value for D_f without significantly distorting the reconstructed voice signal and what are the optimum values for NPktMin and NPktMax? Informal tests were conducted with a special application, which played back a previously recorded message at different user selectable frequencies, to determine the maximum and minimum frequencies at which the differences between the message reproduced at 8 kHz and at the new frequency are negligible. Test results indicate that a fluctuation of 4% around 8 kHz is perfectly acceptable. The choice of NPktMin and NPktMax are directly related with whatever value is attributed to D and to the receiver's intermediate buffer size. Anyway, the minimum value for NPktMin is one and should be, at least, one less than the number of packet arrived before speech reproduction is started. NPktMax should be at least NPktMin+2, so that frequency adjustments aren't always taking place, and smaller than the receiver's intermediate buffer size.

IV. EXPERIMENTAL RESULTS

A large number of tests were conducted in order to determine transmission and reception algorithm's robustness as well as Ethernet's performance as a means for voice and data transmission. Each measurement had an approximate duration of 3 minutes, since this seems a reasonable value for an average telephone call duration. The various tests were executed for packet sizes of 64, 128, and 256 voice samples each, which correspond to 8, 16, and 32 ms of voice, respectively. In all cases, the system was found to provide voice quality service comparable to that of a regular telephone network. Most of the measurements took place in an isolated network where all machines attached were, somehow, related to the tests which were taking place. In this closed environment it is easier to isolate the specific parameter of interest for a particular study. The most important performance metrics used during this experimental study are: percentage of packets lost at the transmitter; average transmission delay - defined to be the time from the moment a packet enters the intermediate transmission buffer to the arrival of transmission acknowledge (successful or not); average access delay - defined to be the time from the moment a packet is submitted for transmission to the arrival of

transmission acknowledge (successful or not); average normalized network load - (n° of bits received/s)/(10 Mbit/s).

We first present system performance in the presence of voice traffic only. Next we analyse voice/data integration in an Ethernet network. This is followed by a detailed discussion of the effects that certain parameters introduce in global system performance.

A. Voice traffic only

In order to estimate the maximum allowable number of active voice stations, in the absence of data traffic and without incurring intolerable packet loss and excessive delay, the following assumptions were made: each traffic generator contributes with a synchronous load (simulating speech traffic) that is an integer multiple of that produced by a real voice channel; we consider that each channel's behaviour is identical to that of a real voice channel, i. e., the percentage of lost packets and the average packet delay is the same for all synchronous channels; the estimate of the number of voice channels is given by the ratio between the average network load and the load of a single voice channel. Fig. 3 shows channel degradation, as the percentage of voice packets discarded at the transmitter, when 4 hosts were contending for the transmission channel. Considering 2% as the maximum limit for packet loss [11], the value of 98 voice channels for 32 ms packets differs from the analytical estimation, presented by Friedman [8], in less than 10%.

Packet loss as a function of network load is illustrated in Fig. 4. Note that packet loss is negligible for loads up 60% and that network performance increases with packet size.

Fig. 5 shows the average transmission delay that voice packets experience. Note that delay is essentially constant for loads up to 60% and that thereafter rises exponentially. For high loads, retransmission frequency, during the vulnerable propagation period, is such that a good portion of transmission trials end up in a collision. This increases delay due to collision resolution which is comparatively larger for smaller packets. If to these delays we add packet acquisition time and the artificial delay introduced at the receiver, thus obtaining the end-to-end delay, we find that it is well within the acceptable limits (100-200 ms) [12].

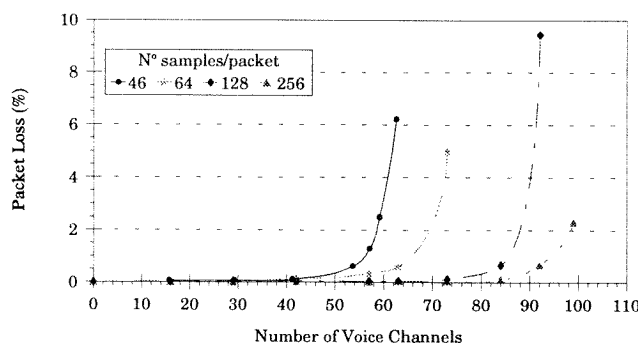


Fig. 3 - Voice channel degradation with increasing number of simultaneous calls for voice traffic only.

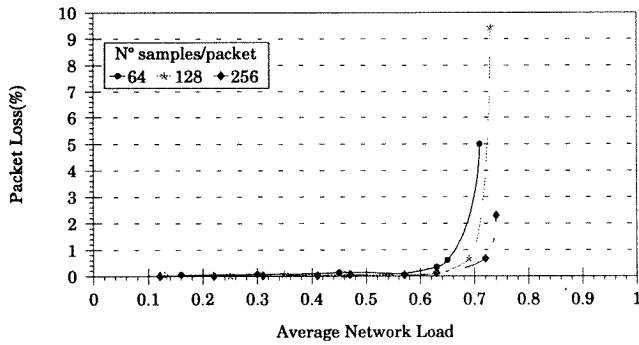


Fig. 4 - Voice channel degradation with increasing load for voice traffic only.

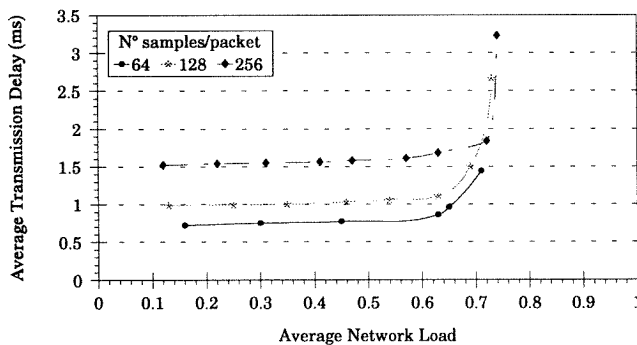


Fig. 5 - Transmission delay with increasing load (voice traffic only).

B. Integrated voice/data traffic

Several tests were performed in order to analyse the behaviour of a voice channel in the presence of real data traffic. Table I collects some performance results of tests made at our University's network. It is a good example of a real data environment with over a hundred different types of equipment running all sorts of applications. Packet loss as well as average transmission delay are small. The establishment of a conversation in such an environment, where data applications are predominant, rises no problems.

The natural question to be asked is how does the voice system respond to a larger data load? A private network with 5 PCs, where data stations generated traffic with an uniformly distributed packet length between 64 and 1500 bytes and packet frequency generation was uniformly distributed to produce a predefined average load, was used in the tests that follow. Packet loss and average transmission delay, for one voice channel, are illustrated in Figs. 6 and 7, respectively. Channel degradation, in the form of lost packets, is negligible for loads up to 55% and is highly dependent on packet size for higher loads. A comparative analysis of Figs. 4, 6 and 5, 7 shows that, in both cases, system performance is identical concerning packet loss and average transmission delay.

Table I
Voice channel performance in a real data traffic environment.

Samples per Packet	Ave. load (%)	Packet loss (%)	Ave. tx delay (ms)
64	3.48	0.00	0.74
	4.38	0.00	0.75
	4.99	0.01	0.75
	7.05	0.00	0.76
	16.57	0.04	0.83
128	3.33	0.01	0.99
	4.20	0.00	0.99
	5.53	0.00	1.01
	8.69	0.02	1.02
	18.27	0.15	1.14
256	4.66	0.00	1.53
	5.23	0.00	1.52
	5.79	0.00	1.53
	7.95	0.02	1.55
	15.08	0.00	1.66

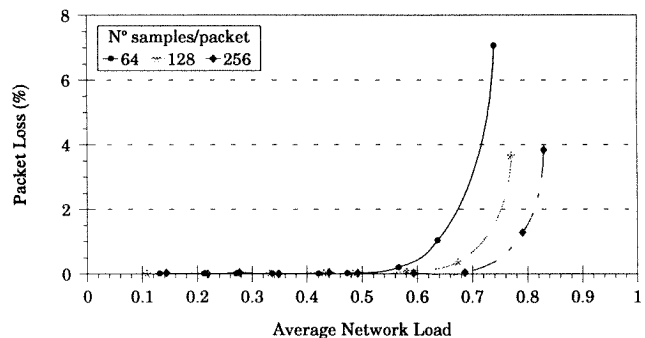


Fig. 6 - Voice channel degradation with increasing load in an integrated voice/data environment.

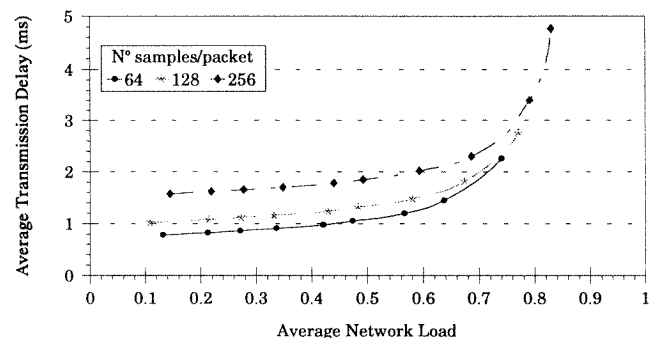


Fig. 7 - Transmission delay with increasing load in an integrated voice/data environment.

The establishment of voice conversations for loads greater than 65-70% is not recommended because at these loads delay rises exponentially and real-time constraints may no longer be kept. The extent to which voice and data may be integrated in an Ethernet is similar to formulating

the following question: How many voice users may be attached to the net, in the presence of data traffic, while preserving minimum voice quality. In order to estimate this value, part of Ethernet's bandwidth was put aside for data and the rest for voice. The estimates presented in Table II were obtained for 2% of lost packets and data loads of 0, 10, and 35%. Note that the end-to-end delays (not shown) are well within the acceptable limits.

C. Effects of other parameters

1) *Influence of active number of hosts:* The above results were all obtained with a relatively small number of machines disputing transmission channel access. From channel efficiency's point of view, the small number of hosts responsible for generating the total load is not so important, once that, for more than 5 machines, overall system performance is not as sensitive to the number of active hosts as it is to the total offered load [20]. However, when we refer to the delay, the same cannot be said. The delay a packet suffers before it is successfully transmitted is intimately related to number of collisions it undergoes, and these depend on the number of machines simultaneously accessing the transmission channel as illustrated in Figs. 8 and 9 (curves identified by markers on the left of the legend were obtained when 4 machines were disputing network access, the others were obtained with 3 machines). The number of host's influence is more evident

Table II
Maximum number of voice users for different data loads.

N° Samples per Packet	0 % Data	10 % Data	35 % Data
	Approx. N° simult. channels	Approx. N° simult. channels	Approx. N° simult. channels
64	70	67	38
128	88	86	48
256	98	96	61

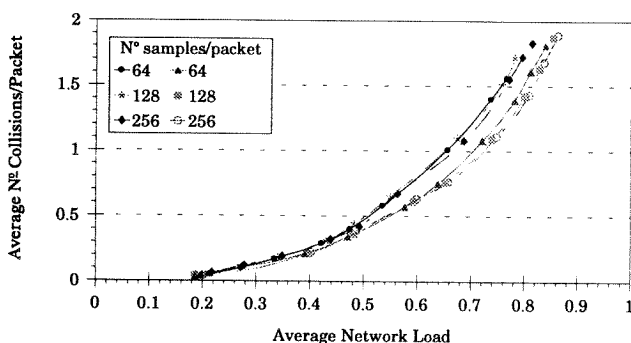


Fig. 8 - Average number of collisions with increasing load when total load is generated by 3 and 4 machines.

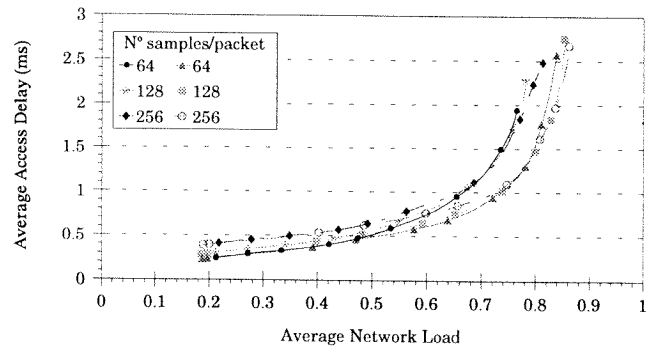


Fig. 9 - Average network access delay with increasing load when total load is generated by 3 and 4 machines.

for loads greater than 50%.

Studies of voice/data traffic on CSMA/CD networks are abundant in the literature. However, the different assumptions and performance metrics used render comparisons difficult. In order to validate the above results, concerning packet loss and delay, some tests were made using, whenever possible, the same performance metrics and implementation characteristics as those presented in the literature. The results in Figs. 3 and 10, obtained in the absence of data traffic and for 5.75 ms voice packets, are comparable to those presented by [1]. Overall system performance, of our implementation, involving voice/data integration produced apparently better results (not presented here) than those presented by [1].

Tests involving a larger number of hosts were not conducted due to the limited number of machines available. However, it is our believe that if the total offered load is distributed by a larger number of hosts, the knee of the various delay and packet loss curves will be slightly shifted to the left.

2) *Buffer Size:* It is possible to increase Ethernet's effective bandwidth by increasing packet size or arranging for larger buffers. Packet size acts directly on the fixed component of the overall delay and indirectly on its variable component. To act directly on the delay's variable component, responsible for most speech loss at high loads, several tests were run with different transmission buffer sizes. The results obtained were as expected and they show a slight increase in system performance at high loads

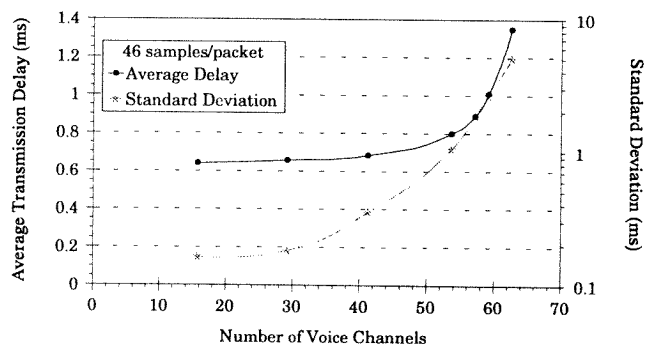


Fig. 10 - Average transmission delay and standard deviation with increasing number of voice channels (voice traffic only).

(for loads of up to 60% the extra buffer size is of little use - packet loss and delay are essentially the same as illustrated in Figs. 11 and 12). However, this improvement is less significant than that obtained when packet size was increased. When we doubled packet size, there was an improvement in packet performance of nearly 7% at 2% of lost packets. When buffer size was doubled, the improvement was less than 2% for the same packet loss (note that in both cases the maximum transmission delay is the same). This suggests that, of the total allowable delay through the system, more delay should be put apart for packet size than for transmission buffer. Similar results are reported in other studies [1].

V. VOICE CHANNEL IMPLEMENTATION OVER TCP/IP

Two major reasons lead to the implementation of a voice channel using the TCP/IP protocols. First, the communication system becomes independent of the network's hardware thus, allowing the establishment of a conversation across two machines no matter the diversity of the underlying hardware technologies. Secondly, it allows the user to simultaneously possess, on the same machine, the vast range of Internet applications (electronic mail, file transfer, remote login, ...) as well as the voice communication system.

A full-duplex voice channel, based on the previously presented transmission and reception algorithms,

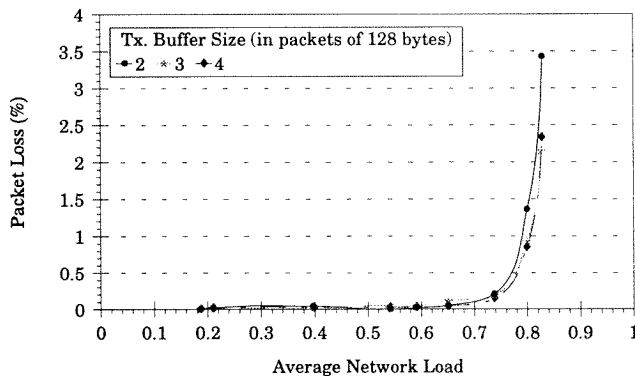


Fig. 11 - Voice channel degradation with increasing load in an integrated voice/data environment for different transmission buffer sizes.

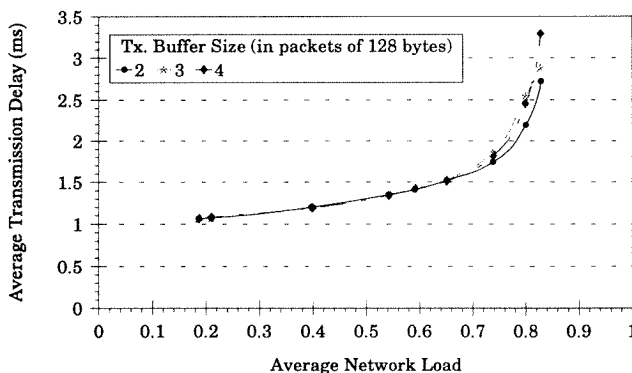


Fig. 12 - Transmission delay with increasing load in an integrated voice/data environment for different transmission buffer sizes.

compatible with the User Datagram Protocol (UDP) was implemented. UDP provides an unreliable connectionless delivery service using the Internet Protocol (IP) to transport messages between machines. It also adds the ability to distinguish among multiple destinations within a given computer.

Similar test, to those presented earlier, were carried out with this voice communication system implementation. The most important result to retain is that such an implementation is possible and practicable. Global system performance does not differ much from that of the voice channel's implementation over the logic link layer, and the conclusions at which we then arrived are here also valid.

VI. SUMMARY AND CONCLUSIONS

An experimental study of a full-duplex packet communication system was presented. The most important conclusion is that voice/data integration is possible in an Ethernet based network, with no access protocol specifications changes needed, as long as it is not overloaded. A voice channel, when submitted to loads of 65 to 70%, suffers negligible degradation in terms of packet loss and overall delay and the quality of the reconstructed signal is acceptable. For such loads, the variable component of the delay is small as well as its variance producing an end-to-end delay well within the acceptable limits of 100-200 ms.

Results show that network performance increases with packet size and that, in a real traffic environment, a voice communication system is possible with less than 2% of lost packets. Packets of 32 ms proved to be the optimum packet size. This value represents a good compromise between the percentage of packet loss, which is inversely proportional to packet length, and the probability of losing a complete phoneme, which rises rapidly for packets larger than 20 ms. The estimate of the maximum number of voice channels, in the absence of data traffic, points to a little less than 100 channels for 32 ms packets and 2% of lost packets.

The reconstruction algorithms, here implemented, proved to be an attractive alternative to the usual method for networks where the mean delay and variance are small.

Introducing feedback into the reproduction frequency, as a means to eliminate information loss due to clock drift, proved to be advantageous. No voice quality degradation results from the variable reproduction frequency.

Finally we find that the Sound Blaster, used as a voice board, is not at all adequate for such a communication system. A custom-designed voice board is necessary to achieve best results.

ACKNOWLEDGEMENT

We wish to thank the Portuguese R&D agency JNICT who supported the MSc work, under grant BM/1770/91-IA, on which this article is based upon.

REFERENCES

- [1] J. D. DeTreville, "A Simulation Based Comparison of Voice Transmission on CSMA/CD Networks and on Token Buses", AT&T Bell Laboratories Technical Journal, vol. 63 No.1, Jan. 1984, pp. 33-55.
- [2] D. H. Johnson, G. C. O'Leary, "A Local Access Network for Packetized digital voice communication", IEEE Transactions on Communications, vol. COM-29, May 1981, pp. 679-688.
- [3] John F. Shoch, "Carrying Voice Traffic Through an Ethernet Local Network, a General Overview", Proc. IFOP WG 6. F Int. Workshop Local Area Comput. Networks, Zurich, Aug. 1980.
- [4] Gary J. Nutt, Douglas L. Bayer, "Performance of CSMA/CD Networks Under Combined Voice and Data Loads", IEEE Transactions on Communications, vol. COM-30, Jan. 1982, pp. 6-11.
- [5] Timothy A. Gonsalves, Fouad A. Tobagi, "Comparative Performance of Voice/Data Area Networks", IEEE Journal on Selected Areas in Communications, vol. 7, No. 5, Jun. 1989, pp. 657-669.
- [6] J. M. Musser, T. T. Liu, G. J. Boggs, "A Local Area Network as a Telephone subscriber loop", IEEE Journal on Selected Areas in Communications, vol. SAC-1, Dec. 1983, pp. 1046-1054.
- [7] Eluzor Friedman, Chaim Ziegler, "Real-Time Voice Communications Over a Token-Passing Ring Local Area Network", in Proc. SIGCOM '86 Symp., Aug. 1986, pp. 52-57.
- [8] Eluzor Friedman, Chaim Ziegler, "Packet Voice Communications Over PC-Based Local Area Networks", IEEE Journal on Selected Areas in Communications, vol. 7, No. 2, Feb. 1989, pp. 211-218.
- [9] J. DeTreville, W. D. Sincoskie, "A Distributed experimental Communication System", IEEE Journal on Selected Areas in Communications, vol. SAC-1, Dec. 1983, pp. 1070-1075.
- [10] Stephen Michaels, "Ten Times Faster", Communications International, vol. 20, No. 4, Apr. 1993, pp. 9-12.
- [11] Nugeghally S. Jayant, Susan W. Christensen, "Effects of Packet Losses in Waveform Coded Speech and Improvements Due to an Odd-Even Sample-Interpolation Procedure", IEEE Transactions on Communications, vol. COM-29, No. 2, Feb. 1981, pp. 101-109.
- [12] Daniel Minoli, "Optimal Packet Length for Packet voice Communication", IEEE Transactions on Communications, vol. COM-27, No. 3, Mar. 1979, pp. 607-611.
- [13] ANSI/IEEE Std 802.3-1985 - Carrier Sense Multiple Access with Collision Detection Access Method and Physical Layer Specifications, IEEE, New York, May 1986.
- [14] Robert M. Metcalfe, David R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks", Communications of the ACM, vol. 19, No. 7, Jul. 1976, pp. 395-404.
- [15] WESTERN DIGITAL, Ethernet PC Adapter Boards WD8003E, Hardware installation guide.
- [16] Joaquim S. Pinto, José L. Oliveira, Joaquim A. Martins, "PC Workstation for Ethernet Monitoring and Management", International Telecommunications Symposium, Set. 1990, pp. 568-570.
- [17] Joaquim A. Martins, Joaquim S. Pinto, José L. Oliveira, "Measuring, Characterizing and Generating LAN Traffic", Conference Proceedings, Singapore ICCS '90, vol. 2, Nov. 1990, pp. 1116-1119.
- [18] Warren A. Montgomery, "Techniques for Packet Voice Synchronization", IEEE Journal on Selected Areas in Communications, vol. SAC-1, No. 6, Dec. 1983, pp. 1022-1028.
- [19] Thomas M. Chen, Jean Walrand, David G. Messerschmitt, "Dynamic Priority Protocols for Packet Voice", IEEE Journal on Selected Areas in Communications, vol. 7, No. 5, Jun. 1989, pp. 632-643.
- [20] John F. Shoch, Jon A. Hupp, "Measured Performance of an Ethernet Local Network", Communications of the ACM, vol. 23, No. 12, Dec. 1980, pp. 711-721.

