

Uma arquitetura modular de *hardware* e *software* para PABX VoIP baseado em Asterisk

Marcos A. P. Cunha, Karinne S. Silva, David F. M. Mota e Alex A. Vasconcellos

Resumo—Este artigo apresenta o desenvolvimento de uma central modular PABX VoIP baseada em Asterisk. Foram construídas três placas de circuito impresso e realizado o *porting* de distribuições Linux, de *drivers* e da plataforma Asterisk. Na primeira parte deste trabalho serão abordados alguns conceitos sobre estruturas de comunicação telefônica. Em seguida, serão apresentadas as arquiteturas de *hardware* e de *software* embarcado utilizadas. Ao final do artigo, será apresentado o produto final desenvolvido e os resultados dos testes de certificação aos quais o produto foi submetido.

Palavras-Chave—PABX, VoIP, Asterisk, Linux Embarcado.

Abstract—This paper presents the development of a modular PABX VoIP system based on Asterisk. Three printed circuit boards were design and the porting of Linux distributions, drivers and Asterisk platform was performed. The first part of the paper will discuss some concepts on telephone communication structures. Then, we present the hardware and embedded software architectures. At the end of the paper the final product developed and the certification test results which the product was subjected will be presented.

Keywords—PABX, VoIP, Asterisk, Embedded Linux.

I. INTRODUÇÃO

Este artigo apresenta a proposta de uma solução PABX baseada em VoIP. O *hardware* deste projeto foi desenvolvido de forma modular e é composto por três placas. A arquitetura do *software* embarcado também foi alvo do projeto.

As três placas de circuito impresso que compõem esta arquitetura são: Placa-mãe, onde é realizado o processamento de rede e tratamento dos protocolos VoIP, como SIP; Placa-filha E1, que possibilita conexão com outras centrais telefônicas através da interface E1; e a Placa-filha FXS, que possibilita a conexão de até 8 ramais analógicos FXS.

A arquitetura de software proposta é composta por três partes principais: sistema operacional Linux, *drivers* para controle das interfaces de hardware e servidor PABX baseado em Asterisk. Além de ferramentas de controle necessárias para realizar a comunicação entre as placas existentes.

II. ESTRUTURA DE COMUNICAÇÃO TELEFÔNICA

A. PSTN – Public Switched Telephony Network

PSTN (em português, rede telefônica pública comutada) é a rede telefônica comutada por circuitos, destinada ao serviço de telefonia tradicional.

No uso da comutação de circuitos, um canal é reservado durante toda ligação, provendo a qualidade de serviço

necessária para transmissão de voz. Em geral, este serviço é prestado pelas companhias de telefonia e tem um custo embutido, mais detalhamento sobre o funcionamento das redes comutadas pode ser encontrado em [1].

Apesar da PSTN estabelecer chamadas de voz entre dois terminais de forma eficiente, há uma grande demanda por novas funcionalidades e menor custo. Isto está levando inúmeras empresas a mudarem seus sistemas de telefonia tradicionais para uma tecnologia baseada em comutação de pacotes, o VoIP.

B. VoIP

A tecnologia de transmissão de Voz sobre o protocolo IP (VoIP) tem sido largamente utilizada no meio corporativo com o objetivo de diminuir custos com ligações interurbanas e internacionais.

O VoIP converte o sinal de voz do telefone em um sinal digital e o envia através da internet usando os protocolos UDP/IP, quando o sinal chega ao destino ele é convertido de volta para a forma analógica. Para utilização de um serviço VoIP o usuário precisa de um provedor de serviço e um telefone IP ou *softphone*.

III. ASTERISK

O Asterisk [2] é um *software* livre que disponibiliza o serviço de um PABX completo utilizando tecnologia VoIP. Inicialmente ele foi desenvolvido pela empresa Digium [3], mas atualmente é mantido através de contribuições de desenvolvedores e usuários espalhados pelo mundo.

Através do Asterisk é possível utilizar todas as funções das centrais telefônicas convencionais, como Unidades de Resposta Automática (URAs), correio de voz, conferência, distribuição automática de chamadas entre outras.

Para realizar integrações com a rede de telefonia convencional, o Asterisk pode ser conectado através de *hardwares* específicos com interfaces FXO, FXS, E1 ou T1. Uma alternativa é utilizar placas fabricadas pela empresa Digium. Tradicionalmente, o Asterisk é instalado em um PC (*Personal Computer*) com sistema operacional Linux, sendo possível instalar uma ou mais placas com interface PCI (*Peripheral Component Interconnect*) para integração com a rede telefônica tradicional.

A arquitetura do Asterisk é apresentada na Figura 1. Onde são exemplificadas algumas aplicações disponibilizadas ao usuário. Os canais consistem em módulos de interfaces de comunicação lógica [4]. Neste projeto foram utilizados os canais SIP e DAHDI.

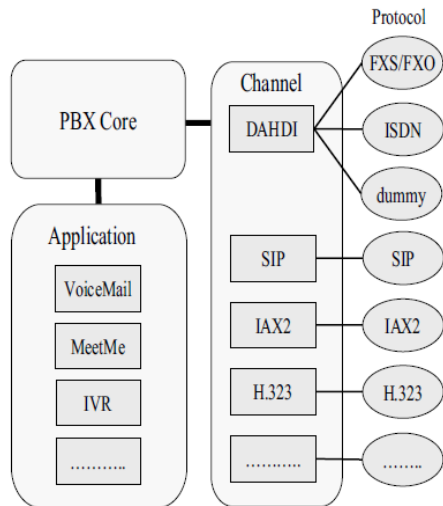


Fig. 1. Arquitetura do Asterisk.

A. SIP

O SIP (*Session Initiation Protocol*) é o protocolo de sinalização utilizado na telefonia IP, sendo baseado em texto, ele é bastante similar ao HTTP (*Hypertext Transfer Protocol*).

Algumas das funcionalidades disponibilizadas através do SIP são: localização de usuários, estabelecimento de chamadas, suporte a *unicast* ou *multicast*, transferência de chamadas, conferência, dentre outros.

B. DAHDI

O DAHDI (*Digium/Asterisk Hardware Device Interface*) é um *driver* utilizado para controlar as placas de interface da Digium. A conexão estabelecida através do DAHDI é denominada *Span* e utiliza o protocolo TDMoE (*TDM over Ethernet*) para realizar transferência de dados [5].

Neste projeto, o DAHDI, versão 2.2.0.2, foi utilizado para manter uma conexão entre a Placa-mãe e as placas-filhas E1 e FXS utilizando um tipo especial de *Span*, o *dynamic*.

O *Span dynamic* é utilizado para estabelecer conexões através de uma interface *ethernet* e sua configuração é realizada através do endereço MAC do equipamento com o qual se deseja conectar.

Para estabelecer uma conexão com sucesso, é necessário que as duas pontas da conexão tenham seus *Spans* direcionados para a outra ponta.

IV. ARQUITETURA DE HARDWARE PROPOSTA

Para este projeto, foi elaborada uma plataforma de *hardware* que consiste no desenvolvimento de três arquiteturas distintas que se comunicam entre si através de interface *ethernet*. Neste contexto, o desenvolvimento da plataforma teve como base o projeto de uma placa de circuito impresso (*Printed Circuit Board - PCB*) denominada de Placa-mãe no qual é concentrado todo o processamento de rede que provê a funcionalidade de IP-PABX ao sistema.

Com o intuito de estender as aplicações da plataforma, foram desenvolvidas outras arquiteturas que resultaram na criação de duas novas PCB's para disponibilização das interfaces E1 e FXS. As três arquiteturas interligadas permitem que a plataforma de *hardware* tenha as funcionalidades de um

PABX com interfaces analógicas e comunicação digital com centrais telefônicas.

A. Placa-mãe

A Placa-mãe desenvolvida neste trabalho é um sistema embarcado capaz de processar pacotes de rede com velocidades de 10/100/1000Mbps e é baseada no processador da família PowerQUICC II Pro da Freescale Semiconductor cuja arquitetura interna consiste no *core* PowerPC de 32bits. O sistema possui duas memórias RAM do tipo DDR2 que juntas somam uma capacidade de armazenamento de dados de 256MB, uma memória ROM do tipo Flash com capacidade de 32MB, um conector para cartão memória do tipo não volátil e um conector de interface SATA para conexão com discos rígidos.

As interfaces que permitem a aplicação de um IP-PABX ao sistema são dois conectores de rede RJ-45, um com funcionalidade do tipo *Fast Ethernet* (100Mbps) e outro *Gigabit Ethernet* (1000Mbps). Como interface de comunicação com o usuário, são disponibilizadas entradas USB *Host* e USB *Device* para fins de armazenamento de informação em *pendrives* e disco rígidos portáteis que suportem USB.

Para efetuar a comunicação com as PCB's de interface E1 e FXS, existem dois conectores do tipo *mezzanine*, onde são disponibilizados para as placas-filhas sinais de controle, comunicação e alimentação.

A arquitetura de *hardware* da Placa-mãe pode ser observada na Figura 2.

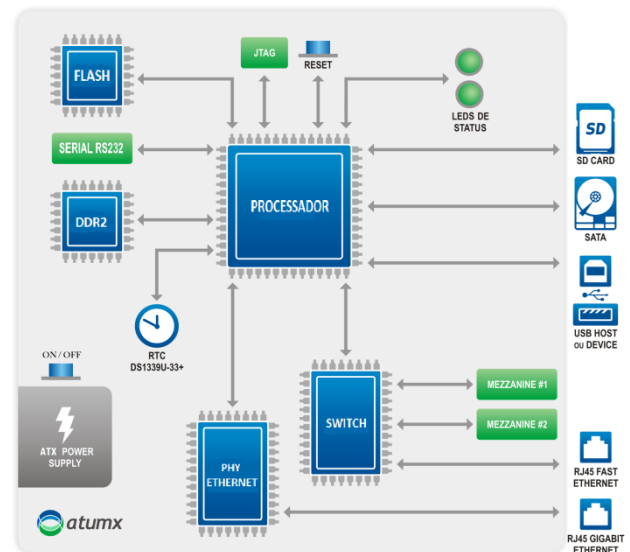


Fig. 2. Arquitetura de hardware da Placa-mãe.

B. Placa-filha E1

A Placa-filha E1 é utilizada para estabelecer conexão entre centrais telefônicas de forma digital, permitindo até 30 chamadas simultâneas, de usuários VoIP ou FXS. A unidade de processamento desta solução é um DSP (*Digital Signal Processor*) da família Blackfin da empresa Analog Devices com uma arquitetura interna de 16bits, podendo ser utilizado tanto em aplicações de processamento de voz, áudio e vídeo como em aplicações de uso geral. Para incorporar um sistema de computação completo, foi utilizada uma memória RAM do tipo SDRAM de 32MB e uma memória não volátil do tipo *Flash* de 4MB.

A interface E1 presente nesta solução utiliza um circuito integrado dedicado conhecido como E1 *Framer*. Este dispositivo tem a função de disponibilizar a camada física E1, prover e gerenciar a sinalização, como também, disponibilizar estatísticas a respeito do *link* E1, por exemplo, escorregamento [6]. A conexão desta interface é feita através de um conector RJ-45.

Este sistema embarcado faz uso da interface *ethernet* para comunicar-se com a Placa-mãe através do conector *mezzanine* descrito no item anterior.

A arquitetura de *hardware* da Placa-filha E1 pode ser observada na Figura 3 a seguir.

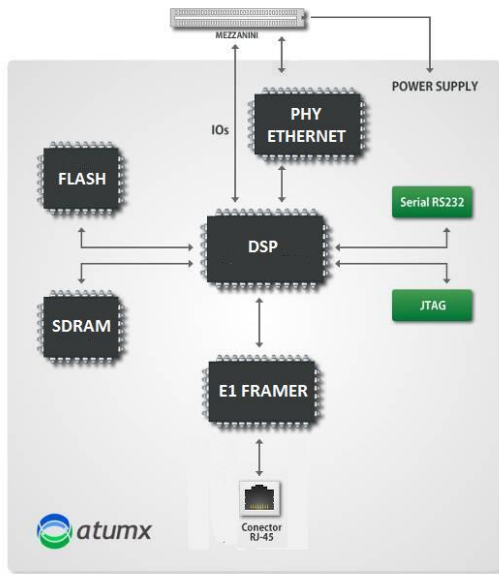


Fig. 3. Arquitetura de *hardware* da Placa-filha E1.

C. Placa-filha FXS

A interface FXS (*Foreign eXchange Subscriber*) fornece a linha analógica que é utilizada para conexão com telefones tradicionais.

Para a elaboração desta plataforma, foi utilizado o mesmo sistema de computação presente na Placa-filha E1 (processador e memórias) e a mesma conexão com a Placa-mãe pelo conector *mezzanine*. A reutilização desta arquitetura foi bastante benéfica devido à redução de custos com desenvolvimento do núcleo de processamento, sendo os esforços dedicados à conexão entre o DSP e o circuito integrado que disponibiliza a interface FXS.

O dispositivo SLAC (*Subscriber Line Audio-processing Circuit*), que compõe a interface FXS, possui características como suporte a *codecs* lineares e filtros *a-law* e *μ-law*, geração de tom DTMF e FSK, e padrão PCM de interface digital [7].

O SLIC (*Subscriber Line Interface Circuit*) que integra a solução FXS em conjunto com o SLAC é responsável pela transmissão da voz, por gerar o sinal de *ring*, pela detecção de falhas e linhas cruzadas [8].

Para conexão com oito telefones analógicos são disponibilizados dois conectores RJ-45 cada um deles com quatro canais FXS.

A arquitetura de *hardware* da Placa-filha FXS é apresentada na Figura 4.

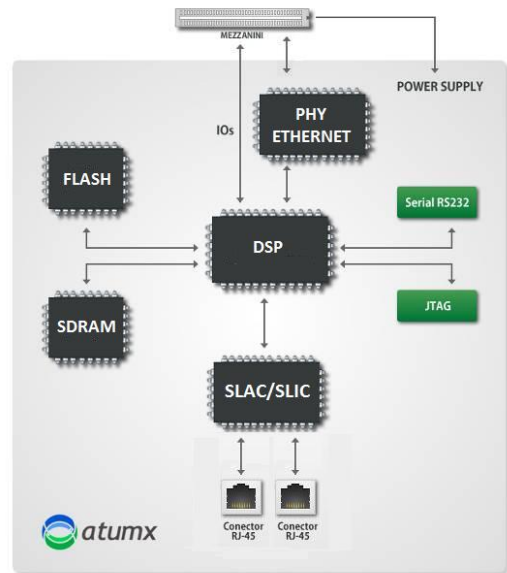


Fig. 4. Arquitetura de *hardware* da Placa-filha FXS.

V. ARQUITETURA DE SOFTWARE PROPOSTA

As três placas desenvolvidas possuem unidades de processamento independentes e específicas para cada aplicação. Neste contexto, faz-se necessária a utilização de sistemas operacionais customizados e conjuntos de aplicativos específicos para cada placa.

A seguir, são apresentados os principais pontos de customização e as ferramentas utilizadas.

A. Linux Embarcado

Os sistemas baseados em Asterisk geralmente são utilizados em conjunto com o sistema operacional Linux e uma plataforma *Personal Computer* (PC). A arquitetura proposta também faz uso do sistema operacional Linux. Mas diferente de sistemas convencionais, a arquitetura proposta é uma plataforma dedicada ao funcionamento dos serviços de PABX e possui recursos limitados. Logo, uma distribuição alternativa do Linux é necessária.

Utilizou-se o LTIB (*Linux Target Image Builder*) [9] como construtor da distribuição para a plataforma alvo. Através desta ferramenta é possível realizar a configuração do *kernel* (foi utilizada a versão 2.6.24) e a seleção das aplicações que serão executadas na plataforma alvo, cujo fabricante do processador provê suporte às evoluções desta [10].

A arquitetura proposta suporta alguns protocolos de comunicação necessários para prover o serviço de PABX e seu gerenciamento, tais como: TCP, IP, UDP, SIP, RTP, RTCP, TDMoE e SNMP. O suporte a alguns destes protocolos deve ser incluso diretamente no *kernel* do sistema operacional. Além disso, alguns aplicativos são necessários utilizar esses recursos disponibilizados pelo *kernel*, como o próprio Asterisk, OpenSSH, OpenVPN, entre outros.

A arquitetura das placas-filhas tem um DSP como elemento de processamento. Assim como na Placa-mãe, é utilizado nas placas-filhas E1 e FXS o sistema operacional Linux, versão 2.6.28 do *kernel*. Isto torna a integração entre os sistemas mais fácil, por utilizar bibliotecas compatíveis. O construtor de distribuição utilizado para as placas-filhas é o *μCLinux* [11]. No entanto, o Asterisk é executado apenas na Placa-mãe, sendo necessário um conjunto reduzido de protocolos de

comunicação para as placas-filhas, não necessitam dos protocolos SIP, RTP e RTCP, entre outros.

B. Asterisk

O Asterisk, versão 1.6.2.9, foi embarcado na Placa-mãe. Através de conexões SSH ou Serial é possível personalizar os arquivos de configuração e realizar o monitoramento das chamadas através do Asterisk CLI (*Command Line Interface*).

A comunicação entre a Placa-mãe e as placas-filhas é realizada através de conexão *ethernet* devido a sua alta taxa de transmissão. Informações de inicialização, bem como, canais de voz e dados de sinalização para as interfaces presentes nas placas-filhas trafegam por esta interface.

O protocolo TDM é utilizado pelas placas-filhas para realizar a comunicação do DSP com o dispositivo de acesso ao meio físico. Na Placa-filha E1 é realizada a comunicação entre o E1 *Framer* e o DSP, e na Placa-filha FXS entre o DSP e o SLAC.

Com o objetivo de manter a compatibilidade dos dados utiliza-se o protocolo TDMoE. Neste caso, os dados de voz e sinalização que estão em formato TDM são encapsulados em pacotes *ethernet* e transmitidos entre as placas. O protocolo TDMoE é suportado pelo Asterisk através dos *drivers* presentes no DAHDI. Nas placas-filhas, o tráfego de dados entre a interface TDM e a interface *ethernet* é realizada em nível de *kernel* do sistema operacional, sendo necessária apenas uma aplicação para realizar a configuração das interfaces TDM e *ethernet* e inicializar a transmissão.

A Placa-filha E1, em conjunto com o Asterisk, suporta vários tipos de sinalizações [12], como ISDN-PRI, ISDN-BRI e MFC/R2. No Brasil, a MFC/R2 é a mais utilizada [12]. Para suporte a esses protocolos no Asterisk, é necessário o *porting* de duas bibliotecas adicionais, a *libpri* [2] e a *libopenr2* [13].

C. Inicialização do Sistema

Na inicialização do sistema se faz necessário estabelecer um *Span dynamic* entre a Placa-mãe e cada uma das placas-filhas possivelmente conectadas na Placa-mãe. Esta conexão é realizada através da interface *ethernet* disponibilizada no conector *mezzanine*.

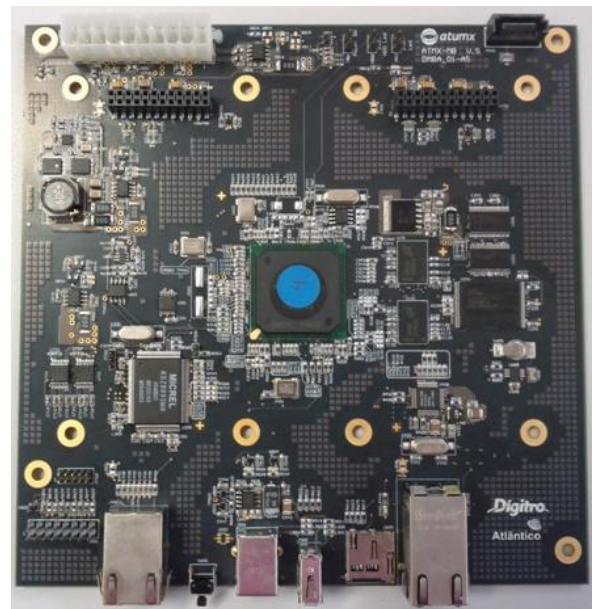
Para realizar tal tarefa automaticamente, foi desenvolvido um processo de inicialização dos *Spans*. Através da comunicação via *sockets*, as placas trocam informações sobre o tipo de placa-filha e o endereço MAC.

Após essa comunicação inicial, os *Spans* são acionados e a comunicação entre as placas através do DAHDI é estabelecida, desta maneira se torna possível para o Asterisk a visualização dos canais disponibilizados pelas placas-filhas.

VI. RESULTADOS

A. Placas de Circuito Impresso

As três placas de circuito impresso: Placa-mãe, Placa-filha E1 e Placa-filha FXS foram manufaturadas, montadas e testadas eletrônica e funcionalmente com sucesso. Cada uma medindo respectivamente: 170x170mm, 65x155mm e 84x155mm. A vista superior de cada uma das placas pode ser observada na Figura 5.



(a)



(b)



(c)

Fig. 5. Vistas superiores das placas desenvolvidas: (a) Placa-mãe. (b) Placa-filha E1. (c) Placa-filha FXS.

B. Formato e Apresentação do Produto Final

Para a elaboração de um produto final, um gabinete mini-ITX foi adquirido para acomodação e alimentação das placas desenvolvidas.

A foto do produto final, composto pela Placa-mãe, placas-filhas e gabinete pode ser visto na Figura 6.

C. Testes VoIP

Com o objetivo de testar as configurações e as funcionalidades do Asterisk foi montado um cenário de testes utilizando um módulo do produto final, dois telefones IP e dois *softphones*. Todos os elementos que compõem os testes devem estar ligados na mesma rede de dados.

O primeiro passo para realização dos testes é a criação dos usuários VoIP nos arquivos de configuração do Asterisk. Em seguida, os telefones e *softphones* devem ter suas contas IP configuradas para o servidor.



Fig. 6. Foto das três placas desenvolvidas montadas no gabinete.

Além de realizar ligações simples entre os telefones, foram testadas com sucesso as seguintes funcionalidades: conferência, URA (Unidade de Resposta Automática), música de espera, siga-me, estacionamento e transferência de chamadas.

D. Testes de Pré-Certificação

Para serem comercializados produtos de telecomunicações, estes devem possuir o Selo Anatel. Para obter a certificação de conformidade do produto, o interessado deve procurar um OCD (Organismo de Certificação Designado), cuja relação encontra-se no site da Anatel (Agência Nacional de Telecomunicações). O OCD escolhido orientará o solicitante em relação aos documentos necessários e procedimentos que devem ser seguidos. Obtida a certificação, o interessado deve requerer à Anatel a homologação do certificado emitido pelo OCD [14].

O OCD designado classificou o equipamento como sendo uma "Central privada de comutação telefônica" categoria I e listou as normas às quais o produto deveria ser homologado, sendo elas [15], [16], [17], [18], [19], [20], [21] e [22].

Dado esses requisitos, o equipamento montado foi submetido aos testes pelo CPQd, que além de OCD, também é um laboratório de ensaio credenciado junto à Anatel.

Não foram encontradas "Não Conformidades" nos testes referentes aos testes às Normativas [15], [16], [17], [18], [19] e [22].

Durante os testes relativos à [20], foram identificados "Não Conformidades" relativas ao reconhecimento do DTMF, cuja solução envolveu uma alteração no oscilador do hardware e alterações de firmware do sistema.

A norma [21] é composta de diversos pontos, entre eles o de "Emissão de Perturbação Radiada" e o de "Imunidade a Descargas Eletrostáticas". No primeiro, uma das frequências ficou acima do limite permitido, tendo sido necessária uma alteração de layout e reteste para aprovação. Durante os testes do segundo ponto, houve perda temporária de desempenho sendo necessária intervenção do operador para que fosse recuperado o funcionamento normal. Este ponto também foi

corrigido com alterações no layout das placas e retestado com sucesso.

Estes testes comprovam que o conjunto de hardware e software associados à Placa-mãe e às interfaces E1 e FXS estão aptos a se integrarem a sistemas telefônicos obedecendo às normas brasileiras.

AGRADECIMENTOS

À equipe de profissionais do Instituto Atlântico que atuou no projeto cujo produto principal originou este trabalho.

À comunidade de software livre pelo desenvolvimento e disponibilização das várias ferramentas utilizadas neste projeto.

REFERÊNCIAS

- [1] Projeto de Redes. Disponível em: <http://www.projetoderedes.com.br/tutoriais/tutorial_rede_telefonica_mutada_01.php>. Acesso em: 12 de abril de 2012.
- [2] Asterisk.org. Disponível em: <<http://www.asterisk.org/>>. Acesso em: 12 de abril de 2012.
- [3] Digium.com. Disponível em: <<http://www.digium.com/en/>>. Acesso em: 12 de abril de 2012.
- [4] F. Iseki, Y. Sato, Moo Wan Kim, "VoIP system based on Asterisk for enterprise network" *13th International Conference on Advanced Communication Technology (ICACT)*, pp.1284-1288, 13-16 Feb. 2011.
- [5] DAHDI. Disponível em: <<http://docs.tzafir.org/il/dahdi-tools/README.html>>. Acesso em: 12 de abril de 2012.
- [6] XRT86VL30 T1/E1/J1 Bits Element – Hardware Manual. Exar, Maio 2008.
- [7] Octal Subscriber Line Audio-processing Circuit Next Generation Carrier Chipset (NGCC). Zarlink Semiconductor, Outubro 2008.
- [8] Subscriber Line Interface Circuit Next Generation Carrier Chipset (NGCC). Zarlink Semiconductor, Dezembro 2009.
- [9] LTIB. Disponível em: <<http://ltib.org/>>. Acesso em: 13 de abril de 2012.
- [10] Freescale. Disponível em: <<http://www.freescale.com/>>. Acesso em: 13 de abril de 2012.
- [11] µCLinux. Disponível em: <<http://www.uclinux.org/>>. Acesso em: 13 de abril de 2012.
- [12] Tutoriais Telefonia Fixa. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialsgin/pagina_1.asp>. Acesso em: 12 de abril de 2012.
- [13] Libopenr2. Disponível em: <<http://www.libopenr2.org/>>. Acesso em: 12 de abril de 2012.
- [14] Certificação de produtos. Disponível em: <<http://www.anatel.gov.br/Portal/exibirPortalRedireciona.do?codigoDocumento=41706>>. Acesso em: 16 de abril de 2012.
- [15] TELEBRAS. SDT 225-100-706 - Especificações gerais de equipamentos multiplex 2048 kbit/s. 1992
- [16] TELEBRAS. SDT 210-110-702 - Especificações de Sinalização entre Registradores para a Rede Nacional de Telefonia Via Terrestre. 1996
- [17] TELEBRAS. SDT 210-110-703 - Especificações de Sinalização de Linha para a Rede Nacional de Telefonia Via Terrestre. 1996
- [18] ETSI. TBR4: Tabela C1 do Anexo C - Camada 2 do protocolo de acesso do usuário RDSI à rede de telecomunicações (DSS1). 1997
- [19] ETSI. TBR4: Tabela D1 do Anexo D - Camada 3 do protocolo de acesso do usuário RDSI à rede de telecomunicações (DSS1). 1997
- [20] ANATEL. Anexo à Resolução nº 512 - Norma para Certificação e Homologação da Interface Analógica de Adaptadores para Terminal de Assinante. 2008
- [21] ANATEL. Anexo à Resolução nº 442 - Regulamento para Certificação de Equipamentos de Telecomunicações Quanto aos Aspectos de Compatibilidade Eletromagnética. 2006
- [22] ANATEL. Anexo à Resolução nº 529 - Regulamento para Certificação de Equipamentos de Telecomunicações Quanto aos Aspectos de Segurança Elétrica. 2009