UNIVERSIDADE DO VALE DO ITAJAÍ

MARCO ANTÔNIO VARGAS

ANALISADOR DE PROTOCOLO PARA SISTEMAS DE PABX: ESTUDO DE CASO DO SISTEMA IMPACTA INTELBRAS

São José

2006

MARCO ANTÔNIO VARGAS

ANALISADOR DE PROTOCOLO PARA SISTEMAS DE PABX: ESTUDO DE CASO DO SISTEMA IMPACTA INTELBRAS

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de Bacharel em Ciência da Computação na Universidade do Vale do Itajaí, Centro de Educação São José.

Orientador: Prof. Dr. Rivalino Matias Júnior

São José

2006

MARCO ANTÔNIO VARGAS

ANALISADOR DE PROTOCOLO PARA SISTEMAS DE PABX: ESTUDO DE CASO DO SISTEMA IMPACTA INTELBRAS

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de Bacharel em Ciência da Computação e aprovado pelo Curso de Ciência da Computação, da Universidade do Vale do Itajaí (SC), Centro de Educação São José.

São José, 14 de dezembro de 2006.

Apresentada à Banca Examinadora formada pelos professores:

Prof. Dr.Rivalino Matias Junior

UNIVALI – Campus São José

Orientador

Prof. Dra. Carla Merkle Westphall, membro da banca examinadora.

Prof. Dra. Michelle Silva Wangham, membro da banca examinadora.

RESUMO

A forte competição do mercado tem obrigado as empresas a dispor de ferramentas que auxiliem o desenvolvimento de seus produtos/serviços, tornando-os cada vez mais rápido sem, no entanto, comprometer a sua qualidade. Dentro deste contexto, surgiu a necessidade de criar uma ferramenta que auxiliasse a análise dos problemas relacionados à troca de mensagens entre **PABX** e **Terminal Inteligente,** ambos produtos da empresa INTELBRAS**.** O objetivo do sistema proposto, portanto, é facilitar a análise dos problemas que o produto INTELBRAS venha a apresentar, referente à comunicação de dados, quer seja na fase de desenvolvimento e implementação, ou após sua comercialização. A proposta trata de um *software* analisador de protocolo (AP) de comunicação entre equipamentos da Plataforma IMPACTA INTELBRAS. Este protocolo é proprietário, tendo sido desenvolvido com base no padrão ISDN Q.921 do ITU-T. De uma forma geral, o sistema captura o que trafega no caminho de comunicação entre PABX e Terminal, registrando em sua base de dados as PDUs (*Protocol Data Unit*) capturadas. Na seqüência, os campos das PDUs capturadas são apresentados na tela de maneira estruturada, permitindo a análise dos dados de forma simples e clara. O sistema ainda inclui a possibilidade do uso de filtros durante a consulta e também a geração de relatórios sobre a base de dados histórica. A captura dos dados utiliza um *hardware* proprietário que atua direto no canal (2B+D) de comunicação entre o PABX e o terminal, disponibilizando-os para o computador por meio de uma porta serial. O *hardware* utiliza um chip responsável por toda comunicação da camada física e de enlace.

Palavras Chaves: Analisador de Protocolo, PABX, Comunicação de Dados.

ABSTRACT

Strong market competition has forced companies toward making use of tools that aid products development turning it faster but without compromising their quality. Concerning to this context raised the idea of creating a tool that helped the analysis of problems related to messages change between Intelligent Terminals and PBX, both INTELBRAS products. Therefore the objective of this proposed system is to easy the analysis of problems that eventually an INTELBRAS products come out, concerning to data communication, be it in development state or after its market instance. This work treats specifically about a protocol communication analyzer between equipments of INTELBRAS IMPACTA platform. This protocol is a proprietary one but was based on an ITU-T ISDN Q.921 standard. In a broad way, the system capture the data crossing the communication way between terminal and PBX, storing the seized PDUs *(Protocol Data Unit)* in a data base. Next, PDUs fields are structurally displayed allowing its clear and easy analysis. The system also includes a possibility of applying filters during data handling and generation of a printed report over history data. The data captures rely on an exclusive *hardware* that acts directly on the data channel (2B+D) between the PBX and terminal. This *hardware* uses a single chip that is responsible for the whole communication of physical and link layers.

Key words: Protocol analyzer, PBX, Data Communication.

Lista de figuras

LISTA DE ABREVIATURAS E SIGLAS

ACK - *Affirmative Acknowledgement*

ADDCP - *Advanced Data Communication Control Procedure*

AGI - *Asterisk Gateway Interface*

AM - *Amplitude Modulation*

AMI - *Alternate Mark Inversion*

ANSI - *American National Standards Institute*

AP – Analisador de Protocolos

API – Application Program Interface

ASC - *American Standard Code*

ASCII - *American Standard Code for Information Interchange*

AVVID - Arquitetura de Voz, Vídeo e Dados Integrados

C / R bit – Command / Response bit

CCITT - *Consultative Commitee for International Telegraph and Telephone*

CPU – *Central Processing Unit*

CRC - *Cyclic Redundancy Checking*

DAC - Distribuidor Automático de Chamadas

DSP - *Digital Signal Processing*

DTMF - *Dual Tone Multi-Frequency*

EA bit – Extension Address bit

EBCDIC - *Extended Binary Coded Decimal Interchange Code*

EIA - *Electronic Industries Association*

ENK – *Enquiry*

ETSI - *European Telecommunication Standard Institute*

FCS - *Frame check sequence*

FIFO - *First In First Out*

FM - *Frequency Modulation*

FTP - *File Transfer Protocol*

GLP - *General Public License*

HDLC - *High-level Data Link Control*

HTTP - *Hyper Text Transfer Protocol*

IAX - *Inter-Asterisk eXchange*

IMP - *Interface Message Processor*

IOM - *ISDN Oriented Modular*

IP - *Internet Protocol*

ISO - *International Organization for Standardization*

ISDN - *Integrated Services Digital Network*

ISP - In-System Programming

ITU-T - *International Telecommunications Union*

JTAG **-** *Joint Test Action Group*

KS - *Key System*

LAN - *Local Area Network*

LAP - *Link Acess Procedure*

LAPB - *Link Acess Procedure Basic*

LAPD - *Link Access Procedures on the D-Channel*

LED – *Light Emitter Diode*

LRC **-** *Longitudinal Redundancy Checking*

LSB - *Least Significative Bit*

MGCP - *Media Gateway Control Protocol*

NACK - *No Affirmative Acknowledgement*

NAT - *Network Address Translation*

P&D - Pesquisa e Desenvolvimento

PABX - *Private Automatic Branch eXchange*

PBX - *Private Branch eXchange*

PADI – Placa de Aquisição Dados INTELBRAS

PC – *Personal Computer*

PCM - *Pulse Code Modulation*

PCPTI - Protocolo de Comunicação PABX TI INTELBRAS

PDU - *Protocol Data Unit*

PSTN - *Public Service Telephony Network*

RDSI - Rede Digital de Serviços Integrados

REJ – *Reject*

RM-OSI - *Reference Model for Open Systems Interconnection*

RR - *Receiver Ready*

RS - *Recommended Standard*

SAP **-** *Service Access Point*

SAPI - *Service Access Point Identifier*

SBUF - *Serial Buffer*

SCI - *Serial Communication Interface*

SDLC - *Synchronous Data Link Control*

SIE - Serial Interface Engine

SIP - *Session Initiated Protocol*

SMTP - *Simple Mail Transfer Protocol*

SRAM - *Sequential Random Access Memory*

SRST - *Survivable Remote Site Telephony*

STFC - Sistema de Telefonia Fixa Comutada

TCP - *Transmission Control Protocol*

TEI - *Terminal Equipament Identifier*

TI - Terminais Inteligentes

UART - *Universal Asynchronous Receiver-Transmitter*

UCP - Unidade Central de Processamento

UDP - *User Datagram Protocol*

UML - *Unified Modeling Language*

URA - Unidade de Resposta Audível

USB - *Universal Serial Bus*

VLAN - *Virtual Local Area Network*

VoiP – *Voice over Internet Protocol*

VRC - *Vertical Redundancy Checking*

WAN - *Wide Area Network*

SUMÁRIO

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

A expansão e popularização da telefonia fixa se deve, em grande parte, à queda de preços nas linhas telefônicas nos últimos anos, e também ao crescimento de novas tecnologias que usam o meio físico para comunicação de dados ou de voz (SPITZ *et al*., 2000). Neste novo cenário, surgiu nas residências e empresas a necessidade de equipamentos que interligassem várias linhas telefônicas e facilitasse o uso destas novas tecnologias. Com isso, foram desenvolvidos os PABXs (*Private Automatic Branch eXchange*), equipamentos destinados para este fim (MARTINS, 2002).

Com o surgimento dos PABXs, houve a necessidade do desenvolvimento de Terminais Telefônicos com mais funcionalidades do que um simples telefone, os chamados KS (*Key System*) ou TI (Terminais Inteligentes). Esses novos TIs, têm como principal finalidade facilitar o uso das principais funcionalidades do PABX, auxiliando na visualização dos estados das linhas e ramais, além de tornar as programações do PABX mais "amigáveis".

Todos os dias surgem novidades para diminuir os custos e o tamanho dos equipamentos. Essa corrida tecnológica faz com que as empresas tenham que se adequar ao mercado, desenvolvendo cada vez mais rápido, novos produtos para não perderem sua fatia de mercado. As empresas que não conseguem acompanhar essa corrida tecnológica e as necessidades do mercado estão fadadas ao fracasso e a falência. Para se tornarem mais ágeis no desenvolvimento de novos produtos, os setores de desenvolvimentos das empresas de tecnologia precisam cada vez mais de equipamentos e softwares que auxiliem e facilitem o trabalho dos técnicos e engenheiros na realização de suas tarefas.

Esse trabalho tem por objetivo criar um *software* que auxilie o setor de P&D (Pesquisa e Desenvolvimento), divisão de terminais da INTELBRAS, a identificar possíveis problemas nos Terminais Inteligentes TI 4245 e 2165 (INTELBRAS, 2006), devido a problemas de comunicação entre seus PABXs Digitais e os Terminais Telefônicos Dedicados da nova Plataforma IMPACTA (INTELBRAS, 2006) ou a novos equipamentos que utilizem o protocolo PCPTI (Protocolo de Comunicação PABX TI INTELBRAS) (INTELBRAS, 2006).

Isso se faz necessário, pois além de problemas triviais também podem ocorrer problemas desconhecidos, que só serão detectados depois da conclusão do projeto, quando o equipamento já se encontra no cliente. Como resultado, tem-se o comprometimento da boa imagem do produto perante a sociedade e podendo perder com isso a confiabilidade nos equipamentos INTELBRAS.

A identificação mais eficiente de falhas de comunicação torna mais fácil, ágil e confiável a conclusão do projeto, pois os profissionais envolvidos podem traçar uma ação mais eficaz para solucionar os problemas detectados.

1.2 PROBLEMA

Com a divisão de tarefas entre as equipes de desenvolvimento, uma parte da equipe fica responsável por desenvolver o TI e os demais alocados no desenvolvimento do PABX. Depois de especificado o protocolo e a forma de comunicação, segue a implementação, e normalmente surgem várias incompatibilidades e erros na fase de integração dos sistemas.

Devido à dificuldade em detectar problemas e as suas localizações, a fim de acionar as equipes responsáveis, surgiu a necessidade de desenvolver um sistema que apresentasse as mensagens que trafegam entre os dois equipamentos de modo a auxiliar o rastreamento dos erros de comunicação.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Desenvolver um *software* para a coleta e análise de dados do protocolo de comunicação PCPTI entre o Sistema de PABX IMPACTA e seus TIs de tecnologia INTELBRAS.

1.3.2 Objetivos Específicos

- Prover uma ferramenta para facilitar a análise de problemas de especificação e implementação de *software*, durante o desenvolvimento e/ou manutenção do protocolo PCPTI;

- Desenvolver facilidades na ferramenta que auxiliem o técnico a identificar problemas relacionados à comunicação de dados envolvendo o protocolo PCPTI, durante a fase de implementação dos sistemas PABX INTELBRAS;
- Desenvolver facilidades na ferramenta que suporte o técnico de campo no diagnóstico da comunicação PABX e Terminal de forma remota.

1.4 ESCOPO E DELIMITAÇÕES

O PABX e TI são conectados entre si por meio de um par de fios de cobre, por onde trafegam os sinais de voz e dados de controle (protocolo PCPTI). Segundo especificações do equipamento, o terminal pode ser conectado a uma distância de até 200 metros do PABX (INTELBRAS, 2006).

Existe um *hardware*, desenvolvido pela equipe de terminais da INTELBRAS, que é ligado no terminal. Essa placa é responsável por capturar os dados que trafegam na linha com o objetivo de disponibilizar esses dados para alguma ferramenta que possa fazer a análise desses dados. O *hardware* possui o *chip* SCOUT-DX do fabricante *Infineon* e o microcontrolador uPSD3234 do fabricante *ST-Microeletronics*. *O SCOUT-DX* é responsável por toda comunicação da camada física e de enlace. Este *chip* já implementa toda a parte da comunicação, sincronismo, início de quadro e retransmissão de dados perdidos. O *chip* uPSD3234 possui uma saída serial padrão RS-232 e uma saída USB (*Universal Serial Bus*) *(INFINEON TECHNOLOGIES, 2002)*. Os dados capturados são tratados e disponibilizados na porta RS-232. O sistema proposto neste trabalho se conecta na serial, captura os dados e os armazena em uma área de dados temporária para posterior processamento. Não faz parte deste trabalho a captura dos dados do meio físico. Então, assume-se que os dados já estejam formatados e entregues nas portas de comunicação para o seu tratamento e decodificação segundo as regras do protocolo PCPTI.

No trabalho, também são decodificadas setenta e quatro PDUs (*Protocol Data Unit*), divididas em seis grupos:

SAPI = 0 Processos de controle e estabelecimento da chamada;

SAPI = 17 Processos de *status*;

SAPI = 18 Processos de inicialização;

SAPI = 19 Processos de programação.

SAPI = 20 Processos de transmissão de dados

SAPI = 63 Funções de manutenção da camada 2

As PDUs com datas e horários capturadas pela ferramenta são armazenadas em arquivos tipo ASCII (*American Standard Code for Information Interchange*) no disco rígido da máquina que esteja operando no modo servidor, onde a ferramenta esteja instalada. Esses dados recuperados podem ser consultados remotamente por meio da interface cliente do sistema proposto.

O sistema permite que o usuário faça um filtro de PDU para captura e visualização destas PDUs. O sistema também disponibiliza um editor que permite a geração e a edição de relatórios. Esses relatórios contêm data, hora e as mensagens trocadas entre PABX e TI.

O AP (Analisador de Protocolo) proposto funciona exclusivamente nos equipamentos PABX Digital da nova plataforma IMPACTA e Terminais Inteligentes TI 4245 e 2165 da INTELBRAS, por se tratar de um protocolo proprietário desenvolvido para a interligação desses equipamentos. Novos equipamentos que venham a utilizar o protocolo PCPTI também são suportados.

1.5 RESULTADOS ESPERADOS

Espera-se ter no final do projeto um *software* de análise do protocolo PCPTI, que atenda a todos os requisitos descritos na seção anterior. A facilidade do uso por parte da equipe técnica é um dos principais requisitos deste AP, tendo em vista que seus objetivos são facilitar e auxiliar as equipes de desenvolvimento e de operação no que tange às suas atividade de diagnóstico de problema envolvendo o protocolo PCPTI. É importante ressaltar que o *software* é uma ferramenta de auxílio para o diagnóstico de problemas e não possui funções que o habilite a identificar problemas de forma autônoma. A detecção e a resolução de

problemas dependerá da competência e experiência do técnico ou engenheiro que estiver utilizando a ferramenta.

1.6 JUSTIFICATIVA

Devido a grande necessidade de agilizar o desenvolvimento de novos produtos, no contexto da área de desenvolvimento na INTELBRAS, percebeu-se que a principal causa de atraso nos projetos é o tempo que se leva para descobrir erros e falhas, quando se trata de comunicações entre equipamentos. Para ajudar a reduzir o tempo gasto na procura destes erros e falhas, surgiu na INTELBRAS a necessidade de desenvolver uma ferramenta para apoiar a atividade de diagnóstico de falhas e erros de comunicação entre os equipamentos.

O protocolo de comunicação usado é proprietário, ou seja, desenvolvido exclusivamente para a comunicação desses equipamentos, tendo como base o protocolo conhecido LAPD (*Link Access Procedures on the D-Channel).* Num levantamento efetuado neste trabalho, não se encontrou nenhuma ferramenta comercial ou de distribuição gratuita que consiga decodificar as PDUs trocadas por estes respectivos equipamentos.

Também foi constatado que a contratação de uma empresa de desenvolvimento de *software* tornaria o projeto inviável, devido ao longo tempo de desenvolvimento (cerca de 18 meses), prazo este estimado com base no desenvolvimento de um sistema com características semelhantes e que foi desenvolvido pela mesma empresa. Além disso, o alto custo financeiro e a necessidade de tempo que a equipe de terminais deveria despender para auxiliar a contratada nesse desenvolvimento, justificaram a iniciativa de desenvolvimento interno desse projeto. De acordo com a gerência de desenvolvimento de PABX, esta ferramenta é útil não só para o processo de desenvolvimento, mas também nos testes de integridade do sistema, na validação do projeto e durante toda a vida útil comercial do equipamento. Atualmente, ocorrem problemas que somente são detectados após meses de comercialização, e em clientes específicos que usam certas facilidades dos equipamentos, cujas situações não foram previstas. Nestes casos, a ferramenta tem por objetivo proporcionar uma interface para acompanhamento remoto do que ocorre com os equipamentos. Esta é uma aplicação para a fase de operação e, portanto manutenção dos sistemas. Com tal funcionalidade a equipe não precisa deslocar-se até o local para analisar o problema, basta um técnico autorizado instalar a

ferramenta no local do equipamento do cliente. Vale lembrar que os TIs e PABXs são fabricados e comercializados em larga escala, e além de serem vendidos para o mercado nacional, também são exportados para toda a América Latina e alguns países da Europa e África, o que torna essa funcionalidade de acesso remoto muito importante.

Com todos esses motivos, o autor viu uma oportunidade de aprimorar seus conhecimentos em desenvolvimentos de *software* e de se aprofundar em vários outros assuntos e técnicas que foram necessários para o desenvolvimento e conclusão do projeto proposto.

1.7 ASPECTOS METODOLÓGICOS

Uma pesquisa é a busca por informações ou respostas sobre algo ou algum assunto que desejamos nos aprofundar mais (SILVA, 2001, p.21). Na maioria das vezes, essas informações são necessárias para solucionar algum problema ou entender algum processo ou sistema. Segundo Santos (2000), uma pesquisa é formada por várias etapas, mas de uma forma geral, se caracteriza sobre três critérios básicos: os objetivos, os procedimentos de coleta e as fontes utilizadas na coleta. Uma das importantes etapas, juntamente com a definição dos objetivos, é a de mostrar o caminho que será utilizado para se chegar a esses objetivos, ou seja, a metodologia aplicada na pesquisa.

1.7.1 Caracterização da pesquisa segundo o objetivo

Em Silva (2001), a pesquisa definida segundo o objetivo pode ser classificada em três tipos distintos: Pesquisa Exploratória, Pesquisa Descritiva e Pesquisa Explicativa.

Esse trabalho tem como principal característica, a de se saber mais sobre o problema, com a finalidade de desenvolver um sistema que auxilie na detecção e, conseqüentemente, contribuir para evitar um defeito no futuro, tendo como base experiências práticas sobre o assunto abordado. Devido às características apresentadas, essa pesquisa pode ser classificada como uma pesquisa Exploratória.

“A Pesquisa Exploratória visa proporcionar maior familiaridade com o problema com vistas a torná-lo explícito ou a construir hipóteses. Envolve levantamento bibliográfico; entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado; análise de

exemplos que estimulem a compreensão. Assume, em geral, as formas de Pesquisas Bibliográficas e Estudos de Caso" (SILVA, 2001, p.21).

1.7.2 Caracterização da pesquisa segundo os procedimentos de coleta

De acordo com Silva (2001), os procedimentos de coletas são métodos práticos que são utilizados para a coleta de dados ou informações, que serão necessárias para o entendimento de um determinado problema, fenômeno ou acontecimento. As formas mais comuns de coleta de informações são: Pesquisa Bibliográfica, Pesquisa Documental, Pesquisa Experimental, Levantamento, Estudo de Caso e Pesquisa *Expost-Facto*.

Este trabalho trata-se do estudo de um protocolo de comunicação proprietário usando como base materiais não publicados, materiais de domínio público com informações e técnicas de implementação de protocolos, manuais de componentes e equipamentos. Desta forma, o trabalho tem características tanto de Pesquisa Bibliográfica quanto de Pesquisa documental.

"Pesquisa Bibliográfica é quando a pesquisa é elaborada a partir de material já publicado, constituído principalmente de livros, artigos de periódicos e atualmente com material disponibilizado na Internet" (SILVA, 2001, p.21).

"Pesquisa documental é quando a pesquisa é elaborada a partir de materiais que não receberam tratamento analítico" (SILVA, 2001, p.21).

1.7.3 Caracterização da pesquisa segundo as fontes de informação

As fontes de informações são lugares de onde se extraem os dados que se precisa. Estas podem ser: o campo, o laboratório ou a bibliografia. Campo é o lugar natural onde acontecem os fatos e fenômenos. Normalmente, se faz por observação direta, levantamento ou estudo de caso (SANTOS, 2000).

Considerando o contexto do trabalho, quanto à fonte de informação este pode ser caracterizado como sendo uma pesquisa bibliográfica e também de laboratório.

2 COMUNICAÇÃO DE DADOS

2.1 INTRODUÇÃO

Atualmente, exige-se que a informação esteja disponível no menor espaço de tempo possível. As empresas em geral necessitam que as diversas fontes de informações sejam convergentes. Portanto, voz, dados e imagens devem conviver de forma harmoniosa para que o máximo de benefícios seja extraído. Isso só é obtido com uma comunicação eficaz. O segmento de telecomunicações vem se integrando cada vez mais às redes de computadores e são as empresas que mais investem neste segmento (ALVES, 1994).

De acordo com Alves (1994), em um sistema de comunicação de dados as informações são processadas, recebidas ou enviadas de uma localidade para outra. Existem vários serviços de comunicação disponíveis e a escolha depende da necessidade de cada instalação. Além dos meios de comunicação, vários outros componentes fazem parte de um sistema de comunicação de dados, mas os dois principais são *hardware* e *software*.

2.2 O MODELO OSI

A seguir será apresentado o modelo OSI e um resumo dos seus principais conceitos.

2.2.1 Conceitos de protocolo no Modelo OSI

O Modelo OSI é apresentado na Figura 1, sendo o primeiro passo da padronização internacional dos protocolos de comunicação, usada nas diversas camadas que envolvem um sistema de comunicação de dados. Este é chamado de referência para a interconexão de sistemas abertos ou simplesmente **RM-OSI** (*Reference Model for Open Systems Interconnection*). Sistemas abertos são sistemas que podem se interconectar com qualquer outro sistema por meio de um conjunto de padrões abertos, tais como o modelo OSI e o TCP/IP (*Transmission Control Protocol/Internet Protocol*) (SPECIALSKI, 2000).

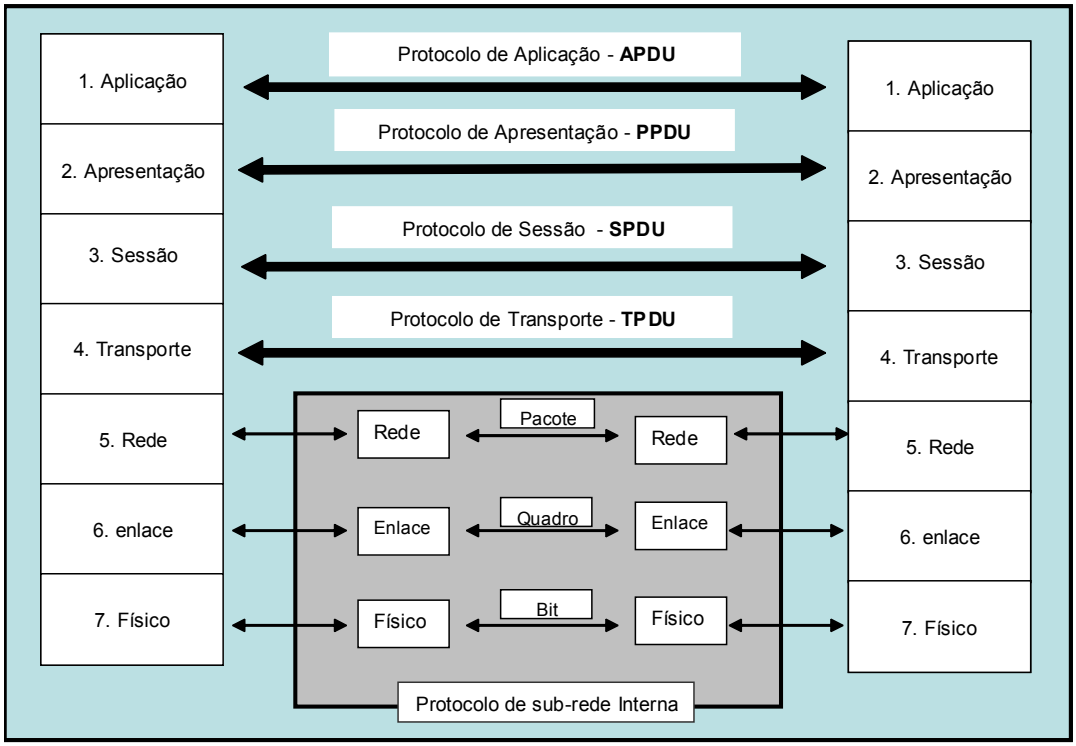

Figura 1 : Arquitetura de sete camadas do modelo OSI.

Fonte: Tanenbaum (2003, p. 41).

A arquitetura OSI é dividida em sete camadas, cujos princípios de definição foram os seguintes (TANENBAUM, 2003):

- cada camada corresponde a um nível de abstração necessário no modelo;
- cada camada deve executar funções próprias e bem definidas;
- as funções de cada camada devem ser escolhidas segundo a definição dos protocolos normalizados internacionalmente;
- os limites entre as camada devem ser definida para a minimizar o fluxo de informação nas interfaces;
- o número de camadas deve ser grande o bastante para que funções distintas não precisem ser colocadas na mesma camada, e ser pequeno o suficiente para que a arquitetura não se torne difícil de controlar.

Como é apresentado na Figura 1, o modelo OSI prevê a comunicação entre sub-redes por meio dos IMPs (*Interface Message Processor*). Neste modelo, os dados são transmitidos da seguinte forma: Um processo emissor deseja enviar uma certa quantidade de dados ao processo receptor, então o emissor envia os dados à camada de aplicação que introduz um cabeçalho de aplicação, e envia a mensagem resultante à camada de Apresentação, a camada de Apresentação por sua vez, introduz na mensagem recebida, um cabeçalho de Apresentação, enviando a mensagem em seguida à camada inferior, no caso a camada de Sessão. A camada de Sessão não toma conhecimento da existência do cabeçalho da camada de Aplicação, considerando este como sendo parte dos dados da mensagem. Este processo de transferência de camada a camada se repete até a camada física, quando os dados serão transmitidos ao sistema destino. No sistema destino os cabeçalhos que foram introduzidos nas camadas do sistema emissor, são interpretados e eliminados nas camadas correspondentes, até que os dados cheguem ao processo receptor (SPECIALSKI, 2000).

O conceito fundamental da transferência de dados é que cada camada foi projetada como se ela fosse realmente horizontal, quando na verdade a transmissão se dá de modo vertical (TANENBAUM, 2003).

2.2.2 Camadas do modelo OSI

A seguir são descritas as principais funções realizadas por cada uma das sete camadas definidas no modelo OSI.

A **camada física** é responsável pela transferência de *bits* num circuito de comunicação. Sua principal função é garantir que cada bit enviado de um lado seja recebido do outro lado sem ter alterado seu valor, ou seja, se o valor do bit enviado for “1” este será recebido como valor “1” e não “0” (SPECIALSKI, 2000).

A c**amada de enlace de dados** tem como função principal a transformação do meio de comunicação “bruto” em uma linha livre de erros de transmissão para a camada de rede. Para isso, o transmissor divide os dados de entrada em quadro de dados e os transmite seqüencialmente. A cada recepção correta do quadro, o receptor envia de volta um quadro de confirmação. Outra função da camada de enlace é, impedir que um transmissor rápido, envie uma quantidade de dados excessiva a um receptor lento, ou seja, é necessário implementar um

controle de fluxo de dados (TANENBAUM, 2003). O item 2.2.3 descreve com mais detalhe esta camada, já que o trabalho se desenvolverá em grande parte neste nível da camada OSI.

A **camada de rede** é responsável pela administração de sub-redes; é ela que define a forma como os pacotes de dados serão encaminhados do emissor ao receptor. As rotas a serem utilizadas podem ser definidas em função de tabelas estáticas ou dinamicamente no momento de cada diálogo em função das condições de tráfego do meio. Esta camada deve ainda, gerenciar problemas de congestionamento, provocados por uma quantidade excessiva de pacotes de dados na rede e devem permitir à interconexão de redes heterogenias (SPECIALSKI, 2000).

A principal função da c**amada de transporte** é recebe os dados da camada acima dela, dividi-los em partes menores, repassar essas unidades à camada de rede e garantir que todas as partes da mensagem chegarão corretamente à outra extremidade. A camada de transporte também determina que tipo de serviço deve ser fornecido a camada de sessão. O tipo de conexão de transporte mais utilizado é, um canal ponto a ponto livre de erros que entrega mensagens na ordem em que eles foram enviados (TANENBAUM, 2003).

A c**amada de sessão** permite que usuários de diferentes máquinas estabeleçam sessões entre eles. Essa camada oferece vários serviços como: o controle de diálogo, o gerenciamento de *token* e a sincronização que permite que transmissões longas continuem a partir do ponto em que estavam ao ocorrer uma falha, isto é possível, fazendo uma verificação periódica durante a transmissão (TANENBAUM, 2003).

A c**amada de apresentação** assume as funções associadas à sintaxe e a semântica dos dados transmitidos. Uma função dessa camada, é a codificação da informação num padrão bem definido como o ASCII, EBCDIC etc. Pode ainda executar outras funções associadas à compressão de dados, utilizando-se do conhecimento do significado da informação, para reduzir a quantidade de informação transmitida, inclusive para implementar funções de criptografia (SPECIALSKI, 2000).

A c**amada de aplicação** contém uma série de protocolos necessários aos usuários. Um protocolo muito utilizado é o HTTP(*Hyper Text Transfer Protocol*). Quando um navegador deseja uma pagina *Web*, este envia o nome da página ao servidor usando HTTP. Outros

protocolos de aplicação são usados para transferência de arquivos e correio eletrônico (TANENBAUM, 2003).

O item a seguir descreve com mais detalhe a camada de enlace, pois o trabalho se desenvolverá em grande parte neste nível da camada OSI.

2.2.3 Camada de Enlace

A principal tarefa da camada de enlace é, converter os dados, sem formatação nenhuma, fornecido pela camada física, em um fluxo de quadros a ser utilizado pela camada de rede. Diversos métodos de enquadramento são usados, como: a contagem de caracteres e a inserção de bytes ou bits. Os protocolos de enlace de dados podem, oferecer recursos de controle de erros para a retransmissão de quadros perdidos ou que contenham erros. Para evitar que um transmissor rápido sobrecarregue um receptor lento, o protocolo dessa camada também pode fornecer um controle de fluxo. O mecanismo de janela deslizante é muito utilizado para integrar o controle de erro e o controle de fluxo de maneira eficiente. Este mecanismo pode ser dividido em categorias de acordo com o tamanho da janela do transmissor e pelo tamanho da janela do receptor. Quando as duas janelas são iguais a 1, é usado o protocolo *stop-and-wait* e quando a janela do transmissor é maior que 1, o receptor pode ser programado para descartar todos os quadros que não o próximo quadro na seqüência ou para armazenar, no buffer, os quadros fora de ordem (TANENBAUM, 2003).

Na camada de enlace existe uma série de protocolos usados para casos específicos. Existem protocolos usados em ambientes livres de erros, no qual o receptor pode manipular qualquer fluxo de dados enviado a este. Existe protocolo para o controle de erros, introduzindo numero de seqüência e utilizando o algoritmo *stop-and-wait*. Existe ainda, protocolo que permite a comunicação bidirecional que, usam o conceito de janela deslizante e protocolos que utiliza a retransmissão seletiva e confirmações negativas. Estes protocolos podem ser modelados usando diversas técnicas, como: o modelo de máquinas de estados finitos e os modelos de redes de Petri.(TANENBAUM, 2003).

Tanenbaum (2003, p. 249) afirma que muitas redes utilizam, na camada de enlace de dados, um dos protocolos orientados a bits, como: o **SDLC** (*Synchronous Data Link Control*), **ADCCP** (*Advanced Data Communication Control Procedure*), **HDLC** (*High-level Data Link*

Control) ou o LAPB (*Link Access Procedure Basic*). Todos os protocolos de enlace são derivados do protocolo **SDLC** da IBM. Depois de desenvolver o SDLC, a IBM o submeteu ao ANSI (*American National Standards Institute*) e a ISO (*International Organization for Standardization*) para a sua aceitação como padrão nos Estados Unidos e no mundo. O ANSI modificou e deu o nome de **ADDCP**, e a ISO alterou transformando-o no **HDLC**. Depois, o CCITT (*Consultative Commitee for International Telegraph and Telephone*) adotou e modificou o HDLC e o transformou em **LAP** (*Link Access Procedure*). Posteriormente o CCITT modificou o padrão novamente e passou a chamá-lo de LAPB, com o intuito de torná-lo mais compatível com uma versão posterior do HDLC.

Segundo Tanenbaum (2003, p. 249), todos estes protocolos são baseados nos mesmos princípios, todos são orientados a bits e utilizam a técnica de inserção de *bits* para transferência de dados. Estes utilizam uma estrutura de quadro similar a apresentada na Figura 2. A seguir é apresentado o que significa cada campo dessa estrutura.

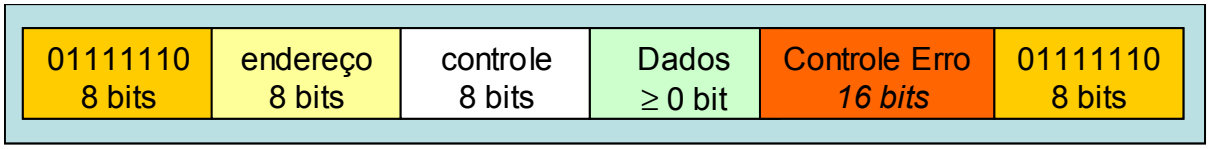

Figura 2 : Formato de quadro básico para protocolos orientados a bits.

Fonte: Tanenbaum (2003, p. 250).

O campo **endereço** é importante principalmente nas linhas com vários terminais, onde é utilizado para identificar um dos terminais. No caso das linhas ponto-a-ponto pode ser usado para distinguir comandos de respostas. O campo **controle** é usado geralmente para números de seqüência de quadros e confirmações. O campo **dados** pode conter qualquer informação e pode ser, algumas vezes longo, embora a eficiência do *checksum* diminuía com o aumento do comprimento do quadro, devido à maior probabilidade de ocorrer vários erros em rajada. O campo **controle de erro** é usado por técnicas de verificação de erros (TANENBAUM, 2003).

Com este descritivo dos campos do quadro básico para protocolos orientados a bits, encerra-se o conceito de camada OSI e na seção a seguir é apresentado os principais elementos de um sistema de comunicação.

2.3 ELEMENTOS DE UM SISTEMA DE COMUNICAÇÃO

Segundo Alves (1994), um sistema de comunicação de dados envolve vários componentes de *hardware*, e devido a necessidade de se transmitir dados através dos meios de comunicação que empregam tecnologias heterogêneas, também são usados componentes de *hardware* da área de telecomunicações. Os principais componentes são: o processador, o modem, a porta de comunicação e a linha de transmissão.

2.3.1 Processador

Almeida (2006) afirma que o **processador** é a parte mais importante para o funcionamento de equipamentos eletrônicos. São circuitos digitais que realizam diversas operações como: cópia de dados, acesso a memórias e operações lógicas e matemáticas. Os mais comuns trabalham apenas com lógica digital binária. Estes possuem geralmente uma pequena memória interna, portas de entrada e de saída e são geralmente ligados a outros circuitos digitais como memórias, multiplexadores e circuitos lógicos. Muitas vezes, também possui uma porta de entrada de instruções, que determinam a tarefa a ser realizada por este. Estas seqüências de instruções, geralmente, estão armazenadas em memórias, e formam o programa a ser executado pelo processador. Em geral, o que diferencia um processador dos outros é sua capacidade de realizar tarefa em um determinado tempo. Os processadores podem ser mais simples e específicos como é o caso dos processadores que controlam eletrodomésticos e dispositivos simples como portões eletrônicos e algumas partes de automóveis, ou ainda, podem ser extremamente complexos e mais genéricos, como nos processadores de computadores pessoais (ALMEIDA, 2006).

Neste trabalho, será usado o microprocessador **uPSD3234** que possui além do processador, memórias de dados onde são armazenados e executados os programas fontes. Esse tipo de processador é dedicado para o desenvolvimento de sistemas embarcados *(ST Microelectronics,* 2004*)*.

2.3.2 Modem

O modem, de uma forma geral, tem a finalidade de converter dados binários que chegam até este em sinais analógicos que possam ser transmitidos pela rede telefônica. Mas também,

convertem sinais analógicos em sinais digitais, ou seja, modulam sinais digitais que são sinais de onda quadrada, em sinais analógicos que são sinais elétricos oscilantes e demodulam sinais analógicos em sinais digitais. O nome modem tem sua origem nas funções MOdular – DEModular (ALVES, 1994).

A natureza do processo de conversão de sinal depende da origem e destino do sinal recebido pelo modem. Em geral, este recebe os sinais binários de um terminal ou computador e converte-os em sinais de freqüência de voz, então transmite esses sinais através do sistema telefônico. Na ponta receptora, um outro modem compatível converte esses sons em sinais binários e envia esses códigos para o terminal ou computador (JORDAN; CHURCHILL, 1994).

No trabalho, é usado o **SCOUT – DX PSB 21373** que é um componente que integra em um mesmo chip o modem e funções específicas usadas em equipamentos de telefonia.

2.3.3 Porta de Comunicação RS 232

A RS (*Recommended Standard)* se refere a uma padronização de uma interface comum para comunicação de dados entre equipamentos, criada no início dos anos 60, por um comitê conhecido atualmente como EIA (*Electronic Industries Association)*. O padrão **RS 232** especifica as tensões, temporizações e funções dos sinais, um protocolo para troca de informações, e as conexões mecânicas. As modificações mais recentes feitas pela EIA foram introduzidas em 1991, mudou o nome de RS 232 para **EIA 232**, algumas linhas de sinais foram renomeadas e várias linhas novas foram definidas.

As maiores dificuldades encontradas pelos usuários na utilização da interface RS 232 incluem pelo menos dois importantes fatores que são a ausência ou conexão errada de sinais de controle, resultando em estouro do buffer ("*overflow*") ou travamento da comunicação, ou função incorreta de comunicação para o cabo em uso, que resulta na inversão das linhas de transmissão e recepção, bem como na inversão de uma ou mais linhas de controle ("*handshaking*"). Felizmente, os *drivers* utilizados são bastante tolerantes aos erros cometidos (CANZIAN, 2000).

Geralmente, usa-se um cabo de dados serial, como o da Figura 3, para fazer a conexão de dois dispositivos que usam a interface RS 232 para se comunicar. Na maioria das vezes, esses cabos possuem em suas extremidades dois conectores DB9 (Conector com 9 pinos), mas o conector DB25 (25 pinos) também é usado.

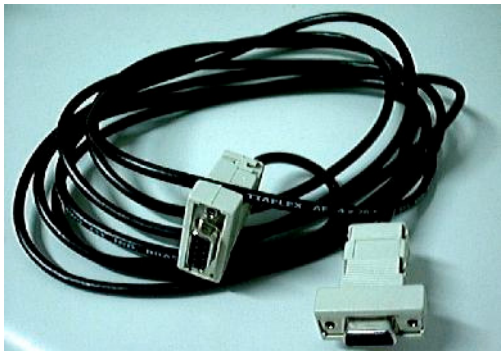

Figura 3 : Cabo de dados serial com dois conectores DB9.

No trabalho proposto, os dados são enviados ao computador por meio de uma porta serial RS 232 disponível na placa de aquisição de dados.

2.4 CÓDIGO E MODOS DE OPERAÇÃO

Nesta parte do trabalho são apresentados, de uma forma geral, alguns assuntos pertinentes à comunicação de dados, como os formatos de codificação e de transmissão, modos de operação, tipos de configurações e verificação de erros. Para grande parte do item 2.4 foi usado como base no livro de Alves (1994) por apresentar uma linguagem acessível e objetiva e os assuntos em questão já estão bem explorados existindo um consenso entre a maioria dos especialistas da área de comunicação de dados sobre esses assuntos.

2.4.1 Formatos de codificação

Alves (1994) afirma que o sistema binário é o sistema usado para representar os dados em uma comunicação de dados, de uma forma geral consiste em estado do bit “1” (ligado) ou “0” (desligado). Desta forma qualquer caractere alfanumérico pode ser representado por uma seqüência de 0´s e 1´s. Os códigos mais comuns são o ASC (*American Standard Code*) e EBCDIC (*Extended Binary Coded Decimal Interchange Code*), que utilizam oito bits, possibilitando com isso até 256 combinações diferentes. No processamento de dados, é usada

uma variação do ASC, que é o ASCII. Esse código representa um caractere de informação com sete bits, sendo possíveis 128 combinações diferentes, boa parte é reservada para representar números, letras maiúsculas e minúsculas, sinais de pontuações e alguns símbolos, os demais são reservados para comandos. É esse tipo de codificação que é usado no trabalho proposto (ALVES, 1994).

2.4.2 Formatos de transmissão

Os formatos mais básicos que se encontram nos meios de transmissão são o serial e o paralelo. A transmissão serial se caracteriza pela transmissão de um bit por vez. O cabo que conecta os dispositivos pode ser mais longo, em virtude de características especiais do sinal que é transmitido. Diversos protocolos de comunicação operam sobre comunicação serial, como, por exemplo, a comunicação com modens.

A **comunicação serial** pode ser de dois tipos: síncrona e assíncrona. Na comunicação síncrona, o transmissor e o receptor devem ser sincronizados para a troca de dados. Geralmente, uma palavra de SINCRONISMO é utilizada para que ambos ajustem o relógio interno, após a sincronização os bits são enviados, seqüencialmente, até uma quantidade pré-combinada entre os dispositivos. Já na assíncrona, não existe a necessidade de sincronização entre os dispositivos, uma vez que os caracteres são transmitidos individualmente e não em blocos como na comunicação síncrona. A transmissão de cada caractere é precedida de um bit de *start* e terminada por 1 (1/2 ou 2) bit (s) de *stop*. A Figura 4 apresenta um quadro assíncrono típico, que contém um bit de *start*, 8 bits de dados, um bit de paridade e um bit de parada. A paridade pode ser par ou ímpar, quando a paridade é par o bit de paridade é gerado de modo que o número de 1s da palavra mais o bit de paridade, seja par (DUQUE, 2001).

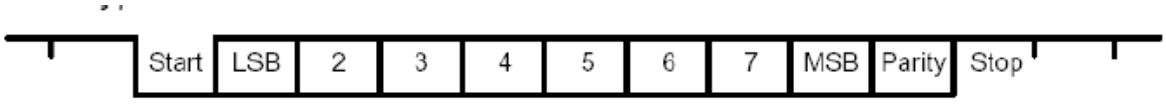

Figura 4 : Quadro assíncrono padrão.

Fonte: Duque (2001).

De acordo com DUQUE (2001), na **comunicação paralela,** um byte, também chamado de palavra, é transmitido de uma vez só ao longo de um barramento, ou seja, todos os bits dessa

palavra são transmitidos simultaneamente. Esse tipo de comunicação tem como característica alta velocidade de transmissão, comparada com a comunicação serial, porém, atinge distâncias curtas de no máximo dois metros de um sistema ao outro.

No trabalho, é usada a comunicação serial assíncrona por se tratar de um modo de transmissão que mais se adequa a este caso devido, suas características.

Na seção a seguir, são apresentados os modos de operação de comunicação de dados.

2.4.3 Modos de operação

O canal de comunicação pode ser definido como um meio físico que interliga dois ou mais sistemas e é usado para transmitir informações de um ponto para outro. Suas limitações dependem das características físicas impostas pelos meios, que podem ser *Simplex*, *Half Duplex e Duplex.*

No modo **Simplex,** o fluxo de informação se dá sempre em um único sentido, ou seja, do sistema origem para o sistema destino (Figura 5) (ALVES, 1994).

Figura 5 : Modo Simplex.

Fonte: Alves (1994, p.45).

No modo ***Half Duplex***, os dois sistemas estão aptos a transmitir os dados, mas essa transmissão não pode ser simultânea, ou seja, só é permitido que um sistema transmita por vez (ver Figura 6). Quando um dos sistemas deseja transmitir, este envia seu sinal de sincronismo e a outra parte que recebe esse sinal, fica pronta para receber os dados. Depois de transmitir todos os dados, o sistema que originou a transmissão, envia um sinal terminando a comunicação (ALVES, 1994).

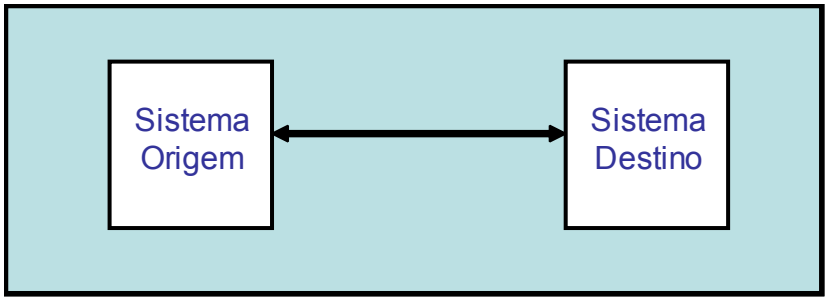

Figura 6 : Modo Half Duplex.

Fonte: Alves (1994, p.45).

No modo **Duplex** ou ***Full Duplex*****,** a transmissão ocorre nos dois sentidos e de forma simultânea (Figura 7) e podem ser empregados uma ou duas linhas. Quando usadas duas linhas, uma é reservada para a transmissão e a outra para a recepção onde a freqüência de comunicação é a mesma. Já com uma linha, o meio é usado tanto para transmitir quanto para receber, neste caso tem que se usar duas freqüências diferentes que não interfiram entre si, uma freqüência para transmissão e outra para recepção. Geralmente, são empregados filtros para evitar a interferência entre as freqüências (ALVES, 1994).

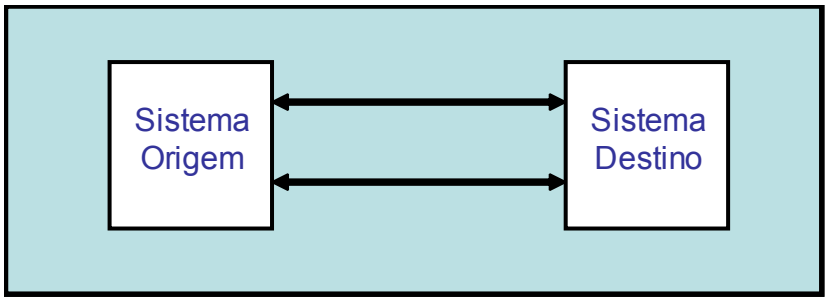

Figura 7 : Modo Full Duplex.

Fonte: Alves (1994, p.46).

Neste trabalho, é usado o modo *Full Duplex* com dois fios, onde consiste na divisão do meio físico em tempos iguais. Esses tempos são como se fossem canais de dados no qual cada sistema poderá transmitir seus dados. Mas todas essas funções são de responsabilidade do **SCOUT – DX PSB 21373**.

A seguir, são apresentados os tipos de configurações físicas possíveis.

2.4.4 Tipos de configuração

Na comunicação de dados, são possíveis dois tipos de configurações físicas bem distintas: a configuração ponto-a-ponto e a configuração multiponto.

A **ponto-a-ponto** pode ser entendida como se existisse apenas um sistema no enlace. Para esse tipo de ligação, não existe a situação de *Polling* e *Select* entre os sistemas e quando um dos sistemas deseja transmitir é empregado um ”convite”, mas para evitar que dois sistemas enviem o “convite” simultaneamente, normalmente, é definido qual é o sistema primário e qual é a secundário. Quando se inicia uma transmissão, é enviado um **ENK** (*Enquiry),* que é um caractere de controle, pela transmissora e esta fica aguardando um caractere **ACK** (*Affirmative Acknowledgement*) se o sistema receptor estiver pronto para receber a mensagem, caso contrário, receberá um NACK (*No Affirmative Acknowledgement*).

A ligação **multiponto** é aquela em que vários sistemas estão interligados no mesmo enlace. A controladora de terminais possui uma habilidade para controlar vários terminais e cada terminal é reconhecido na rede por seu endereço. A controladora também tem seu endereço, pois pode haver mais de uma controladora no enlace. Neste caso, é usada a técnica de *polling* e *select*, para controlar o fluxo de informações na rede (ALVES, 1994).

Na ligação entre o PABX e TI é usada uma configuração ponto-a-ponto. Quando se conecta o Analisador de Protocolo proposto, o sistema adquire características de um sistema multiponto.

Por fim, é apresentada a seguir uma breve descrição sobre técnicas de verificação de erros.

2.4.5 Verificação de erros

Quando um sistema recebe um bloco de dados, deve verificar se os dados não estão corrompidos. Existem vários métodos para a verificação de erro, mas os mais comuns são o **VRC** (*Vertical Redundancy Checking*), o **LRC** (*Longitudinal Redundancy Checking*) e o **CRC** (*Cyclic Redundancy Checking*). Quando é usado o protocolo HDLC o sistema receptor deve enviar a seqüência RR (*Receiver Ready*) ou REJ (*Reject*) dependendo do caso, para o sistema transmissor, este por sua vez enviará um novo bloco ou retransmitirá aquele com erro, dependendo do caractere enviado pelo sistema receptor (ALVES, 1994).

O método **VRC** usa a técnica de paridade par ou ímpar. O método **LRC** já é mais eficaz que o VRC, pois checa todos os bits da mensagem tanto no sistema transmissor quando no receptor, gerando contadores de bits “1” para cada bloco de mensagem que serão comparados mais tarde. Se os contadores do sistema receptor e transmissor não coincidirem significa que a mensagem contém erro (ALVES, 1994).

O **CRC** é o método mais confiável entre os três, mas também é o mais complexo. Geralmente, são empregados o CRC-12 e CRC-16 para códigos de seis e oito bits respectivamente, mas existem outros modos como o CRC-32. O número indica o grau do polinômio em base 2. Quanto maior o grau, maior é a capacidade de detecção de erros. Este funciona da seguinte forma: um bloco de dados é dividido por um polinômio padrão, resultando em um resto da divisão. Este resto é transmitido junto com o bloco de dados. No receptor é feita novamente a divisão do bloco de dados, mais (+) o resto pelo polinômio padrão, se o resultado for zero, então não houve erros de transmissão (ALVES, 1994).

Neste trabalho o **SCOUT – DX PSB 21373** é o responsável pela correção de erros o qual usa o método CRC 16 por se tratar de um método bastante eficiente e que melhor se adequou ao caso em questão.

2.5 MEIOS DE TRANSMISSÃO

Segundo Alves (1994), o meio de transmissão é o meio por onde trafegam os fluxos de dados entre dois pontos. O termo linha é muito usado e pode ser um par de fios, um cabo de fibras óticas ou cabos coaxiais etc.

2.5.1 Par trançado

O par trançado, ou par de fios, são arranjados assim, pois permite a diminuição da indução de ruídos e mantém as propriedades elétricas constantes, este pode ser usado na transmissão analógica e digital. O par trançado sobre influência do meio externo e a perda de energia, é proporcional à distância. Isto justifica a limitação de distância imposta onde este condutor pode ser usado. Outros fatores que influenciam também na rede que usa o par trançado são: a

qualidade do condutor, bitola dos fios utilizados e as técnicas usadas para dirigir a informação ao longo do enlace (ALVES, 1994).

Os efeitos de indução do meio externos podem ser minimizados usando uma blindagem no par de fios. O uso de par trançado, sem blindagem, com conectores RJ-45 são muito usados em redes de computadores. A principal vantagem de se usar esse meio para se transmitir dados é o baixo custo e a facilidade de conexão e de manutenção em relação ao outros meios conhecidos (ALVES, 1994).

2.5.2 Fibras óticas

Fibras óticas transmitem sinais e luz codificados dentro do espectro de freqüência do infravermelho. A luz é enviada através de um cabo ótico, geralmente feito ou de material plástico ou vidro, revestido por material de baixo índice de refração. Os sistemas de fibras óticas são compostos, além dos cabos, por dois conversores, um converte sinais elétricos para sinais ópticos e outro usado para fazer a conversão do sinal óptico para sinal elétrico. Estes sistemas possuem, além disso, um transmissor e um receptor. No transmissor, existe uma fonte de luz que pode ser de dois tipos: LED (diodos eletroluminescentes) ou LASER e no transmissor tem um fotodetector (Alves, 1994).

De acordo com Alves (1994), os sistemas de fibras óticas se caracterizam pela alta taxa de transmissão dos dados, isto é devido à atenuação do sinal na fibra ótica não depender da freqüência, neste caso a atenuação é causada por distorção ou absorção da luz por elementos do condutor, deste modo a qualidade do material usado para fabricar a fibra é fundamental para seu bom desempenho. As principais desvantagens dos sistemas de fibras óticas são sua dificuldade na instalação e manutenção, por possuírem dimensões muito pequenas e de difícil conexão aos dispositivos externos, tudo isso implica em um alto custo na implementação de sistemas que usam fibras óticas (ALVES, 1994).

2.5.3 Sem Fio

Sem fio ou *Wireless* é o meio de transmissão não guiado e que mais sofre interferência do meio externo, por este usar o ar como meio para transmitir seus dados. Mas, em contrapartida, é o que mais cresce no mercado em termos de utilização (SOARES, 1997). Esse crescimento

é devido a sua flexibilidade, mobilidade e fácil uso por parte do usuário, eliminando todo e qualquer tipo de fio ou cabo. A comunicação *sem fio* abrange desde aplicações em rede locais sem fio até comunicações via satélite (SOARES, 1997).

As ondas eletromagnéticas se encarregam pelo transporte dos sinais. Os sinais de rádio são emitidos por um sistema transmissor e captados por um sistema receptor. Os principais parâmetros nos sistemas de freqüência de rádio são: a freqüência, a potência e a modulação. A freqüência de uma onda de rádio é expressa em *hertz* (Hz) e pode chegar a mais de 30GHz. A potência é expressa em *Watts* e, de uma forma geral, é esta que limita a distância de um sistema sem fio. Quanto maior a potência maior a distância que um sistema pode se comunicar. A modulação se refere ao método pelo qual a freqüência da portadora do transmissor é modificada para conduzir os dados. Os dois métodos mais comuns são o AM (*Amplitude Modulation*) e o FM (*Frequency Modulation*).

Alguns requisitos mínimos são necessários para que essa comunicação ocorra de forma satisfatória, assim sendo, o sinal deve ser transmitido com potência suficiente para que possa ser recuperada pelo receptor, a propagação deve ocorrer com a mínima distorção possível e essas características devem ser mantidas independentes da tecnologia e distância aplicadas ao sistema.

Como comentado anteriormente, a conexão do TI ao PABX usa o par trançado, por se tratar de um meio barato, de fácil conexão e manutenção usada largamente na área de telefonia (JORDAN; CHURCHILL, 1994).

O próximo capítulo aborda a estrutura e tecnologias de PABX, sendo este o ambiente de desenvolvimento deste trabalho.

3 VISÃO GERAL DA TECNOLOGIA DE PABX

Nesta seção, são apresentados um breve histórico sobre PABX, as características básicas de PABX tradicionais e a nova tendência de PABX IP. É apresentada também a plataforma IMPACTA INTELBRAS, juntamente com exemplos de analisadores de protocolo e o protocolo PCPTI, que será usado no decorrer deste trabalho.

3.1 INTRODUÇÃO

O termo PABX se refere a qualquer sistema automático, de propriedade de uma organização, que desempenhe a função de comutação, tanto para assinantes internos quanto para externos por meio do acesso à tradicional rede de telefonia (MARTINS, 2002).

A história da evolução do PABX seguiu praticamente em paralelo ao desenvolvimento das centrais públicas de telefonia, que teve sua origem pouco depois da invenção do telefone (BELLAMY, 1991). O objetivo das centrais públicas automáticas era permitir que a pessoa que desejasse telefonar pudesse se conectar com o destino automaticamente, sem a necessidade da telefonista. Para isso, o aparelho enviava sinais elétricos especiais para certos instrumentos da central telefônica, que por sua vez ligavam a pessoa com o telefone desejado. O primeiro sistema que desempenhava tal função surgiu em 1879, desenvolvido pelos irmãos Thomas e Daniel Connelly, juntamente com Thomas J. McTighe (MARTINS, 2002). O primeiro PBX (*Private Branch eXchange*) com controle computadorizado apareceu por volta de 1963, antes do sistema no. 1 ESS da AT&T (BELLAMY, 1991). Tais sistemas, onde para cada conexão entre assinantes existem fios (ou circuitos) dedicados dentro das centrais, são classificados como sistemas de tecnologia espacial. Essa denominação surge depois da introdução da tecnologia temporal, onde os circuitos são compartilhados de forma que para cada fonte está associado um intervalo regular de tempo no qual esta pode enviar e receber sinais elétricos. Desde então, uma grande quantidade de fabricantes tem lançado sistemas de PABX com controle computadorizado num mercado que se tornou um dos mais competitivos e inovadores.

3.2 ARQUITETURA BÁSICA

Um sistema PABX é composto por uma série de interfaces, específicas para determinados tipos de funções. Algumas são essenciais num sistema desse tipo, como as interfaces para os assinantes internos (conhecidas como interfaces de **ramal**), as interfaces para as STFC (Sistema de Telefonia Fixa Comutada), que possibilitam o acesso aos assinantes externos (conhecidas como interfaces de **tronco**, **linha** ou **juntor**) e a UCP (Unidade Central de Processamento), talvez a parte mais importante de uma arquitetura de PABX é responsável pela execução do *software* de comunicação que opera todas as funcionalidades do sistema. Estas interfaces permitem que o PABX desempenhe sua principal função que é a comutação de seus circuitos para que pessoas possam se comunicar. A Figura 8 apresenta uma estrutura básica de PABX. Outras interfaces são muito particulares, como interface para porteiro eletrônico, interface para equipamentos que utilizam infravermelho etc. Estas interfaces, normalmente, são proprietárias, ou seja, são projetadas especificamente para um determinado sistema, não havendo, portanto interoperabilidade física entre fabricantes (diferente do que ocorre, por exemplo, no mundo da informática) (INTELBRAS, 2006).

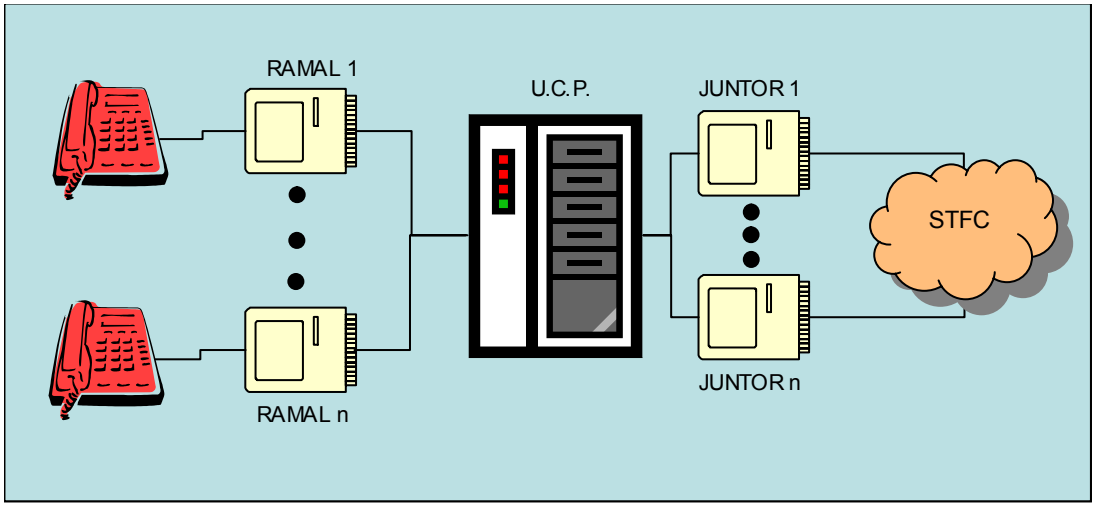

Figura 8 : Arquitetura Básica de um PABX.

Apesar da não haver compatibilidade física entre elas, existe geralmente a interoperabilidade funcional. Por exemplo, pode-se conectar uma interface de ramal analógico de um sistema PABX de um fabricante em uma interface de juntor analógico de um sistema PABX de outro

fabricante, isto é chamado de sub-sistema. A Figura 9 apresenta um PABX funcionando como sub-sistema de outro.

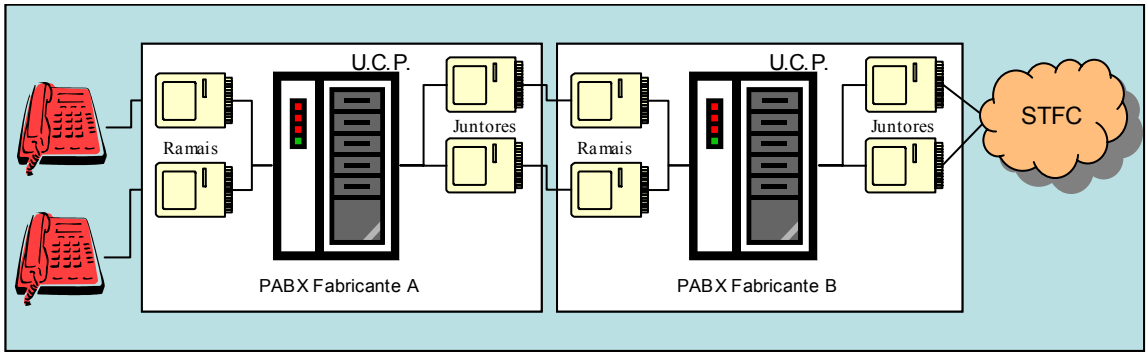

Figura 9 : Sub-Sistema de PABX.

Até há pouco tempo, os sistemas PABX empregavam quase que exclusivamente a tecnologia temporal, mas após a explosão da Internet muitos sistemas passaram também a adotar a tecnologia IP, tornando-se ou sistemas híbridos ou puramente IP.

3.3 PABX IP

Sato (2004) afirma que a rede de telefonia tradicional está desatualizada diante do mercado global, pois suas funções são limitadas e estão ficando ultrapassadas em relação às necessidades das empresas. Muitos estão convergindo à rede de dados e de voz para simplificar sua operação, obter novas funcionalidades, reduzir os custos e aumentar a produtividade dos funcionários.

A Telefonia IP ou VoIP (Voz sobre IP) é uma tecnologia que permite realizar chamadas telefônicas sobre uma rede de dados IP como se estivesse utilizando a rede STFC (Sistema de Telefonia Fixa Comutada). O sucesso da VoIP é devido ao baixo preço nas ligações de longa distância e por causa das novas funcionalidades para a telefonia, permitindo a convergência de serviços de dados, voz, fax e vídeo numa única rede IP.

Esta nova tecnologia está redefinindo a arquitetura de um PABX. Muitos dos componentes são distribuídos ao longo da rede IP para transmitir informações de voz e controle da ligação. A Figura 10 apresenta uma arquitetura básica de um PABX IP (SATO, 2004).

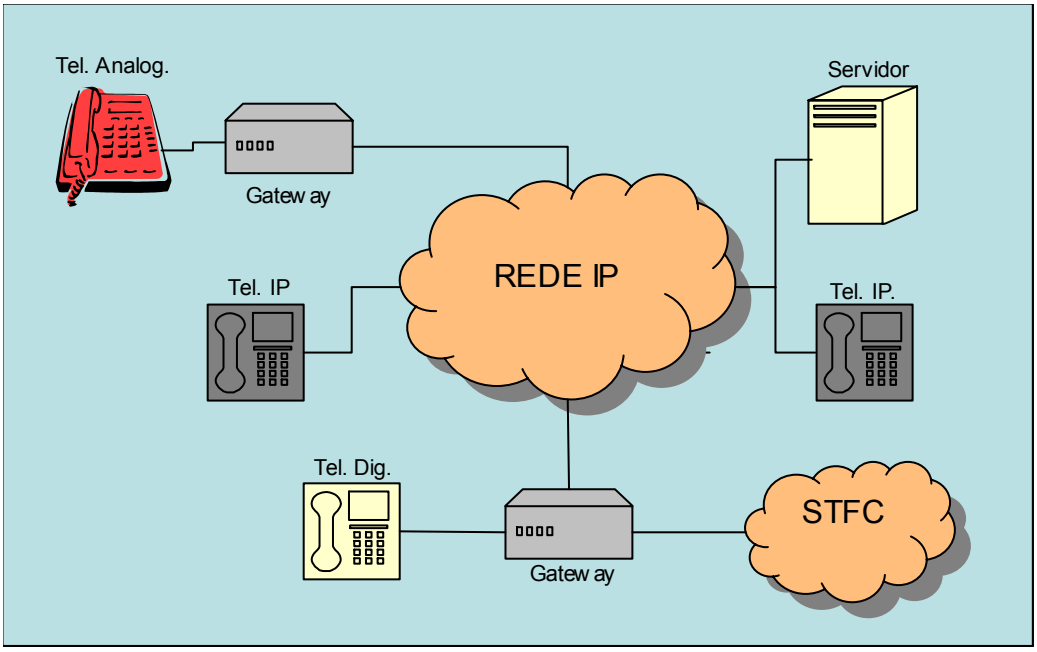

Figura 10 : Arquitetura PABX IP.

Fonte: Sato (2004).

Segundo Sato (2004), os principais componentes da tecnologia VoIP são:

- **servidor**: é o responsável por executar aplicações num sistema operacional padrão;
- ***endpoints*** ou dispositivos de ponta: são os telefones IPs que conectam-se diretamente na rede IP. Esses equipamentos necessitam de um endereço IP. Diferente do PABX tradicional, dois telefones IPs conversam diretamente, sem utilizar recursos do servidor;
- ***gateway***: são interfaces ou equipamentos que convertem a sinalização e o canal de voz para a rede IP, fazendo a integração com a rede STFC e para permitir utilizar os telefones analógicos ou digitais existentes, reduzindo os custos da migração para a nova arquitetura;
- **módulo de interconexão**: é realizado por meio da rede IP. Pode haver uma degradação na qualidade da voz se acontecer algum congestionamento ao longo do trajeto dos dados.

Sato (2004) afirma que muitas empresas estão acreditando que a Telefonia IP é o futuro das telecomunicações e estão implantando os sistemas PABX IP para usar os recursos avançados desta nova arquitetura. Utilizar a rede de dados para interligar todas as unidades da empresa por meio de um PABX IP possibilita uma grande flexibilidade, escritórios remotos podem ser

incorporados dentro de uma única plataforma global de sistema de comunicação e os aplicativos corporativos podem ser facilmente integrados com as aplicações de telefonia no ambiente IP. Isto permite que os operadores do PABX IP utilizem ferramentas familiares para operar o sistema. Outra vantagem é que servidores PABX IP possibilitam que empresas cresçam seus sistemas de telefonia de acordo com o crescimento da empresa, resultando em diminuição dos custos. Mas existem ainda alguns problemas como, por exemplo, a segurança. Para resolver isso, novas técnicas de segurança estão sendo desenvolvidas para impedir que ataques de *hackers* causem a paralisação das redes. As dificuldades e os limites desta nova arquitetura estão sendo superados e a confiabilidade e disponibilidades desta nova arquitetura estão cada vez melhores. Espera-se que essas novas capacidades continuem evoluindo do PABX tradicional para o PABX IP, cheio de novas aplicações que estão mudando a maneira de se comunicar, aumentando a produtividade, reduzindo os custos e tendo grande mobilidade.

3.3.1 ASTERISK

O ASTERISK PBX é na opinião de Gonçalves (2005) a revolução na área de PABX baseado em *software* e telefonia IP. Assim que essa tecnologia se disseminar e atingir uma grande parte do mercado, fará com que o PABX de qualquer empresa possa falar com o PABX de qualquer outra por meio da Internet. A economia em DDD e DDI é só a "ponta do Iceberg", dessa nova tecnologia.

O Asterisk é um *software* de PABX que usa o conceito de *software* livre GPL (*General Public License*), que é executado em Plataforma *Linux* e em outras plataformas UNIX, com ou sem *hardware* conectado à rede pública de telefonia, PSTN (*Public Service Telephony Network*) o que permite uma conectividade em tempo real com as redes VOIP.

Com o Asterisk conectado à rede de dados de uma empresa, é possível que sejam criados novos serviços de telefonia como, por exemplo, conectar empregados trabalhando em qualquer lugar do mundo ao PABX do escritório, conectar escritórios em vários estados, permitir que seus empregados possam ter acesso ao PABX da companhia quando estiverem viajando, entre outras. GOLÇALVES (2005).

Para Gonçalves (2005), o Asterisk é o Apache[1] da Telefonia. Este apresenta ainda várias vantagens, porém com algumas limitações:

Redução de custo extrema: se for comparar com um PABX normal, analógico de quatro troncos e dezesseis ramais, talvez a diferença seja pequena, mas quando se compara com equipamentos que apresentam serviços mais avançados como VoIP, URA (Unidade De Resposta Audível) e DAC (Distribuidor Automático de Chamadas), a diferença vai à mais de dez para um em custo (GONÇALVES, 2005).

Ambiente de desenvolvimento fácil e rápido: pode ser programado em linguagem C usando as APIs nativas, ou em qualquer outra linguagem usando AGI (*Asterisk Gateway Interface*).

Rico e abrangente em recursos: poucos são os recursos encontrados em PABX vendidos no mercado que não possam ser encontrados ou criados no Asterisk.

Roda em Linux e é de código aberto: uma das coisas mais interessantes do Linux é a comunidade de *software* livre, milhares de questões e relatos de problemas são enviados através dos fóruns de *software* em código aberto.

Limitações de acesso à Rede Pública no Brasil: Ainda falta no *Asterisk* um *driver* para acesso a R2[2] Brasil com código aberto. Existem algumas implementações no Brasil, mas o código por enquanto está fechado, isso limita o acesso a rede pública. Felizmente, em Santa Catarina, tanto a GVT como a Brasil Telecom dispõe de sinalização ISDN[3] (*Integrated Services Digital Network*). Em alguns lugares como São Paulo, é difícil conseguir um ISDN e o mais comuns são ainda circuitos E1[4] com sinalização R2 (INTELBRAS, 2006).

---

[1] O `Apache` é um servidor Web configurável, robusto e de alta *performance*.

[2] R2 É o tipo de sinalização de linha (isto é, que envia informações como ocupação, desconexão, atendimento, etc.) mais utilizada em juntores digitais e que caracteriza-se por codificar as informações de sinalização em grupos de quatro bits (2 para TX e 2 para RX) por canal.

[3] ISDN – ou RDSI (Rede Digital de Serviços Integrados) é a digitalização da rede telefônica para tráfego simultâneo de voz, dados, imagens, aplicações e serviços multimídia.

[4] Sistema de transmissão a 2.048 Mbps, com 32 canais digitais, cada um com uma velocidade de 64kbps, sendo 30 canais de voz ou dados, um canal para sincronismo e um canal para sinalização telefônica.

Limitações da arquitetura: o *Asterisk* usa a CPU do servidor para processar os canais de voz, ao invés de ter um DSP dedicado para cada canal. Enquanto isto permitiu que o custo fosse reduzido para as placas E1/T1, o sistema fica muito dependente do desempenho da CPU.

É recomendável preservar ao máximo a CPU do *Asterisk*, executá-lo sempre em uma máquina dedicada e testar o dimensionamento antes de implantar, deve também, de preferência, ser implementado sempre em uma VLAN específica para VoIP, qualquer tempestade de *broadcasts* causadas por *loops* ou vírus pode corromper o seu funcionamento, devido ao uso de CPU das placas de rede quando este fenômeno acontece.

O *Asterisk* necessita de protocolo de sinalização para estabelecer as conexões, determinar o ponto de destino e também questões relacionadas à sinalização de telefonia como campainhas, identificador de chamadas e desconexão. É comum utilizar o SIP (*Session Initiated Protocol*), muito embora outros protocolos também sejam expressivos no mercado como o H.323, o MGCP (*Media Gateway Control Protocol*) e, recentemente, o IAX (*Inter-Asterisk eXchange*) que é excepcional quando se trata de NAT (*Network Address Translation*) (GONÇALVES, 2005).

3.3.2 Solução CISCO

A solução de telefonia IP da CISCO é baseada na arquitetura AVVID (Arquitetura de Voz, Vídeo e Dados Integrados) que permite a criação de novas aplicações que ampliam as possibilidades da telefonia convencional. Mesmo assim, é possível a total integração com o sistema telefônico e de correio de voz existentes, permitindo que se faça uma transição suave, colocando os novos dispositivos à medida que houver necessidade. A arquitetura AVVID permite uma infra-estrutura para criação de uma única rede convergente que pode transportar voz, vídeo e tráfego de dados, simultaneamente, mantendo a qualidade de serviço e segurança para as redes corporativas. Como a Telefonia IP usa as redes de comunicações de dados existentes para efetuar chamadas telefônicas, isto gera uma grande economia, como por exemplo, nas chamadas interurbanas que podem custar o valor de uma ligação local. A plataforma de telefonia IP Cisco é composta basicamente por telefones IP e o **Agente de processamento de chamadas (Call Manager®)** (CISCO, 2006)**.**

Os telefones IP são dispositivos que têm todas as funcionalidades de um telefone digital com funções muito mais sofisticadas como a habilidade de acessar páginas *Web*, podendo ser aparelhos de mesa ou *SoftPhones* dando uma mobilidade sem comparação com a telefonia convencional. Como esses aparelhos se conectam com a rede de comunicações por meio de uma conexão *Ethernet* convencional, as ampliações da rede e reconfigurações são virtualmente instantâneas e sem custo. Os usuários do sistema só precisam conectá-los à tomada *Ethernet* e o telefone, automaticamente, se registrar no sistema *Call Manager,* ou então, simplesmente, utilizam a funcionalidade de se logar diretamente no aparelho, fornecendo o seu nome (enquanto usuário do sistema) e uma senha para uso da rede de telefônia IP. Tais funcionalidades, eliminam custos e o tempo de espera por técnicos e equipes de instalação que repassavam cabos telefônicos e reconfiguravam as conexões nos armários distribuidores para conduzir o antigo telefone até o local de trabalho (CISCO, 2006).

O *Call Manager* é o cérebro do sistema de telefonia IP da Cisco, este é o responsável pelo gerenciamento das chamadas telefônicas e distribuição de todas as funcionalidades e possibilidades da telefonia IP para os demais dispositivos que compõem a rede telefônica IP. Estes dispositivos periféricos complementam e agregam à solução funcionalidades de *gateways* de voz para a rede pública e para os sistemas de telefonia tradicional. Com essa solução é possível criar uma estrutura centralizada de processamento de chamadas telefônicas, facilitando a administração e reduzindo os custos ao dispensar a instalação desta funcionalidade nos sites remotos. Nesta situação, é possível garantir a integridade da rede telefônica IP por meio do uso da característica SRST (*Survivable Remote Site Telephony)* da Cisco, que é uma versão do *software* Cisco IOS® para os roteadores. No caso de alguma falha ocorrer nos canais de comunicação da localidade remota, o SRST presente no roteador proverá as funcionalidades básicas do *Call Manager* até que o canal de comunicação seja reparado (CISCO, 2006).

Os ***Gateways* de Voz** são responsáveis por interconectar a Telefonia IP com a rede pública de telefonia. Possuem completa linha de interfaces analógicas e digitais para voz. Os *gateways* de voz permitem que as ligações telefônicas entrem e saiam da corporação e ainda que o sistema de Telefonia IP se integre com PABXs convencionais e Correios de Voz legados, possibilitando total flexibilidade e liberdade para migração das plataformas antigas para a Telefonia IP Cisco.

Com a Telefonia IP, é possível criar uma plataforma aberta para o desenvolvimento de aplicações dando às organizações uma grande oportunidade de lidar com as mudanças no ambiente em que vivem não deixando-as dependentes de uma única tecnologia. Esta ainda permite que usuários sejam produtivos mesmo longe do escritório. Estas soluções foram desenhadas para atender as necessidades atuais de mobilidade, característica muito importante, principalmente para vendedores, executivos, consultores, aqueles que trabalham em casa e as equipes de manutenção, que gastam muito tempo fora do escritório, mas ainda assim necessitam acessar a rede independente de onde estejam (CISCO, 2006).

A seguir, é apresentada a nova plataforma de PABX INTELBRAS, chamada de plataforma IMPACTA.

3.4 A PLATAFORMA IMPACTA (INTELBRAS)

A plataforma IMPACTA é um novo conceito de equipamento para a INTELBRAS, esta que sempre usou um *software* dedicado para cada central e uma central para cada necessidade e, especificamente, desenvolvida para linhas telefônicas. Essa plataforma é um primeiro passo para a convergência de equipamentos de comunicação de voz com a rede de dados, já que ela apresenta além das placas tradicionais de ramais, linhas analógicas e digitais como E1 e ISDN, uma placa de VOIP (Voz sobre IP). A Figura 11 apresenta o diagrama de um PABX IMPACTA.

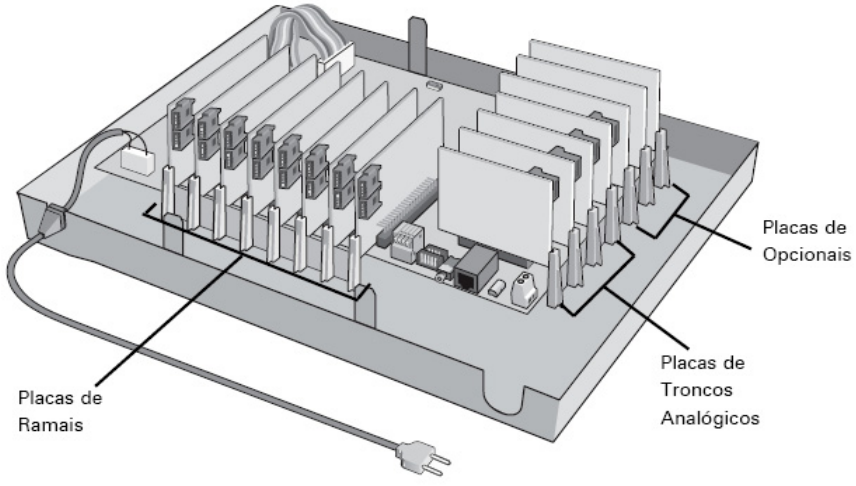

Figura 11 : Diagrama do PABX IMPACTA 68.

Fonte: INTELBRAS (2006, p.18).

O *software* dessa plataforma foi desenvolvido em uma linguagem orientada à Objeto (C++), tendo como base a idéia da arquitetura OSI, fazendo com que este seja bastante versátil, flexível e portável. Todas essas qualidades fazem com que a IMPACTA possa ser ágil em alterações que o mercado e a tecnologia impõem, podendo se configurar e adaptar-se para atender às várias circunstâncias de campo e cliente e ainda permitir que seja utilizado em outras plataformas de *hardware*. O que não era possível na plataforma antiga, já que o *software* era desenvolvido em *Assembly* de forma estruturada, o que dificultava a manutenção, alteração e inclusão de novas facilidades. Além disso, a plataforma antiga deixava o *software* "engessado" ao *hardware*, ou seja, se desejasse mudar o microprocessador todo o *software* deveria ser refeito pelo fato de cada microprocessador usar seu próprio *core*, que são os códigos e funções de programações do microprocessador. A seguir, na Figura 12, é apresentada a arquitetura de três camadas da comunicação PABX TI INTELBRAS.

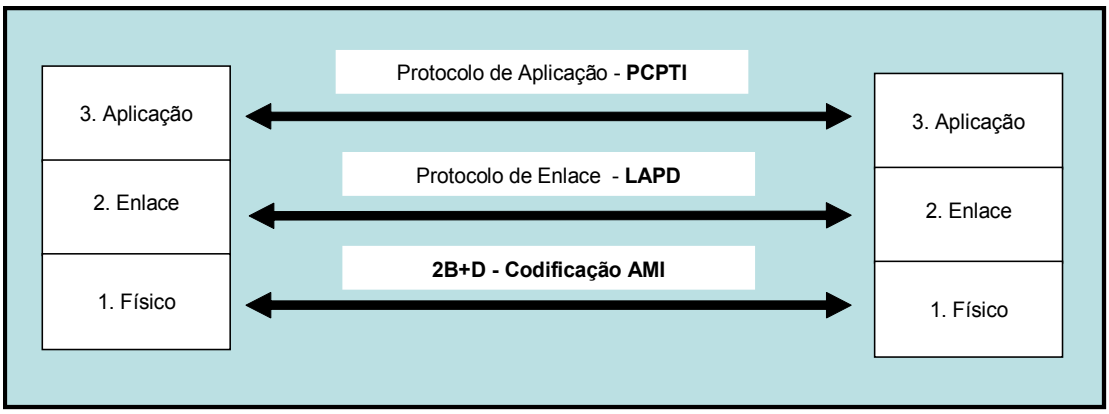

Figura 12 : Arquitetura de três camadas da comunicação PABX TI INTELBRAS.

A IMPACTA utiliza o conceito de "portas" e não mais de ramais ou juntores, se dividindo, basicamente, em quatro equipamentos. A IMPACTA 16, que é uma central de pequeno porte destinada a residências, escritórios e micro-empresas com capacidade para até 12 ramais. A IMPACTA 68, uma central destinada às pequenas empresas com capacidade máxima de 32 ramais. A IMPACTA 140 e a IMPACTA 220, são centrais consideradas de médio porte indicadas para médias empresas, que podem chegar até 80 e 160 ramais, respectivamente.

A filosofia do projeto da IMPACTA baseia-se em um equipamento que seja fácil de operar, por isso foram mantidos os códigos e operação básica das plataformas atuais da INTELBRAS, com uma numeração aberta para ramais, juntores e programações, o que permite que qualquer um destes três possa assumir qualquer seqüência de números, essa

facilidade não era permitida nas plataformas anteriores. Com isto, se a IMPACTA for substituir qualquer outro equipamento em campo, mesmo que não seja uma central da INTELBRAS, o técnico instalador pode configurar os códigos e ramais de tal maneira que coincidam com a antiga tornando transparente a mudança para o usuário (INTELBRAS, 2006).

A plataforma IMPACTA possui várias interfaces que pode ser conectada ao PABX, como o Clic Fone e a Mesa Operadora Virtual, que são softwares instalados no PC que rodam em sistemas operacionais Windows ou Linux, configurações via PDA e ainda os TIs NKT 4245 e NKT 2165.

A seguir, são listadas algumas funcionalidades não disponíveis nos PABXs anteriores e que agira estão na plataforma IMPACTA[5]:

- Placa VoIP que permite a realização de chamadas com tecnologia VoIP;
- Placa Ethernet que garante o acesso remoto do PABX por meio da rede de dados;
- Plano de numeração completamente configurável;
- Programação de funcionalidades via interface USB;
- Programação de funcilidades via interface infravermelho;
- Gravação de chamadas;
- Custo mais baixo comparado com os PABXs INTELBRAS Atuais.

O protocolo de comunicação usado entre o TI e o PABX IMPACTA é o PCPTI, que é uma modificação do protocolo LAPD. Deste modo, a seguir será apresentada uma visão geral do protocolo LAPD e posteriormente do protocolo PCPTI, sendo este último o protocolo para o qual o *software* AP foi desenvolvido.

---

[5] Todas as vantagens da plataforma IMPACTA podem ser consultadas no *site* <http://www.intelbras.com.br> do fabricante INTELBRAS.

3.5 O PROTOCOLO LAPD

Para que os equipamentos inter operem, é preciso uma série de regras, essas regras são chamadas de protocolo. O protocolo da camada 2 que é utilizado na comunicação entre o TI e o PABX é o protocolo da camada 2 do ISDN. Este protocolo é um protocolo orientado a bit e se chama *Link Access Procedures on the D-Channel* (LAPD) e é uma derivação do protocolo HDLC (*High Level Data Link Control*). O AP proposto não faz o tratamento do LAPD, que é de responsabilidade da placa PADI (Placa de Aquisição de dados). A função desta placa está detalhada no item 4.2.

A estrutura do quadro do LAPD é apresentada na Figura 13:

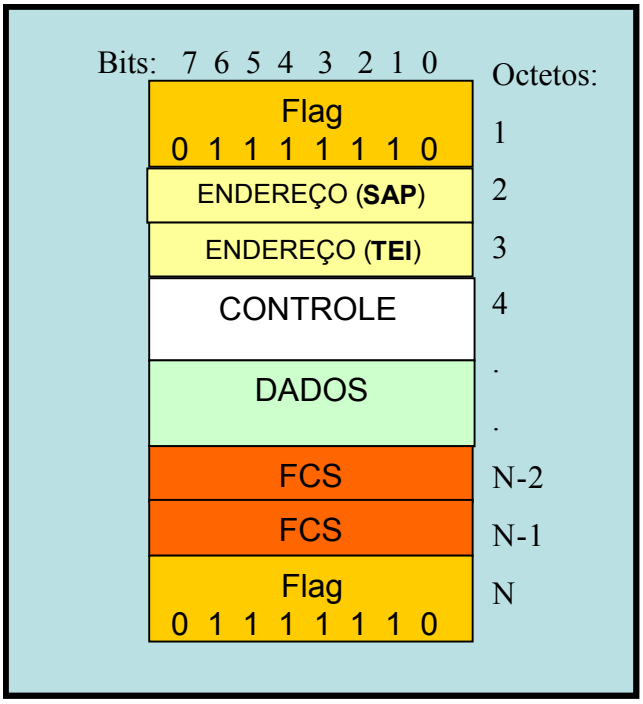

Figura 13 : Estrutura do quadro LAPD.

A camada 2 provê um enlace de comunicação sem erros entre dispositivos adjacentes. A seguir, é apresentada a descrição dos campos da estrutura do protocolo LAPD:

FLAGS: indicam o início e fim de cada quadro;

ENDEREÇO: com base na Recomendação Q.921 da ITU-T (09/97), o campo de endereço tem dezesseis bits divididos em dois sub-campos, **SAPI (***Service Access Point Identifier* **) e**

TEI (*Terminal Equipament Identifier).* A estrutura do quadro do campo de endereço é apresentada na Figura 14;

CONTROLE: identifica o tipo de quadro e pode transportar números seqüenciais e de reconhecimento. Pode ter um ou dois octetos, dependendo do tipo de quadro;

DADOS: contém as informações de camada 3 (camada de rede), dados de usuário ou informações de gerenciamento LAPD. O número de octetos neste caso é variável.

FCS: (*Frame Check Sequence)*, usado para detecção de erro.

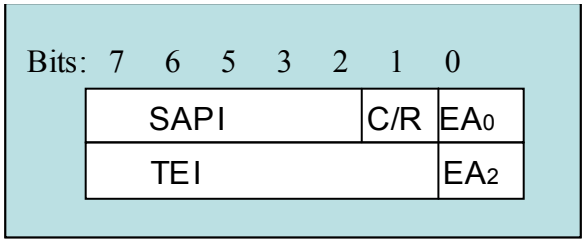

Figura 14 : Estrutura do quadro de Endereçamento.

Fonte: ITU-T (1997, p.10).

A estrutura do quadro do campo de endereço, e possui os seguintes campos;

C / R – Bit utilizado para distinguir comando (1) de resposta (0);

EA bits – *Extension Address* bit; "0" indica presença de octetos adicionais e "1": indica que é o octeto final;

TEI – *Terminal End Point Identifier* ou Identificador do Equipamento Terminal. Usado para identificar para que terminal é destinado a mensagem.

SAPI – Identificador do SAP, identifica o processo ou "aplicação" de camada 3, ou seja, identifica o ponto de acesso ao serviço. É o *link* lógico. O SAPI indica qual o processo da camada 3 deve ser chamado para tratar a mensagem. É semelhante a porta no TCP. No ISDN, o campo SAPI possui os seguintes valores e a seguintes entidades relacionadas com a camada 3:

Valor SAPI	*Entidade Relacionada na Camada 3*
0	Processos de controle de chamada
1	Comunicação modo pacote usando I451
16	Comunicação modo pacote usando X.25 PLP
63	Funções de gerenciamento da camada 2
Outros	Reservados para padronizações futuras

Tabela 1: Valores possíveis do SAPI.

Fonte: ITU-T (1997, p.10).

3.6 O PROTOCOLO PCPTI (INTELBRAS)

O PCPTI é similar ao LAPD apresentado no item 3.5, porém foi otimizado e modificado para as necessidades existentes. O AP trata esse protocolo e não o LAPD. A seguir, são apresentados a estrutura deste protocolo e seus campos de dados e controle.

A estrutura do quadro do PCPTI é apresentada na Figura 15:

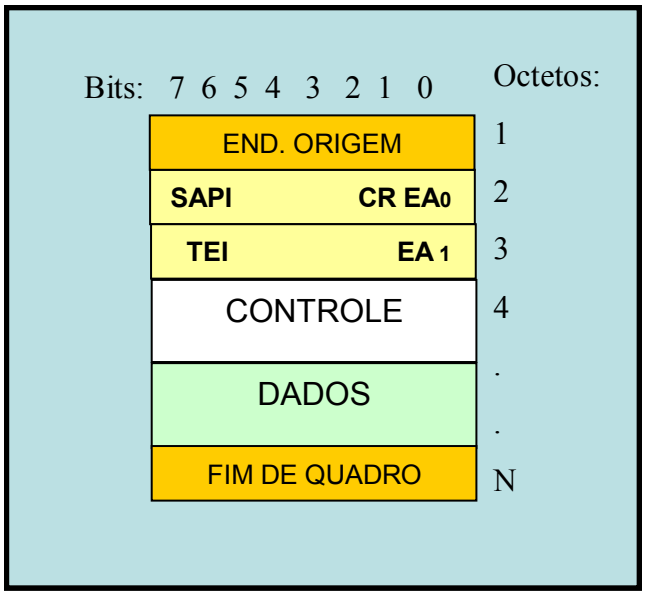

Figura 15 : Estrutura do PCPTI.

END. ORIGEM: indica quem é a fonte que gerou a mensagem; **RX** indica que a mensagem foi transmitida pelo PABX e **TX** indica que o TI que transmitiu a mensagem.

ENDEREÇO: é semelhante ao endereço do LAPD apresentado na Figura 13. Mas no caso do PCPTI por ser usada apenas a comunicação ponto-a-ponto, foi definido um número fixo (64h) para o campo TEI. Também foram desenvolvidos outros SAPIs, apresentados a seguir:

- **SAPI = 0 Processos de Controle e Estabelecimento da Chamada:** todas as mensagens referentes ao controle e estabelecimento de uma chamada devem ser enviadas com o campo **SAPI = 0**.
- **SAPI = 17 Processos de Status:** todas as mensagens referentes ao status do PABX (status dos ramais, vias, grupo, rota, etc...), devem ser enviadas com o campo **SAPI = 17**.
- **SAPI = 18 Processos de Inicialização:** todas as mensagens referentes a inicialização (parâmetros que o PABX passa durante a inicialização ou quando o programador altera algum parâmetro), devem ser enviadas com o campo **SAPI = 18**.
- **SAPI = 19 Processos de Programação:** todas as mensagens referentes a programação (pesquisa de contato na agenda, de ramal, de rota, grupo, etc..), devem ser enviadas com o campo **SAPI = 19**.
- **SAPI = 20 Processos de Transmissão de Dados**: todas as mensagens referentes a transmissão de dados (atualização de *software* do TI, SMS, transmissão da configuração do TI para um FTP, etc..), devem ser enviadas com o campo SAPI = 20.
- **SAPI = 63 Funções de Manutenção da Camada 2:** todas as mensagens referentes a camada 2, estabelecimento de *link*, pedidos de endereço, pedidos de retransmissão etc, devem ser enviadas com o campo SAPI = 63.

C/R – Bit (1) ou comando significa que mensagem foi transmitida pelo PABX e resposta (0) significa que a mensagem foi transmitida pelo TI.

EA0 – "0".

EA1 – "1".

CONTROLE: neste campo é enviado o número de seqüência dos quadros, este tem tamanho de um octetos. Caso particular que a seqüência não é respeitada é quando o PABX está iniciando a comunicação com o TI.

DADOS: contém as informações referentes ao gerenciamento e controle de chamadas ou de status do PABX, por exemplo, que número deve ser discado, qual o número do ramal que ligou para o TI ou ainda quais ramais ou linha estão ocupadas e quais estão livres etc. O número de octetos neste caso é variável.

FIM DE QUADRO: é preenchido pela palavra "\n\r".

A próxima seção apresenta algumas tecnologias voltadas para a análise de protocolos, que serviram de base para a especificação dos requisitos do *software* que foi desenvolvido neste trabalho. Ressalta-se que nenhuma destas tecnologias são adequadas para a utilização em conjunto com o ambiente de PABX da INTELBRAS, haja vista que o protocolo considerado neste caso é proprietário e por isso os produtos de mercado, descritos a seguir, não são compatíveis com o mesmo. Esta incompatibilidade e, portanto, necessidade da implementação de um *software* para este fim, se configura a principal motivação para o desenvolvimento do trabalho.

3.7 ANALISADORES DE PROTOCOLOS

Analisadores de protocolo são instrumentos capazes de monitorar o fluxo de dados de uma rede em tempo real, possuindo diferentes características, funcionalidades e diferentes formas de apresentar as informações capturadas (PINTO, 2004).

Não existe atualmente APs desenvolvidos especialmente para serem usados no desenvolvimento de um PABX, pois os fabricantes não usam arquitetura e protocolo padrão. Cada fabricante desenvolve seus próprios padrões e não divulgam, o que dificulta uma análise mais precisa de seus produtos a fim de se desenvolver ferramentas para análise que poderiam facilitar o desenvolvimento de produtos eletrônicos.

Empresas como a INTELBRAS usam ferramentas como osciloscópios e analisadores lógicos para descobrir defeitos em *hardware* e ajudar na depuração destes circuitos. A HP (*Hewlett-*

Packard) e a *Tektronix* são dois exemplos de empresas que desenvolvem esse tipo de equipamentos. A seguir, é apresentado um exemplo de analisador lógico da *Tektronix*.

3.7.1 Analisador Lógico TLA5000[6]

Possui uma interface *Windows* de fácil operação usada para verificar, depurar, monitorar e medir o desempenho de um *hardware* digital, ideal para medir a velocidade no barramento de dados desse tipo de circuito. Permite a aquisição de sinais em tempo real e um grande número de registro dos dados. Possui um sofisticado *trigger* que permite que seja detectada e identificada a ocorrência de falhas e erros de operação e de inicialização em qualquer projeto digital. Também pode ser usado como um analisador de estados. Permite a detecção de difíceis problemas como erro de lógica digital, ruídos e problemas de sincronização.

Muitos projetos podem apresentar anomalias digitais ou analógicas. Com o módulo *iView*™ pode se ver claramente, no monitor do analisador lógico, como as anomalias analógicas afetam os sinais digitais (TEKTRONIX, 2006).

A seguir, são apresentados alguns analisadores de protocolos para redes IP *código aberto* que se encontra no mercado. Devido à tendência atual dos PABXs baseados em tecnologia TCP/IP, estes Analisadores de protocolos serão ferramentas, cada vez mais úteis, na área de pesquisa, desenvolvimento e suporte a PABX.

3.7.2 Ethereal[7]

"É um analisador gráfico de protocolos de rede para ambientes Unix e Windows. Permite o exame dos dados trafegados na rede ou de registros de monitoração armazenados em disco. Neste último caso, possibilita percorrer os dados, visualizando informações de vários níveis para cada pacote" (RONCERO *et al*., 2005).

---

[7] http://www.tek.com

[7] http://www.ethereal.com

O Ethereal é talvez um dos melhores analisadores de protocolos de código aberto atualmente disponível. Por ser um *software* de código aberto é possível desenvolver e adicionar novos tipos de protocolos, bastando para isso que seja adicionando novos *plugins* ou mesmo editando o código fonte do Ethereal (ETHEREAL, 2006). São alguns recursos do Ethereal:

- disponível para UNIX e Windows;
- filtra e procura pacotes segundo vários critérios;
- mostra os pacotes com detalhes e informação do protocolo e de forma colorida, conforme especificado nos filtros;
- grava e abre arquivos com os dados dos pacotes capturados;
- importa e exporta pacote de dados para um grande número de programas de captura de dados;
- cria estatísticas;
- analisa um grande número de protocolos. A lista completa pode ser vista no *site* oficial da ferramenta.

3.7.3 IPTraf[8]

Segundo RONCERO (*et al*. 2005) "o IPTraf é um utilitário de modo texto para levantamento de estatísticas de rede para Linux. Esta ferramenta agrupa uma série de informações como o total de pacotes e *bytes* trafegados pela rede, indicadores de atividade, detalhamento do tráfego TCP e UDP, e total de pacotes e *bytes* trafegados pela estação de trabalho local". Suporta os seguintes protocolos IP, TCP, UDP, ICMP, IGMP, IGP, IGRP, OSPF, ARP, RARP. Apresenta os seguintes recursos:

- um monitor que mostra o tráfego de informações IP em uma rede de dados. Inclui *flags* de informações TCP, contador de pacotes e de bytes.

---

[8] http://iptraf.seul.org/

- interface que mostra estatística geral e detalhada de pacote IP, TCP, UDP, ICMP e outros contadores de pacotes e de *checksum.* A interface pode ser ativada por tamanho dos pacotes de dados;
- filtros de TCP, UDP e outros protocolos permitem que seja visto somente o tráfego de protocolos que interesse;
- registro dos pacotes;
- suporte para Ethernet, FDDI, ISDN, SLIP, PPP, e interfaces do tipo *loopback*.

As informações geradas pelo IPTraf pode ser de grande valor para a implantação de uma rede de dados, correção de erros em LANs e para o rastreamento de vários IPs (IPTRAF, 2005). Existe ainda o **Ksniffer** que é uma versão gráfica do IPTraf, que inclui outras funcionalidades como gráficos de uso da rede em geral ou mesmo de protocolos em vários níveis da pilha TCP/IP (RONCERO *et al*., 2005).

3.7.4 NTop[9]

NTop é um aplicativo que permite o monitoramento da atividade da rede, de forma similar à ferramenta *Top* do Unix, que informa quais são os processos que a CPU utiliza e o desempenho dela. Possui também uma interface HTML com uma série de estatísticas e gráficos (RONCERO *et al*., 2005). Suas principais vantagens são:

- interface *web*, que permite que dados e configurações possam ser acessados por meio de qualquer *browser;*
- restrições de configuração e administração via interface *web;*
- uso reduzido de CPU e memória que variam de acordo com o tamanho da rede e volume de tráfego;

[9] http://www.ntop.org/

Todas essa qualidades fazem do NTop uma ferramenta adequada para o monitoramento de um grupo de redes.

O Ntop pode ainda classificar o tráfego na rede conforme vários protocolos e vários critérios, mostra estatísticas do tráfego, armazena em disco essas estatísticas, identifica a identidade do computador do usuário e muitos outros recursos. Suporta os protocolos IPv4/IPv6, IPX, DecNet, AppleTalk, Netbios, OSI, DLC etc (NTOP. ORG, 2006).

3.7.5 TCPDump[10]

"O TCPDUMP é um programa que coloca a interface de rede em modo promíscuo, ou seja, aceitando todos os pacotes que trafegam pela rede. Possui um mecanismo poderoso de filtragem de pacotes, de modo a armazenar apenas os dados que sejam de interesse" (RONCERO *et al*., 2005). É formado por módulos como o TCPDstat que lê o arquivo do TCPDump, usando a biblioteca *pcap,* e imprime as estatísticas do registro de monitoração. A sua saída inclui o número de pacotes, taxa média de transmissão e o seu desvio padrão, o número de pares únicos de endereços fonte e destino, e o número de pacotes e de *bytes* por protocolo. Este também oferece dados úteis para se encontrar uma anomalia no registro de monitoração. Por exemplo, o tráfego intenso de ICMP ou entre um par de endereços específicos, pode ser sinal de algum tipo de ataque de negação de serviço (DoS). O **TCPSlice** também é um programa para extração de partes de arquivos de registro gerados pelo TCPDump. Este também pode ser usado para reunir vários destes arquivos. Sua função é copiar para a saída padrão todos os pacotes que estejam dentro de um intervalo de tempo especificado (RONCERO *et al*., 2005).

3.7.6 Sniffer Enterprise[11]

É uma solução de arquitetura desenvolvida pela *Network General™ Corporation* que permite não só a captura de dados, mais muito mais informações, como métricas de desempenho de uma rede e tempo de resposta. Fornece informações que ajudam na detecção de tráfego

---

[10] http://www.tcpdump.org/

malicioso que pode passar pelo *firewall* e informações que permite reduzir o congestionamento na rede. Com esse tipo de informações é possível que se tome ações rápidas para identificar e resolver problemas de desempenho em qualquer lugar da rede antes que estes impactem no serviço para o usuário ou na produtividade do negócio. Essa solução é composta basicamente por três partes; o *Sniffer Enterprise Platform, o Sniffer Enterprise Intelligence e o Sniffer Enterprise Management.*

Sniffer Enterprise Platform é o "coração" da solução, contém um incomparável conjunto de funções para análise de desempenho de rede, inclui monitoração por tempo real e histórico, busca e reparos de erros, análise da causa principal e mais, tem capacidade de ser usada como uma ferramenta de gerenciamento de rede, portável ou um sistema distribuído que poderá trabalhar virtualmente em qualquer tipo de LAN, WAN ou tecnologia ATM.

Na camada *Sniffer Enterprise Intelligence,* se encontra vários módulos de *software* que permitem que o gerente da rede possa adicionar novas funcionalidades dedicadas ao seu meio físico em particular. Ajuda a garantir a qualidade e confiabilidade de VoIP, analisa a desempenho de uma rede *sem-fio* e muito mais. Este módulo proporciona aos gerentes de rede uma ampla flexibilidade e a possibilidade de agregar valor ao serviço justificando o investimento feito na aquisição desta solução.

O *Sniffer Enterprise Management,* consiste em três importantes aplicações, o *Sniffer Enterprise Visualizer* que é uma poderosa ferramenta que apresenta um rico conjunto de informações e métricas na aplicação de desempenho da rede de dados. Essas informações podem também ser usadas como auxílio na busca e eliminação de erros, problema de resolução, capacidade de planejamento etc. O *Sniffer Enterprise Administrator* permite que toda a solução seja configurada e gerenciada como uma simples e integrada solução. E o *Sniffer Enterprise NetVigi,* dispõe de um amplo painel de controle para que seja gerenciada de forma fácil e completa a performance de rede (NETWORK GENERAL, 2005).

---

[11] http://www.networkgeneral.com/

3.7.7 Appera™ Application Manager[12]

É uma nova solução que pode ser agregada ao *Sniffer Distributed*, que estende a já consagrada análise de pacotes do *Sniffer* para uma análise de fluxo de dados de aplicação. Analisando a aplicação a partir dos dados visto na rede, o Appera disponibiliza uma visão das aplicações mais críticas da empresa tais como SAP, *PeopleSoft*, e *Oracle*, bem como aplicações desenvolvidas internamente. Quando ocorre um problema de desempenho, o Appera permite que o operador identifique rapidamente os servidores, aplicações e VLAN envolvidas e também o período de tempo em que ocorreu este problema. Provê ainda, informações para que haja controle se a aplicação estiver com o desempenho desejado em termos de banda e tempo de resposta, analisa uma conversa baseada na origem, destino, aplicação envolvida, tempo e interface, possui recursos de "*drill-down*" que permite encontrar "o que, onde, quem, quanto, por quanto tempo e como" para qualquer conversa na rede (NETWORK GENERAL, 2005).

3.7.8 *InfiniStream Network Management*[13]

Essa ferramenta permite que o administrador de redes *Fast Ethernet* ou Gigabit capture uma alta quantidade de tráfego para que se possa resolver um problema intermitente ou que já tenha ocorrido e demorou a ser reportado. Este captura todos os pacotes que passam no segmento conectado ao *hardware*, ininterruptamente. Desta forma, nada é perdido e a análise pode ser feita em pequenas quantidades de dados ou em transações que demoram um tempo e precisam de uma capacidade de captura muito grande. Uma vez solicitado, os dados são transferidos para a console do operador, onde podem ser então filtrados por meio do *software* de *Data Mining* e analisados com toda a decodificação e tecnologia *Expert* do *Sniffer*. O módulo *Sniffer Voice* também pode ser adicionado ao produto para análise de redes que contenham voz e vídeo sobre IP (NETSAFE, 2004).

---

[12] http://www.networkgeneral.com/
[13] http://www.netsafe.com.br

3.7.9 Observer®[14]

O *Observer* é um *software* analisador de rede LAN, *Wireless*, Gigabit, *Token Ring* e redes FDDI. Proporciona medidas em tempo real, captura de pacotes e decodificação. Segundo a *Network Instruments* (2006), o *Observer* oferece vários benefícios, tais como; transformar o PC *(Personal Computer)* ou Laptop em um poderoso analisador. Permite capturar, analisar e decodificar o tráfego em tempo real, fazer uma avaliação imediata de ações decorrente de mudanças na rede, monitorar graficamente em tempo real a utilização da largura de banda de dados, gravar o histórico de dados, permite ainda, fazer identificação de vírus e ataques de *hackers*. Essa ferramenta analisa mais de 550 tipos de protocolos primários e inúmeros subprotocolos, incluindo o *sem fio*. Na página do fabricante, pode-se encontrar uma lista completa de suas características e benefícios que são inúmeros (NETWORK INSTRUMENTS, 2006).

[14] http://www.networkinstruments.com/

4 ARQUITERURA DO *SOFTWARE* ANALISADOR DE PROTOCOLOS

4.1 INTRODUÇÃO

Como já comentado no Capítulo 1, o *software* construído neste trabalho tem como função coletar, interpretar e apresentar os dados que trafegam entre o PABX e o TI, tomando como base o protocolo PCPTI. A seguir, são apresentados os principais componentes (*hardware* e *software*) que fazem parte deste trabalho.

4.2 PLACA DE AQUISIÇÃO DE DADOS

A aquisição dos dados do meio físico é de responsabilidade de um *hardware* desenvolvido, por técnicos da INTELBRAS, para tal função e é chamado de PADI (Placa de Aquisição de Dados INTELBRAS). Esta placa deve ficar entre o PABX e o TI para que os dados possam ser capturados, tratados e disponibilizados para o analisador de protocolo por meio da porta serial RS-232. A PADI deve se conectar ao computador, que estará executando o *software* do AP, por meio de um cabo serial. A Figura 16 apresenta o sistema de aquisição e o AP integrados ao ambiente PABX.

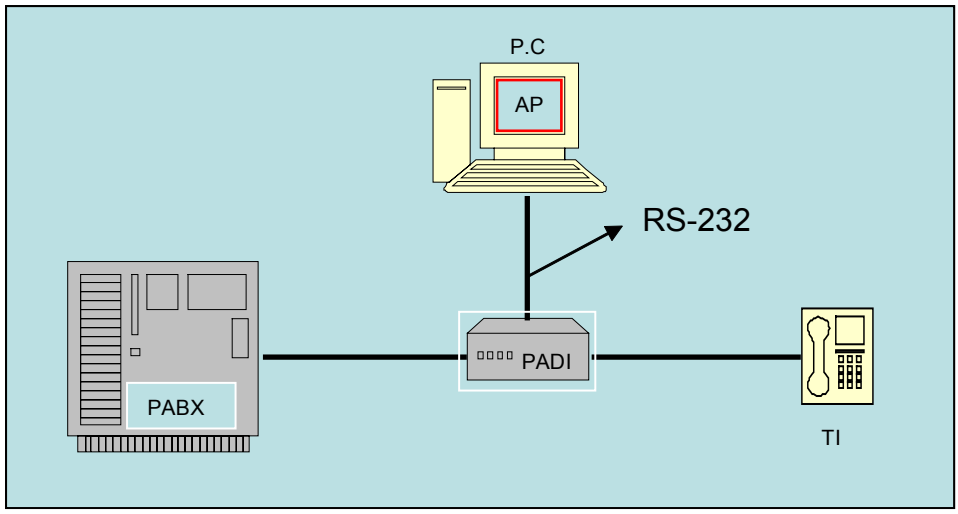

Figura 16 : Visão geral do ambiente de desenvolvimento do trabalho.

A PADI como o próprio nome sugere é o *hardware* responsável pela captura dos dados do meio físico e é composto basicamente por dois componentes; o microprocessador, no caso o Microcontrolador uPSD3234 e outro *chip,* o SCOUT DX PSB 21373, responsável pelo controle e tratamento dos dados do canal 2B+D (Dois Canais de Áudio e um de Dados de Controle), ou seja, que atua na camada "1" do modelo OSI. Mais detalhes sobre esses componentes são apresentados nos itens 4.2.1 e 4.2.2.

A Figura 17 apresenta um diagrama de blocos básico da PADI. Um cabo RJ-11 é conectado do ramal digital do PABX à PADI, por onde os dados entram e são tratados no SCOUT que é o elemento responsável por funções da camada 2. Posteriormente, os dados tratados e provavelmente livres de erros são passados para o uPSD3234 que os formata para o padrão PCPTI e disponibiliza-os na porta serial RS 232 conectada ao PC. Outro cabo RJ-11 é ligado da PADI ao TI, tornando-a transparente para o sistema. As setas simbolizam os fluxos de dados.

Atualmente, uma interface USB está disponível na placa, mas não está sendo usada. Esta será usada em novas versões do *software* como sugere o Capitulo 6.

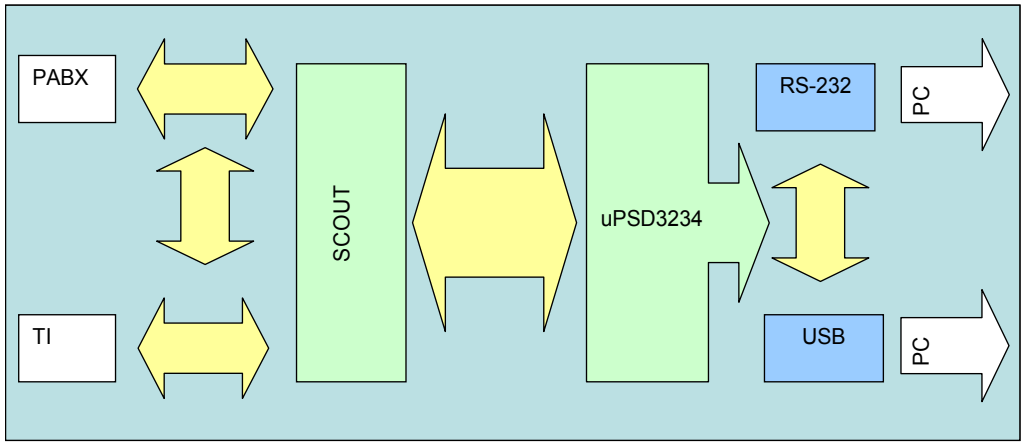

Figura 17 : Diagrama de blocos (básico) da PADI.

4.2.1 Microcontrolador uPSD3234.

O manual do *ST Microelectronics* (2004) afirma que o **uPSD3234** não é um simples microcontrolador, este é um componente da *ST Microelectronics* que encapsula em um único *chip* um microcontrolador que usa os códigos de programação do microcontrolador 8032,

mais uma memória *flash* primária programável de 256K bytes separadas em quatro bancos de 64k bytes cada, com segurança de conteúdo que bloqueia o acesso aos dados programados. Possui também, 8 Kbytes de memória **SRAM** (*Sequential Random Access Memory*) com opção de ser conectada uma bateria de *backup* para que os dados armazenados não sejam perdidos, uma memória *flash* secundária de 32K bytes, uma interface **I^2C** capaz de funcionar como mestre ou escravo, para conexão de periféricos, um conversor Analógico/Digital, seis portas de entrada/saída, ISP (*In-System Programming*) via **JTAG (***Joint Test Action Group***).**

O **uPSD3234** possui ainda uma interface **USB** 1.1 *Slow mode* (1.5 Mbits/s) e duas portas **UART** (*Universal Asynchronous Receiver-Transmitter*). Esse componente pode ser alimentado com uma tensão de 4.5 à 5.5 volts ou de 3.0 à 3.6 volts dependendo da versão do *chip*. A memória *flash* primária é usada como memória de programa, ou seja, onde são executadas as instruções do microprocessador 8032. A memória de dados é usada para guardar os valores das variáveis.

O uPSD3234 possui duas portas seriais padrão conectadas aos pinos P3.0 (RX1), P3.1 (TX1) e P1.2 (RX2), P1.3 (TX2) e trabalham de forma independente uma da outra. Também são portas *full-duplex*, ou seja, podem transmitir e receber dados simultaneamente. Os dados recebidos ou transmitidos podem ser acessados por meio de um registro especial chamado de SBUF (*Serial Buffer*). Neste caso, quando se deseja transmitir, copia-se o byte para o registro e quando se recebe um byte lê-se desse endereço. Embora tenha o mesmo nome, fisicamente, o SBUF são posições de memórias separadas. Além do SBUF o *chip* possui um registro de controle e programação chamados de **SCON** e **SCON2** respectivamente. Nestes registros, são programados diferentes modos de operação da serial, encontram-se também, bits de interrupção e bits que indicam quando um *byte* está pronto para ser lido ou pronto para ser transmitido. A UART pode ser programada para trabalhar em quatro modos diferentes de operação:

- **Modo 0**: neste modo oito bits são transmitidos ou recebidos, o LSB (*Least Significative Bit*), ou seja, o bit menos significativo é o primeiro a ser transmitido/recebido. O *baud rate ou* taxa de transmissão/recepção, é fixada em 1/12 da fOSC (freqüência de operação do uPSD3234).

- **Modo 1**: dez bits são transmitidos ou recebidos, um *start bit* (0), oito bits de dados, sendo o LSB primeiro, e um *stop bit*. O *baud rate* é variável.
- **Modo 2**: este modo se caracteriza pela transmissão/recepção de onze bits, um *start bit* (0), oito bits de dados, sendo o LSB primeiro, um bit usado para indicar a paridade e um *stop bit*. Neste modo, o *baud rate* é programável, podendo ser de 1/32 ou 1/64 da fOSC.
- **Modo 3**: o modo 3 é igual ao modo 2, diferenciando somente o *baud rate*, que neste caso é variável.

Esse componente possui também uma porta USB O SIE (*Serial Interface Engine*), é um módulo da USB responsável pelo protocolo de comunicação de baixo nível e checagem de erros, além disso, detecta a palavra de sincronismo da USB. Possui sete registros de controle o UADR, UCON0, UCON1, UCON2, UISTA, UIEN e USTA e mais três registros de comunicação o UDT0, UDT1, que são registros de transmissão de dados e o UDR0, registro usado para receber dados. Quando um dado é recebido, uma interrupção é gerada e indica que existe um byte que pode ser lido no registrador UDR. Quando se deseja transmitir, copia-se o valor desejado para a posição de memória UDT, neste momento, uma interrupção também é gerada indicando para o microcontrolador que pode transmitir o dado *(ST Microelectronics,* 2004*)*.

4.2.2 SCOUT – DX PSB 21373

De acordo com o manual do componente da *Infineon* (2002) o **SCOUT-DX** integra todas as funções necessárias para uma completa solução de um terminal digital de voz, integrando funcionalidades de *Codec* de áudio com viva-voz e uma interface a dois fios, em um único *chip*. Indicado para implementar um terminal básico de voz com funções de telefone que pode ser ligado a um PABX. O *codec* é responsável pela codificação, decodificação, funções de filtragem e geração de tom de campainha, sinais DTMF (*Dual Tone Multi-Frequency*) e áudio. Possui ainda três entradas e duas saídas analógicas com amplificadores programáveis e uma SCI (*Serial Communication Interface*) que é uma interface serial para conexão com um microcontrolador. O SCOUT-DX é, basicamente, dividido em duas partes: a parte do transdutor e a parte do *codec*.

A parte do transdutor compõe um *transceiver* a dois fios com codificação AMI (*Alternate Mark Inversion*) em canal 2B+D, ou seja, dois canais de voz e um de dados, para *loop* (circuito elétrico fechado) de até 1.8 km de distância. O transdutor, possui também, procedimentos de ativação e desativação automática para estado de *power down* (baixo consumo). Controlador de HDLC, responsável pelo controle e tratamento dos dados dos canais B1, B2 e D (2B+D). Dispõe ainda de um buffer com 64 bytes que se comporta como um fila do tipo FIFO, ou seja, o primeiro a entrar é o primeiro a sair, usado para a transferência dos pacotes de dados. É implementado ainda no transceptor, um monitor de dados e protocolo de canal para o controle de dispositivos conectados ao SCOUT-DX. Nesta parte, que é realizada a **Camada Física** ou camada "1" do modelo OSI.

A parte do *codec* se comporta como um DSP (*Digital Signal Processing*), e é responsável por todo o tratamento de áudio. Compatível com as especificações G.712 da ITU-T (*International Telecommunications Union*) e NET33 da ETSI (*European Telecommunication Standard Institute*). A codificação/decodificação é feita de acordo com PCM (*Pulse Code Modulation*) lei/μ especificado pela G.711 da ITU-T.

O *codec* permite ainda uma configuração flexível de todas as suas funções internas, por meio de registros específicos. Possui três entradas analógicas para a conexão do microfone do viva-voz, do fone de cabeça e do monofone e duas saídas; uma para o auto-falante do viva-voz e outra para ser conectada a cápsula receptora de áudio do monofone. Os níveis de áudio dessas entradas e saídas podem ser programados separadamente (Infineon, 2002).

O SCOUT – DX PSB 21373 tem como responsabilidade a execução de algumas tarefas importantes em um sistema de comunicação, entre estas estão a conversão da estrutura de quadro (*frame*) entre RDSI (Rede Digital de Serviços Integrados) e interface IOM (*ISDN Oriented Modular*), a codificação AMI, a recuperação do relógio de recepção, o sincronismo do relógio IOM-2 com a interface de linha, a ativação e a desativação de processos, disparados por primitivas recebidas do canal IOM C/I (Comando/Indicação) ou por *INFOs* recebidas. Para que não haja colisão entre os pacotes, são criados na inicialização dos equipamentos, quadros e cada quadros têm seus limites. A Figura 18 apresenta os princípios gerais do esquema de comunicação da interface de linha.

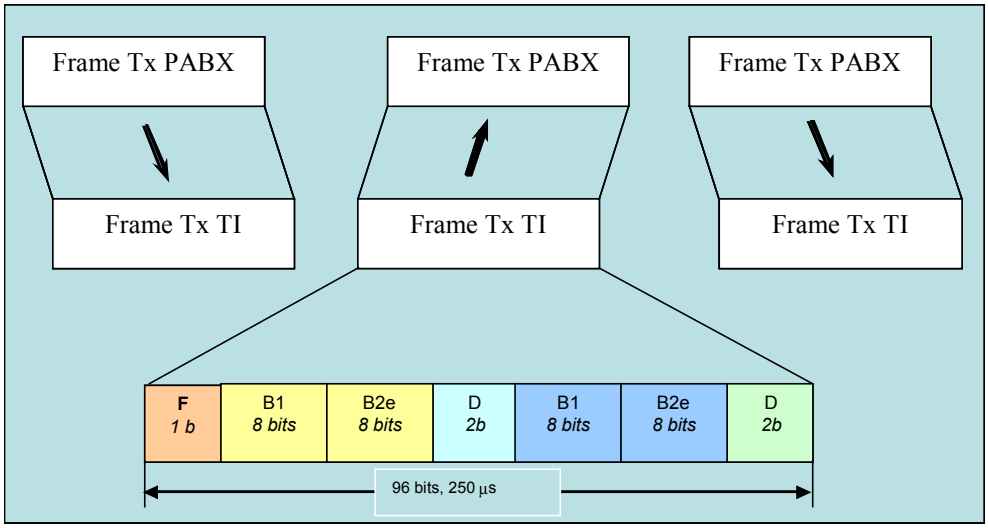

Figura 18 : Estrutura da Interface de Linha.

Fonte: I*nfineon Technologies* (2002, P. 66).

O canal "B" e o canal "D" são embaralhados de acordo com o polinômio da equação 1:

$$X^9 + X^5 + 1 \quad (1)$$

A codificação AMI é usada para interface de linha. A lógica "0" corresponde a um nível neutro na linha, uma lógica "1" é codificada com pulsos positivos e negativos alternados. A codificação AMI é apresentada na Figura 19.

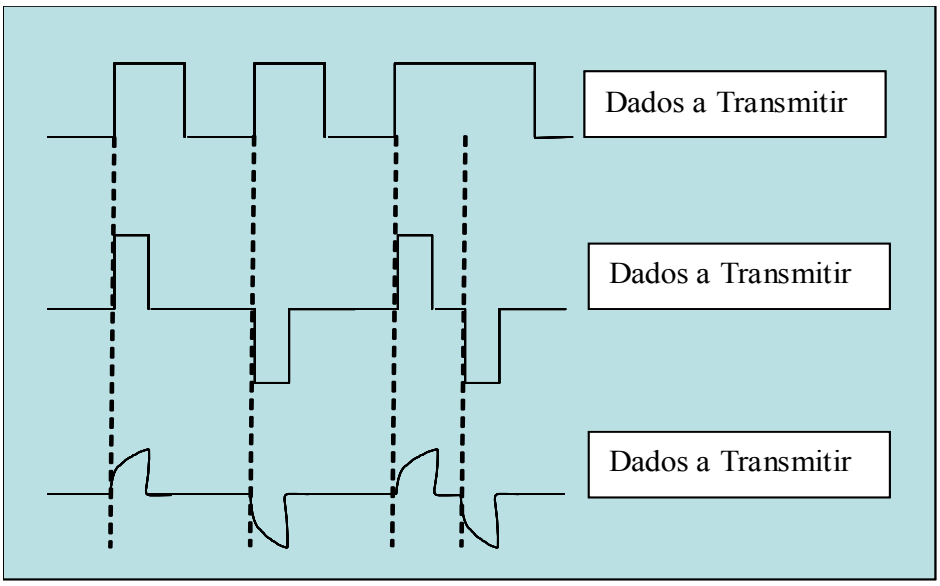

Figura 19 : Codificação AMI.

Cada equipamento tem seu espaço de tempo para transmitirem os dados. A Codificação AMI permite que sejam conectados até oito terminais no mesmo meio físico. Mas no caso do PABX IMPACTA INTELBRAS, é usada somente a conexão ponto-a–ponto.

A sincronização e definições são de responsabilidade do chip *Infíneon.* Depois de sincronizado as partes envolvidas na comunicação, o meio físico já está pronto para receber o tráfego de dados.

4.3 ESTRUTURA DO *SOFTWARE*

Nesta seção, será apresentada a estrutura geral do *software* desenvolvido neste trabalho.

4.3.1 Visão Geral

A Figura 20 ilustra o posicionamento do AP e sistema de aquisição integrados ao ambiente de PABX. Neste caso, a figura representa uma configuração onde o AP está sendo controlado remotamente, por meio da sua função cliente/servidor.

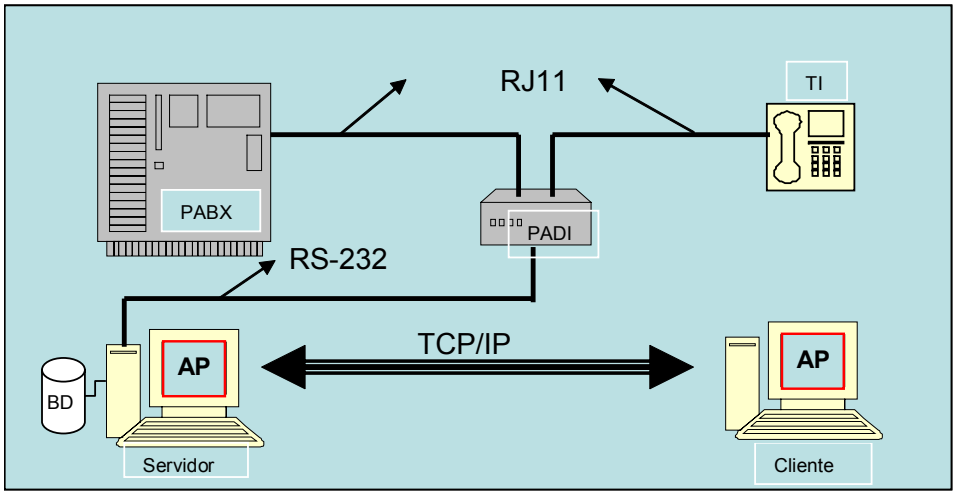

Figura 20 : Arquitetura cliente–servidor do AP.

Todos os dados referentes ao tráfego são armazenados na base de dados localizado no servidor, já os dados referentes à configuração da ferramenta são armazenados localmente. Caso o usuário configure o sistema como cliente, os dados não são mais capturados da porta serial que passam a ser capturados via comunicação com o servidor.

Para que a segurança dos dados seja mantida e nenhuma pessoa sem autorização tenha acesso a esses dados, na conexão do cliente com o servidor é solicitado ao usuário que entre com uma senha configurada no servidor e só é permitida a conexão de um cliente ao servidor por vez. Toda a comunicação entre cliente e servidor é com base em um protocolo simples apresentado no item 4.3.2.

De forma geral, o sistema em modo servidor captura os dados e mostra na tela, caso esteja conectado com um cliente este continua mostrando os dados na tela e "ecoa" esses dados para o cliente que recebe e mostra em sua tela disponibilizando os dados para análise.

A estrutura básica do *software* é apresentada na Figura 21 na qual consiste numa estrutura em camadas. Como já comentado anteriormente, o *software* possui uma interface cliente/servidor para que se possa acessar remotamente os dados da comunicação entre PABX e TI. O programa, tanto cliente como servidor, é o mesmo, bastando que o usuário configure como este deseja que o sistema opere. Caso o usuário escolha que o *software* trabalhe como servidor, os dados são capturados da porta serial. Caso a opção seja cliente os dados são recebidos via TCP/IP.

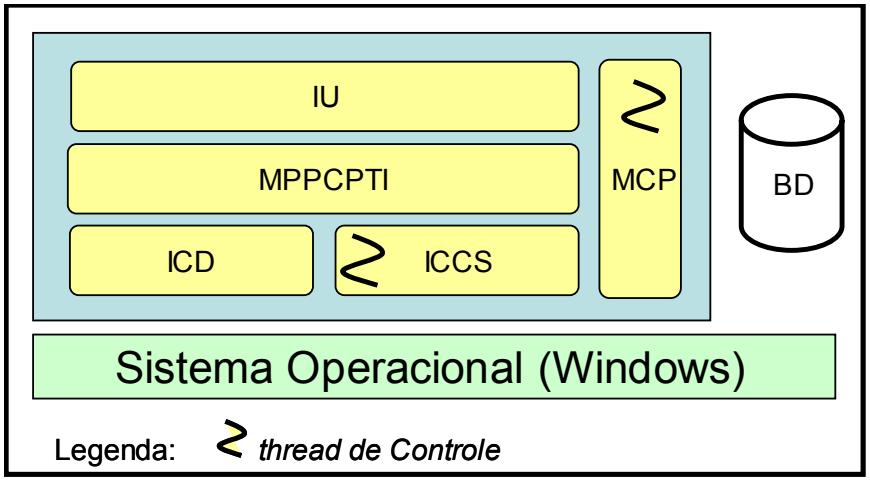

Figura 21 : Estrutura do *Software* analisador de protocolo.

A estrutura do *software* do AP possui os seguintes módulos:

MCP: Módulo de Controle Principal, é o controle principal do programa, responsável por gerenciar todo o *software* do AP

. Foi criada uma *thread* para essa parte do programa.

ICCS: Interface de comunicação Cliente-Servidor, também é uma *thread* implementada em forma de máquina de estados que quando está executando como servidor fica esperando que o cliente faça contato para inicializar a comunicação. Quando o programa estiver no papel de cliente, este inicia o contato com o servidor para que eles se comuniquem.

ICD: Interface Controle de Dados é a parte do programa responsável pelo acesso ao meio externo (interface Serial). Os dados são armazenados em um *buffer* de tamanho fixo, que é acessado a cada 2ms (0,002 segundo) pelo MCP. Para que não haja perda de dados o tamanho do *buffer* foi dimensionado segundo a equação 2. O MCP lê byte por byte do buffer até encontrar a palavra fim de linha representada pelos caracteres"\n\r" e então é montada a mensagem e passada para a camada superior (MPPCPTI).

$$N = p \cdot N = p \cdot \frac{p}{(1-p)} = \frac{p^2}{(1-p)} = \frac{\left(\frac{\lambda}{\mu}\right)^2}{1-\frac{\lambda}{\mu}} \qquad (2)$$

Fonte: Tanenbaum (2003, p. 746).

Sendo:

N: número de elementos no *buffer,* ou seja, o tamanho mínimo do *buffer*.

λ : número médio de bytes por segundo que chegam no *buffer*.

μ : número médio de bytes que são retirados do *buffer* por segundo.

Por exemplo, com uma taxa de chegada de λ = 4775 bytes por segundo ou 38200bps que é o usado atualmente, e uma taxa de consumo da fila de μ =50 bytes por segundo, tem-se:

N= [(4775/50)^2] / [1 – (4775/50)] = 9120,25 / 94,5 ≅ 97 bytes.

Em média, sempre terá 97 bytes no *buffer*, portanto este é o tamanho mínimo que o *buffer* tem que ter. No trabalho foi usado um *buffer* com o dobro deste tamanho, ou seja, 194 bytes.

IU: Interface Usuário é responsável pela interface do programa e o usuário do sistema, ou seja, são todas as telas e comandos de entrada de dados do programa;

BD: Base de Dados é onde são armazenados os dados para posterior consulta e geração de relatórios. Toda a base de dados é armazenada no servidor;

MPPCPTI: Módulo de Processamento do Protocolo de Comunicação PABX TI INTELBRAS é implementada em forma de máquina de estados e é responsável pelo tratamento e gerenciamento do protocolo PCPTI, trata as PDUs e repassa para a SAP correspondente. Todas as SAPs, também são implementados em forma de máquinas de estados com chamada a funções de tratamento de mensagens, com base no protocolo PCPTI. Essa parte não pode ser apresentada no trabalho de uma forma mais detalhada, devido ao termo de proteção de propriedade intelectual da INTELBRAS, o que protege os detalhes técnicos das tecnologias envolvidas.

A Figura 22 apresenta a estrutura do MPPCPTI e seus campos de dados.

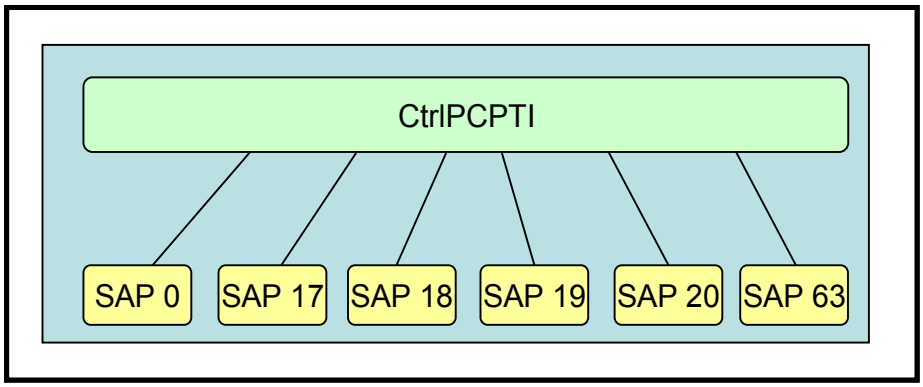

Figura 22 : Estrutura do MPPCPTI.

O MPPCPTI possui os seguintes campos:

CtrlPCPTI:. Controle do PCPTI analisa a mensagem que neste chega e repassa para a SAP correspondente aquele tipo de mensagem;

SAP0:. em particular o SAP0 é responsável pelo processo de controle e estabelecimento da chamada;

SAP17: responsável por tratar mensagens de estados de linha e ramais do PABX, também usado para transmitir relógio em tempo real com data e hora;

SAP18: todo o processo de inicialização da comunicação TI com o PABX é feito por essa parte do programa;

SAP19: essa parte do programa é responsável pelo tratamento de mensagens de programação de facilidades do PABX;

SAP20: parte do programa, responsável pelo tratamento de mensagens referente a transmissão de dados entre as TIs;

SAP63: essa parte do programa trata as mensagens de manutenção da camada 2.

O *software* funciona da seguinte forma, após feitas as conexões necessárias o usuário executa o programa por meio de um ícone que é apresentado na área de trabalho do PC, o programa do AP é carregado, mas todas as funções ficam bloqueadas, então o usuário faz a escolha do modo de operação da ferramenta: ou cliente ou servidor. Após ter feito esta escolha, o sistema solicita que o usuário entre com uma senha de acesso, caso a senha esteja correta as funções disponíveis para aquele módulo escolhido, são liberadas. A partir deste momento, a ferramenta está liberada para uso.

Na próxima seção, será apresentado o protocolo usado para a comunicação cliente/servidor do AP.

4.3.2 Protocolo Cliente-Servidor

Como apresenta a Figura 23, o *software* usa um protocolo especialmente criado para esta comunicação Cliente-Servidor chamado PCS (Protocolo Cliente-Servidor). As trocas de mensagens entre cliente e servidor são feitas com utilização de soquetes de fluxo (TCP). O servidor espera uma tentativa de conexão do cliente. Quando um aplicativo cliente se conecta ao servidor, o aplicativo servidor envia uma mensagem para o cliente indicando que a conexão foi bem sucedida e o cliente exibe ao usuário a mensagem de conectado. O cliente pode configurar a porta e o IP antes de iniciar a conexão com o servidor. No lado do servidor, o usuário pode configurar a porta que o servidor irá disponibilizar para conexão. A configuração da porta a ser usada na conexão deve ser feita com números acima de 1024, pois muitos sistemas operacionais reservam números de porta abaixo desse valor para serviços de sistema.

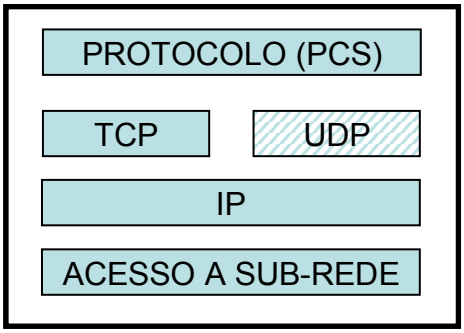

Figura 23 : Camadas de comunicação cliente–servidor do AP.

A Figura 24 apresenta a PDU do protocolo PCS.

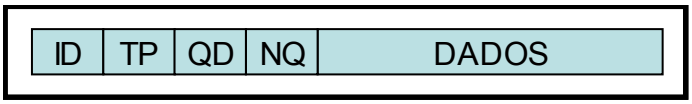

Figura 24 : PDU do PCS.

Todos os campos possuem tamanho de 1 byte, com exceção do campo de dados que possui o limite máximo de 250 bytes.

ID: identificação do quadro, essa identificação é criada na inicialização, de uma forma aleatória, pelo servidor e repassada para o cliente. Seu valor pode ser de 00h até F0h;

TP: tipo da mensagem, que pode ser:

⇒ Tipos de mensagens enviadas do Servidor para o Cliente:

- E0h: inicialização. Utilizado na inicialização da comunicação para sincronizar as mensagens;
- E1h: resposta de validação da senha;
- E2h: PDUs capturados pelo PADI, usado para o servidor enviar para o cliente as PDUs que foram capturados da linha por meio da PADI;
- E3h: envio de relatório. Será enviada uma PDU por vez.

⇒ Tipos de mensagens enviados do Cliente para o Servidor:

- D0h: envio de senha;
- D1h: pedido de PDUs capturadas pela PADI;
- D2h: pedido de encerramento do envio de PDUs do PADI;
- D3h: pedido de relatório;
- D4h: pedido de encerramento do envio do relatório;
- D5h: recebeu o dado correto, pode mandar próximo quadro;
- D6h: retransmissão, o quadro recebido apresenta inconformidade;
- D7h: pedido para a gravação em arquivo das PDUs capturadas pelo PADI;
- D8h: alteração de senha de acesso;
- D9h: mensagem inválida;
- DAh: fim da conexão.

QD: quantidade de byte de dados.

NQ: número do quadro para que se siga uma seqüência, caso haja uma quebra na seqüência o dado é retransmitido. Quando chega à FFh é iniciado para 00.

Para todo quadro transmitido pelo servidor é aguardada uma resposta de confirmação, por parte do cliente, caso isso não ocorra, dentro de 2 segundos, o quadro é retransmitido.

4.3.3 Conexão ao servidor (Inicialização da comunicação cliente-servidor)

Para exemplificar essa etapa foram elaborados diagramas de estados. O diagrama de estados do início de conexão da parte servidor se encontra na Figura 65 (**apêndice A)** e o diagrama de estados de início de conexão do cliente na Figura 66 (**apêndice A.)** A seqüência normal de conexão cliente-servidor ocorre de acordo com os seguintes passos:

- a parte do servidor sempre deve ser iniciada primeiro;
- o servidor fica monitorando se algum cliente solicita uma conexão;

- o cliente envia uma mensagem de solicitação de conexão pela porta TCP e para o endereço IP pré-programado;
- o servidor recebendo esta solicitação de conexão do cliente envia uma mensagem permitindo a conexão e fica aguardando o envio da senha por parte do cliente;
- o cliente recebe essa confirmação de conexão do servidor, pede ao usuário que entre com uma senha, a qual dará permissão ao cliente para que este possa solicitar dados ao servidor. Enquanto essa senha não for validada pelo servidor, não é permitido ao cliente efetuar nenhuma tarefa que dependa deste servidor, ou seja, a conexão só é realmente efetivada após a validação da senha de acesso. Após 60 segundos sem receber a senha, a conexão é fechada;
- o usuário entra com a senha no lado cliente;
- o cliente envia a senha ao servidor que autentica o usuário e a senha;
- o servidor envia ao cliente a resposta da autenticação. Se a senha estiver correta a conexão é estabelecida. Caso contrário, após cinco tentativas cuja autenticação falhe, o servidor fecha a conexão e envia uma desconexão ao cliente que mostra ao usuário uma mensagem que o sistema irá ser finalizado por excesso de tentativa de senha. Caso o usuário queira tentar novamente, tem que iniciar novamente o programa no modo cliente e solicitar uma nova conexão ao servidor.

A partir deste momento, o servidor fica aguardando solicitações do cliente.

4.3.4 Autenticação por senha

Toda vez que se deseja usar a ferramenta no modo cliente o usuário deve entrar com uma senha que é solicitada na inicialização do programa. A senha é armazenada no servidor que inicialmente tem um valor *default* de “Senha123” que o usuário pode mudar a qualquer momento. Neste protótipo, as senhas não estão em modo criptografado e não se teve preocupação quando a segurança no envio da senha, ficando para outra fase do projeto, pois este vai ser continuado pela INTELBRAS.

4.3.5 Transmissão dos dados capturados pela PADI

Quando o cliente quiser mostrar os dados capturados pela PADI, este envia uma mensagem do tipo D1h ao servidor que ao receber esta mensagem inicia a transmissão das PDUs. O cliente neste momento está aguardando o envio da PDU por parte do servidor. Caso a mensagem esteja correta, segundo a verificação do cliente com base na seqüência dos quadros, é enviada ao servidor uma mensagem do tipo D5h, caso contrário, ou seja, se houver inconsistência na mensagem, o cliente envia a mensagem do tipo D6h pedindo a retransmissão da última PDU enviada pelo servidor. Esse fluxo de mensagens ocorre até que o cliente envie a mensagem do tipo D2h pedindo o fim do envio das PDUs.

4.3.6 Transmissão dos dados contidos em arquivo

A transmissão dos dados que estão armazenados em arquivos, é semelhante à transmissão das PDUs capturadas pela PADI, mudando apenas o tipo de mensagens.

Quando o cliente desejar ter acesso aos relatórios contidos em arquivos armazenados no servidor, este envia uma mensagem do tipo D3h com o campo de dados preenchido com a data e hora inicial seguidos da data e hora final do relatório. O servidor ao receber esta mensagem inicia a transmissão das PDUs, o cliente neste momento está aguardando o envio da PDU por parte do servidor. Caso a PDU esteja correta, segundo a verificação do cliente com base na seqüência dos quadros, é enviada ao servidor uma mensagem do tipo D5h, para que este envie a próxima PDU, caso contrário, ou seja, se houver inconsistência na mensagem, o cliente envia a mensagem do tipo D6h pedindo a retransmissão da última PDU enviada pelo servidor. Esse fluxo de mensagens pode se encerrar por três motivos: se forem enviadas todas as PDUs contidas no intervalo de tempo solicitado, ou a última PDU do relatório foi enviada, ou ainda se o cliente enviar a mensagem do tipo D4h pedindo o fim do envio do relatório.

Os diagramas de estados que mostram os fluxo de mensagem apresentados nos itens 4.3.5 e 4.3.6 se encontra no apêndice. O diagrama de estados da parte do servidor conectado encontra-se na Figura 67 (**apêndice A)** e o diagrama de estados do cliente conectado encontra-se na Figura 68 do (**apêndice A).**

5 IMPLEMENTAÇÃO DO PROTÓTIPO DO *SOFTWARE* ANALISADOR DE PROTOCOLOS.

Nesta seção, será apresentado o protótipo do *software* que foi desenvolvido.

5.1 SISTEMA PROPOSTO

5.1.1 Visão Geral

Na Figura 25 a seguir é apresentado um fluxo de tarefas que mostra como funciona o AP.

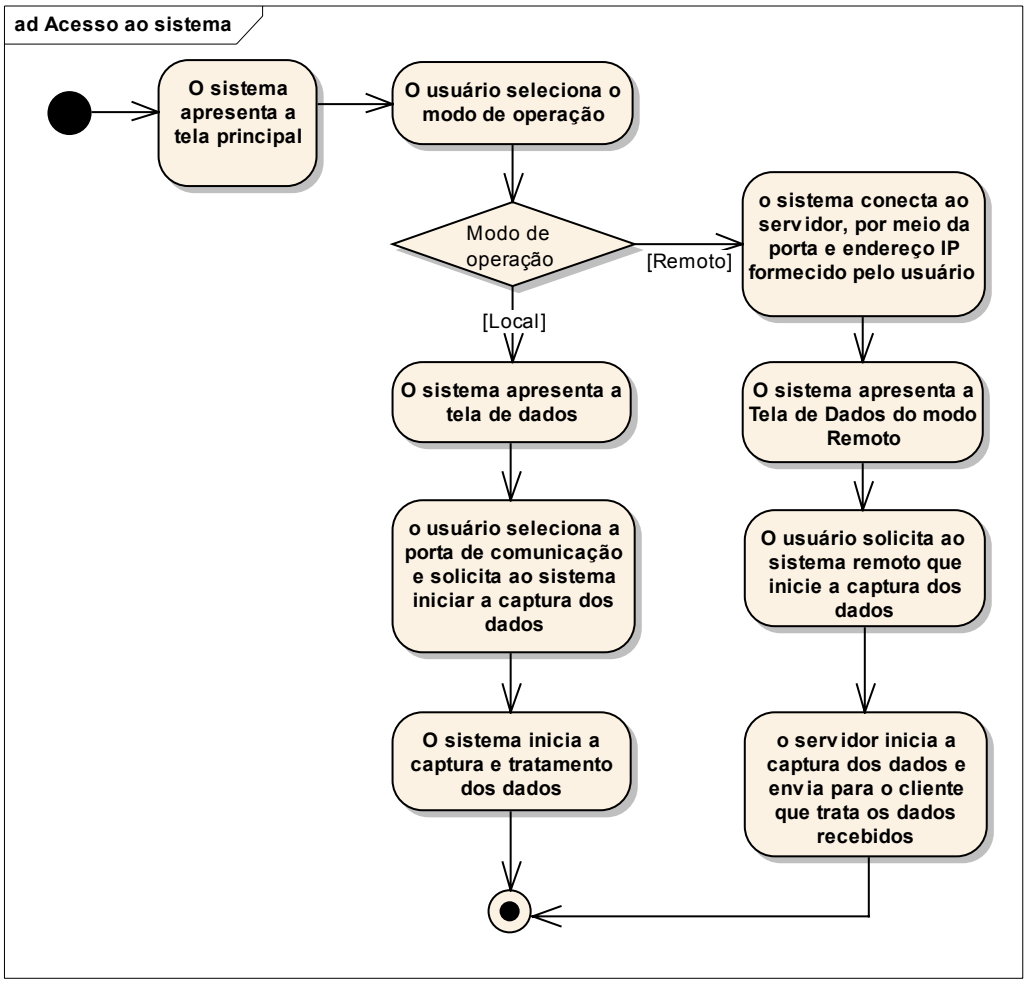

Figura 25 : Fluxograma do funcionamento básico do AP.

5.2 REQUISITOS FUNCIONAIS

REF-001 O sistema deve permitir que um usuário possa efetuar o login no sistema.

REF-002 O sistema deve permitir o acesso remoto a um servidor que captura os dados.

REF-003 O sistema deve permitir o acesso as portas seriais para a captura dos dados.

REF-004 O sistema deve ser capaz de analisar as PDUs recebidas, tratá-las segundo o protocolo PCPTI e apresentá-las na tela de dados para o usuário.

REF-005 O sistema deve permitir que o usuário possa consultar os *logs* de dados capturados.

REF-006 O sistema deve permitir que o usuário configure qual tipo de mensagens ele deseja visualizar (Filtros de mensagens).

REF-007 O sistema deve permitir que o usuário possa consultar remotamente(no Cliente) e em tempo real, os logs de dados capturados pelo servidor.

REF-008 O sistema deve permitir a consulta remotamente das informações dos dados armazenados em arquivo.

REF-009 O sistema deve permitir a alteração da senha de acesso remotamente.

REF-010 O sistema deve permitir o registro dos dados capturados em um arquivo (log).

REF-011 O sistema deve permitir a consulta das informações dos dados armazenados em arquivo.

REF-012 O sistema deve permitir o registro das configurações em um arquivo.

REF-013 O sistema deve permitir que as configurações armazenadas em arquivo sejam carregadas.

REF-014 O sistema deve permitir que o usuário configure as portas seriais.

REF-015 O sistema deve permitir que o usuário configure as portas TCP/IP e endereço IP que o sistema usará para se comunicar entre cliente e servidor.

REF-016 O sistema deve permitir a alteração da senha de acesso tanto no cliente como no servidor.

REF-017 O sistema deve apresentar um editor para que o usuário possa editar (recortar, colar, copiar, apagar) os logs de mensagens recebidos ou qualquer outro arquivo de formato texto (.txt). Deve permitir ainda carregar e gravar os dados em arquivos.

5.3 REQUISITOS NÃO FUNCIONAIS

5.3.1 Segurança

A seguir na Figura 26 são apresentados os requisitos de segurança que o AP possui.

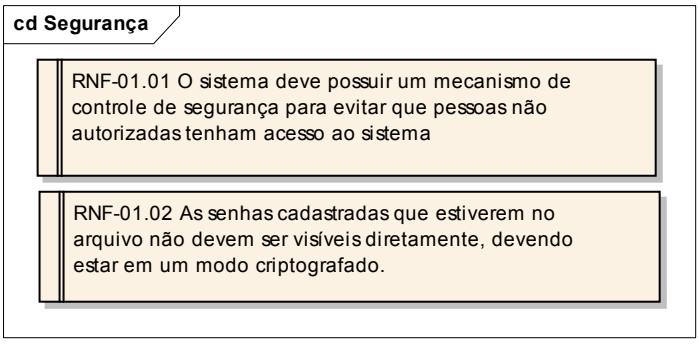

Figura 26 : Requisitos de Segurança.

Foi implementado um mecanismo de controle de acesso com senha para evitar que pessoas sem autorização tenham acesso aos dados referente a comunicação entre cliente e servidor.

As senhas cadastradas serão gravadas em um arquivo de configurações na máquina onde se executa o servidor do AP. Neste protótipo, as senhas não precisão estar em modo criptografado.

5.3.2 Usabilidade

Como é apresentado na Figura 27, o AP apresenta os seguintes requisitos de Usabilidade.

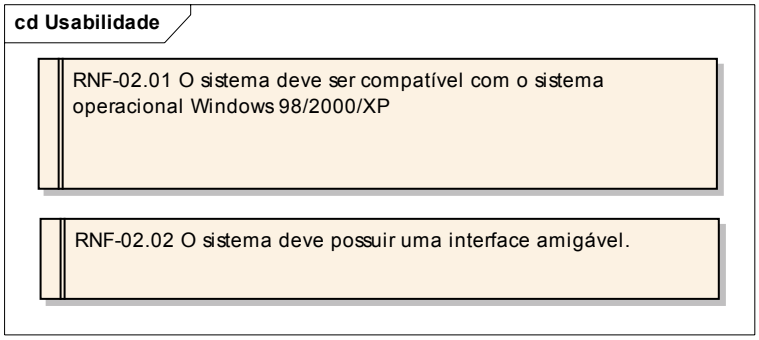

Figura 27 : Requisitos de Usabilidade

O AP foi desenvolvido para ser compatível com sistema operacional Windows 98/2000/XP, não devendo executar em Windows 95 por não ter biblioteca de arquivos compatíveis. Também não pode ser executado em sistema operacional *Linux***.**

O AP foi desenvolvido levando-se em conta algumas regras ergonômicas, fazendo com que o *software* ficasse fácil de utilizar e com uma interface amigável, sem a necessidade de se fazer curso preparatório para a operação e utilização do sistema.

5.3.3 Confiabilidade

A Figura 28 a seguir apresenta o requisito de confiabilidade do AP.

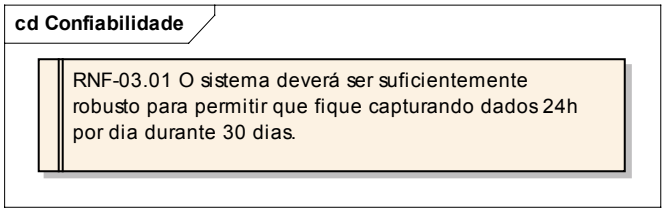

Figura 28 : Requisito de Confiabilidade

O AP pode ficar capturando dados 24h horas por dia, durante trinta dias sem interrupção, enquanto a PADI estiver enviando os dados e enquanto o AP estiver sendo executado. A única limitação é o limite em que os dados são apresentados, não sendo possível apresentar mais do que 32.768 PDUs devido a uma limitação do componente da *Borland* usado na aplicação. Neste caso, quando este número for atingido a PDU mais antiga é eliminada e é apresentada a PDU mais recente, funcionado como um *buffer* circular. A solução para isso é

habilitar o log das PDUs, que será armazenado em disco, pois neste caso o limite de armazenamento é definido pela capacidade do disco onde estão sendo armazenadas as PDUs.

5.3.4 Desempenho

O requisito de desempenho é apresentado na Figura 29.

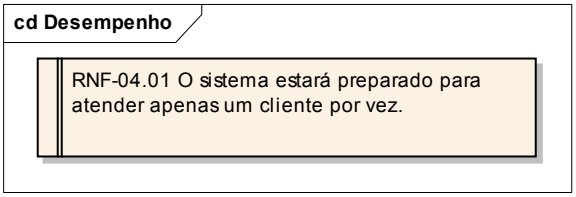

Figura 29 : Requisito de Desempenho

O AP permite que apenas um cliente se conecte ao servidor, de cada vez, pois nessa primeira versão o servidor não está preparado para suportar múltiplas conexões. Mas, para futuras versões, pretende-se implementar esta funcionalidade.

5.3.5 *Software* e *Hardware*

Na Figura 30 são apresentados os requisitos de *Software* e *hardware* do AP.

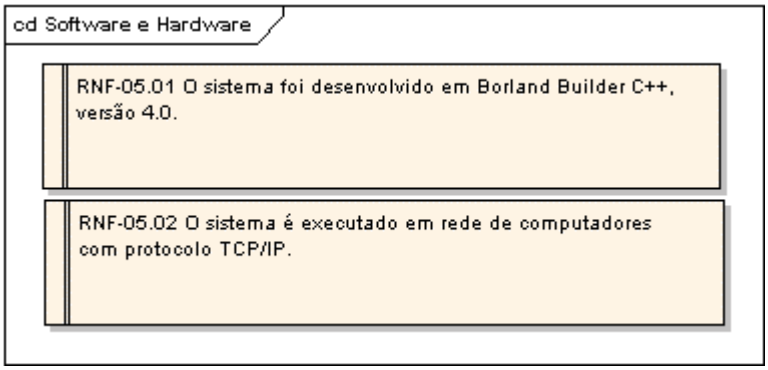

Figura 30 : Requisitos de *Software* e *Hardware*

O AP foi desenvolvido em *Borland Builder* C++, versão 4.0. usando componentes da própria *Borland* para fazer a manipulação de arquivos e para desenvolver a comunicação entre cliente e servidor.

O sistema é executado em rede de computadores com protocolo TCP/IP. Neste caso, escolheu-se o TCP como protocolo de transporte, por este ser um protocolo mais confiável do que o UDP. Em contra partida, o TCP gera mais *overhead* do que o UDP, devido ao estabelecimento e encerramento de conexões, o que não influencia neste projeto devido ao modo de operação sempre conectado entre cliente e servidor, não exigindo freqüentes estabelecimentos/encerramentos de conexões entre cliente e servidor.

5.4 REGRAS DE NEGÓCIO

RNE-001 A senha deve ter tamanho mínimo de 6 caracteres e tamanho máximo de 14.

RNE-002 Os arquivos com PDUs a serem carregadas terão que existir, caso contrário deve existir um alerta para o usuário.

RNE-003 Só é possível uma solicitação de serviço remoto por vez.

RNE-004 Deve existir um arquivo de configurações. Caso não exista, este arquivo é criado no momento da próxima gravação das configurações solicitada pelo usuário.

RNE-005 Caso já existir um arquivo de PDU e o cliente solicitar uma gravação remotamente neste arquivo, os dados gravados anteriormente são perdidos (vale sempre a última gravação).

RNE-006 As datas das PDUs recebidas no cliente são a mesma data do servidor.

RNE-007 O número máximo de PDUs recebidas é de 32.768, assim que este número for alcançado, é apagada a PDU mais antiga e acrescentada a mais atual.

RNE-008 o número máximo que o campo de dados na PDU pode ter é de 100 bytes.

RNE-009 Só é possível uma conexão de cliente ao servidor de cada vez.

RNE-010 Se já existir um arquivo de configuração e o usuário gravar a nova configuração, os dados gravados anteriormente são perdidos (vale sempre a última configuração gravada).

5.5 CASOS DE USO

Nesta seção, são apresentados os casos de uso, ou seja, a seqüência de passos que devem ser seguido pelo *software* para realizar uma determinada tarefa. São apresentados, em cada caso de uso, quais requisitos aquele determinado caso de uso atende, as restrições, o cenário e no final o diagrama de robustez que mostra como os componentes, classes e usuário interagem. No **apêndice B,** são apresentados os diagramas de seqüência que mostram o fluxo de mensagens entre os objetos de acordo como foi descrito nos casos de uso.

Com a finalidade de melhorar a organização e facilitar o entendimento dos casos de uso, estes foram divididos em módulos como é apresentado na Figura 31 a seguir.

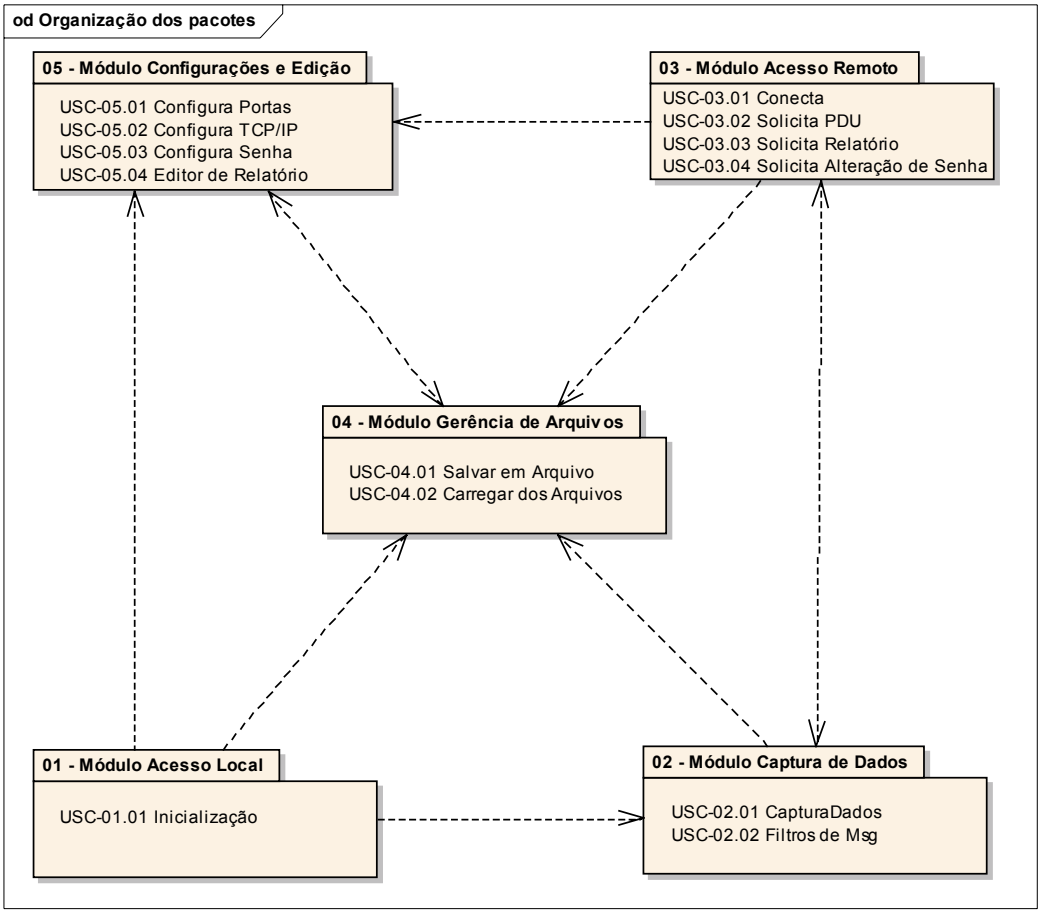

Figura 31 : Visão geral dos módulos de casos de uso

5.5.1 UC - 01.01 Inicialização

Seqüências de passos da inicialização do sistema

Requisito:

- REF-001 O sistema deve permitir que um usuário possa efetuar o *login* no sistema.

Restrições:

- Pré-condição. O sistema deve estar instalado em um micro computador.
- Pós-condição. O sistema estará pronto para que o usuário use suas funções.

Cenário:

A. Iniciando Modo Local {Caminho Principal}.

A.1 Executar o programa APTI.

A.2 O sistema operacional executa o programa APTI.

A.3 O APTI apresenta a tela inicial FORM_PRINCIPAL.

A.4. O usuário faz a escolha do modo de operação da ferramenta: ou cliente (Remoto) ou servidor(Local). Caso escolha modo local, segue a seqüência (A.5), caso contrário vai para B.

A.5 No modo Local (Servidor), o APTI apresenta a tela de dados FORM_TELADADOS.

A.6 Vai para o item C.

B. Iniciando Modo Remoto {Caminho Alternativo}.

B.1 O sistema apresenta a tela FORM_IP solicitando que o usuário entre com o endereço IP do servidor.

B.2 O APTI conecta ao servidor por meio da porta pré-programada e ao endereço IP que o cliente entrou.

B.3 O sistema apresenta a tela FORM_SENHA solicitando que o usuário entre com uma senha de acesso.

B.4 O usuário entra com a senha de acesso.

B.5 O APTI cliente envia a senha para o APTI servidor.

B.6 O APTI servidor valida a senha e responde ao APTI Cliente.

B.7 O APTI apresenta a tela de dados FORM_TELADADOS.

B.8 vai para o item C.

C. A partir deste momento a ferramenta está liberada para uso.

O diagrama de robustez da inicialização é apresentado na Figura 32 a seguir, e o diagrama de seqüência encontra-se na Figura 69 (**apêndice B).**

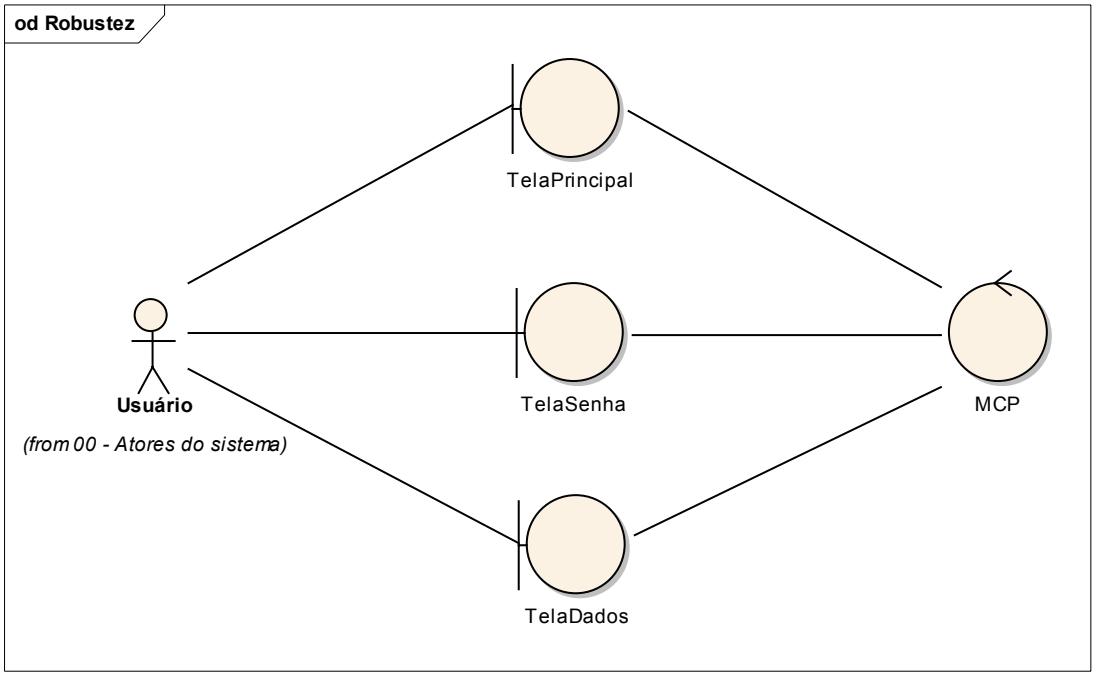

Figura 32 : Diagrama de Robustez – Inicialização.

5.5.2 USC-02.01 Captura dos Dados

Seqüência de passos referentes a captura de dados da porta serial.

Requisitos:

- REF-003 O sistema deve permitir o acesso as portas seriais para a captura dos dados.
- REF-004 O sistema deve ser capaz de analisar as PDUs recebidas, tratá-las segundo o protocolo PCPTI e apresentá-las na tela de dados para o usuário.
- REF-005 O sistema deve permitir que o usuário possa consultar os *logs* de dados capturados.

Restrições:

Pré-condição: O APTI deve estar iniciado em modo local.

Pré-condição: A tela de dados FORM_TELADADOS deve estar sendo apresentada ao usuário.

Cenário:

A. Inicia a Captura de Dados {Caminho Principal}.

A.1 O usuário seleciona a porta onde está conectada a PADI.

A.2 O usuário pressiona o botão de início de captura dos dados. O usuário também pode escolher em ir para o passo B ou C.

A.3 O sistema abre a porta selecionada.

A.4 O sistema armazena em um *buffer* os dados que chegam na porta selecionada.

A.5 O sistema analisa o *buffer* em tempos em tempos e vai montando a PDU até achar o fim de quadro.

A.6 O sistema repassa a PDU para o MPPCPTI.

A.7 O MPPCPTI analisa o tipo da PDU e direciona para a SAP correspondente.

A.8 A SAP responsável analisa a PDU e mostra na tela de dados.

B. Pausa a aquisição dos dados {Caminho Alternativo}.

B.1 O usuário pressiona o botão de PAUSA de aquisição de dados.

B.2 O sistema não lê mais os dados que chegam na porta selecionada.

B.3 Retorna ao passo A.2.

C. Pára a aquisição de dados {Caminho Alternativo}.

C.1 O usuário pressiona o botão de PARADA de aquisição de dados.

C.2 O sistema fecha a porta selecionada.

C.3 Retorna ao passo A.2.

A seguir na Figura 33 é apresentado o diagrama de robustez da captura de dados e o diagrama de seqüência é apresentado na Figura 70 (**apêndice B).**

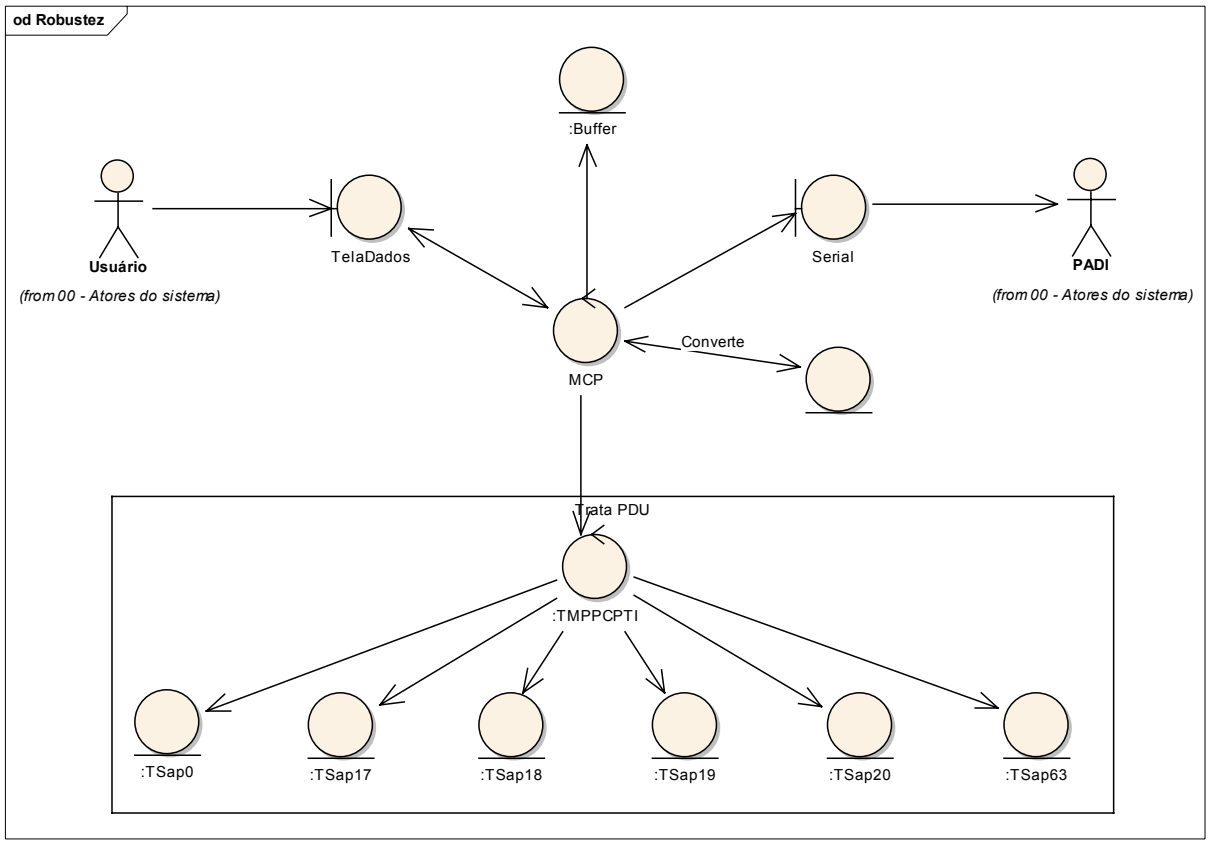

Figura 33 : Diagrama de Robustez - Captura dos dados.

5.5.3 USC-02.02 Filtros de Mensagens

Requisitos:

- REF-006 O sistema deve permitir que o usuário configure qual tipo de mensagens ele deseja visualizar (Filtros de mensagens).

Cenário:

A. Filtro de PDU {Caminho Principal}.

A.1 Na tela de dados o usuário pressiona o botão Filtros.

A.2 O sistema apresenta a tela de filtros.

A.3 O operador seleciona as PDUs que ele deseja que seja mostrada e confirma.

A.4 O sistema grava estas configurações.

A.5 O sistema apresenta as PDUs na tela de dados.

Na Figura 34 é apresentado o diagrama de robustez dos filtros de mensagens e o diagrama de seqüência encontra-se no **apêndice B** na Figura 71**.**

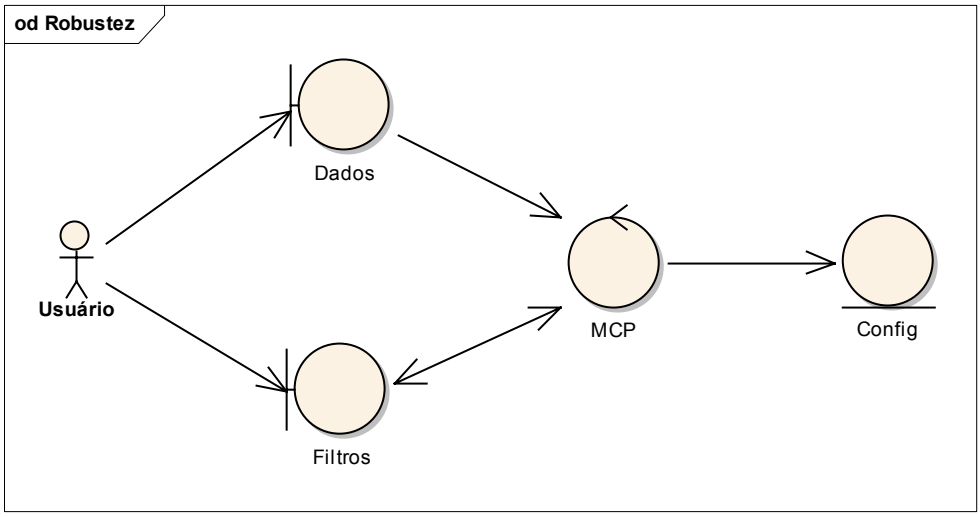

Figura 34 : Diagrama de Robustez - Filtros de mensagens

5.5.4 USC-03.01 Conecta

Seqüências de passos referentes ao acesso remoto dos dados e funcionalidades do sistema.

Requisito:

- REF-002 O sistema deve permitir o acesso remoto a um servidor que captura os dados.

Restrições:

- Pré-condição. 1. É necessário que se já tenha executado os passos do UCInicialização.
- Pré-condição. 2. A parte do servidor sempre deve ser iniciada primeiro.

Cenário:

A. Inicio da Conexão {Caminho Principal}.

A.1O servidor fica monitorando se algum cliente solicita uma conexão.

A.2 O cliente apresenta uma tela FORM_IP, solicitando o endereço IP do Servidor.

A.3 O usuário preenche o endereço do IP.

A.4 O cliente envia uma mensagem de solicitação de conexão pela porta TCP pré-programada e para o endereço IP solicitado.

A.5 O servidor recebendo esta solicitação de conexão do cliente envia uma mensagem permitindo a conexão.

A.6 O servidor fica aguardando o envio da senha por parte do cliente.

A.7 O cliente recebe confirmação de conexão do servidor.

A.8 O cliente apresenta uma tela FORM_SENHA solicitando a senha de acesso.

A.9 O usuário preenche a senha e confirma.

A.10 O cliente envia a senha ao servidor que autentica essa senha.

A.11 O servidor valida a senha fornecida.

A.12 O servidor envia ao cliente a resposta da autenticação.

A.13 A conexão cliente servidor está estabelecida.

A.14 O sistema apresenta a tela de dados FORM_TELADADOS.

A.15 Fecha Conexão.

B. Não conseguiu conectar {Caminho Alternativo}.

B.1 Se no passo A.5, o cliente não conseguiu se conectar a porta TCP e ao IP solicitado.

B.2 vai ao passo A.15.

C. Senha Incorreta {Caminho Alternativo}.

C.1 Se no passo A.11, a senha não puder ser validada, o servidor envia ao cliente que a senha é incorreta.

C.2 O cliente apresenta ao usuário uma mensagem "Senha incorreta!".

C.3 Volta ao passo A.8.

C.4 Após cinco tentativas vai para o passo A.15.

O diagrama de robustez referente a conexão é apresentado na Figura 35 e o diagrama de seqüência pode ser visto na Figura 72 do **apêndice B.**

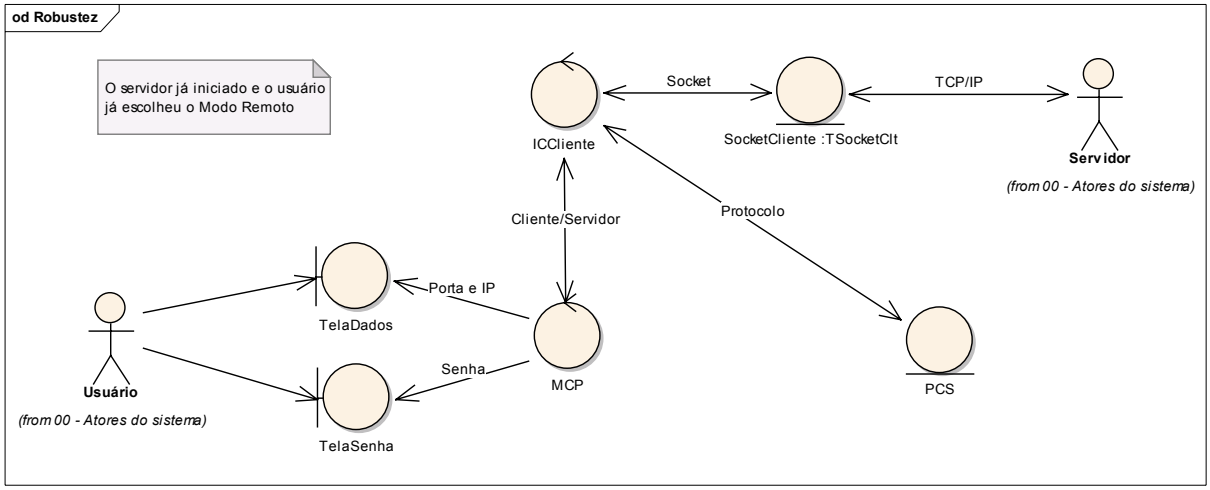

Figura 35 : Diagrama de Robustez – Conecta.

5.5.5 USC-03.02 Solicita PDU

Requisitos:

- REF-007 O sistema deve permitir que o usuário possa consultar remotamente (no Cliente) e em tempo real, os logs de dados capturados pelo servidor.

Cenário:

A. Pede PDU {Caminho Principal}.

A.1 O cliente pede a primeira PDU.

A.2 O servidor recebe o Pedido.

A.3 O servidor envia a PDU.

A.4 O cliente fica aguardando o envio da PDU por parte do servidor.

A.5 O cliente recebe a PDU.

A.6 Mensagem correta, cliente pede próxima PDU.

A.7 Volta para passo A.2.

A.8 Cliente pede o fim do envio das PDUs.

B. PDU Incorreta {Caminho Alternativo}.

B.1 Caso no passo A.6 a mensagem esteja incorreta, cliente pede retransmissão da última PDU enviada pelo servidor.

B.2 Vai para o passo A.7.

B.3 Após cinco tentativas vai para o passo A.8.

Na Figura 36 a seguir, encontra-se o diagrama de robustez referente a solicitação de PDU, no **Apêndice B** na Figura 73 é apresentado o diagrama de seqüência desta funcionalidade.

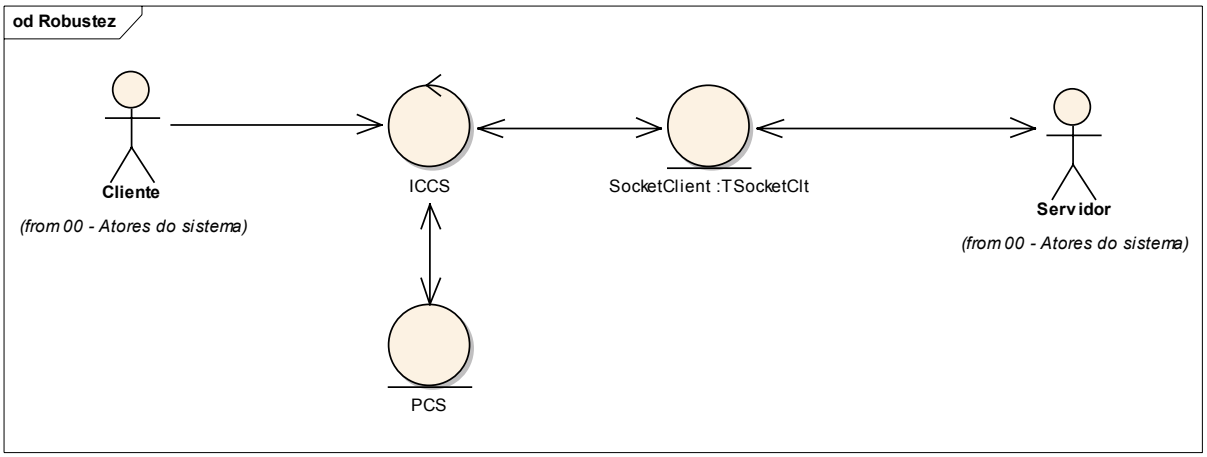

Figura 36 : Diagrama de Robustez - Solicita PDU.

5.5.6 USC-03.03 Solicita Relatório

Requisitos:

- REF-008 O sistema deve permitir a consulta remotamente das informações dos dados armazenados em arquivo.

Restrições:

- Pré-condição. O módulo cliente precisa estar conectado com o servidor.
- Pós-condição. Os relatórios foram consultados.

Cenário:

A. Verifica Relatório {Caminho Principal}.

A.1O usuário pressiona o ícone de pedido de relatório.

A.2 Sistema apresenta uma tela FORM_RELATORIO solicitando que usuário entre com data inicial e final do relatório.

A.3 Cliente pede ao servidor o envio de relatórios passando data início e data Fim.

A.4 O servidor recebe o pedido.

A.5 O servidor envia a PDU armazenada na data solicitada.

A.6 O cliente fica aguardando o envio da PDU por parte do servidor.

A.7 O cliente recebe a PDU.

A.8 Mensagem correta, cliente pede próxima PDU.

A.9 Vai para o passo A.4.

A.10 Cliente pede o fim do envio das PDUs.

B. Mensagem Incorreta {Caminho Alternativo}.

B.1 Caso no passo A.7 a mensagem esteja incorreta, cliente pede retransmissão da última PDU enviada pelo servidor.

B.2 Vai para o passo A.4.

B.3 Após cinco tentativas vai para o passo A.10.

O diagrama de robustez da solicitação de relatório é apresentado na Figura 37 a seguir, e o diagrama de seqüência é apresentado na Figura 74 (**apêndice B).**

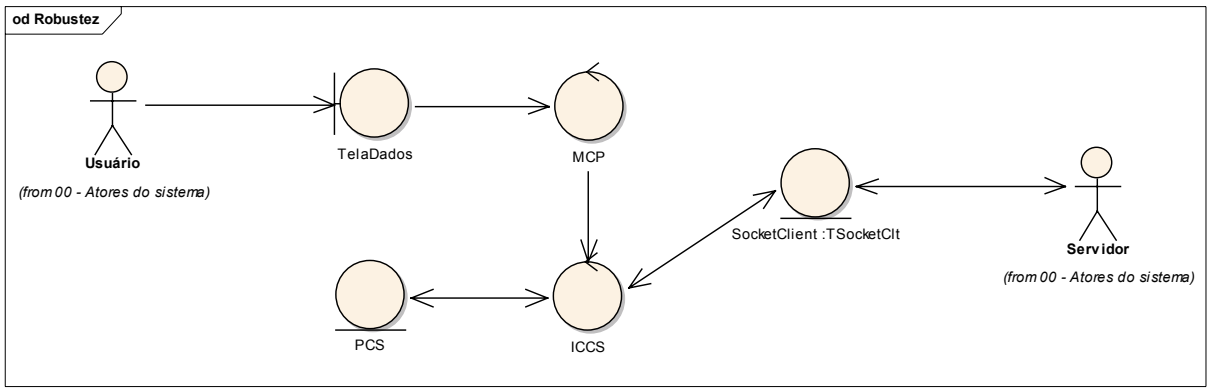

Figura 37 : Diagrama de Robustez - Solicita relatório.

5.5.7 USC-03.04 Solicita Alteração de Senha

Altera a senha de acesso

Requisitos:

- REF-009 O sistema deve permitir a alteração da senha de acesso remotamente.

Cenário:

A. Altera Senha {Caminho Principal}.

A.1 O Cliente pressiona o ícone de Alteração de senha.

A.2 O Sistema apresenta tela FORM_SENHA solicitando que usuário entre com Senha Antiga e Senha Nova.

A.3 O usuário entra com as senhas.

A.4 O cliente envia as senhas ao servidor.

A.5 O servidor autentica a senha.

A.6 O servidor muda a senha.

A.7 O servidor envia ao cliente a resposta da autenticação e mudança.

A.8 A senha é alterada.

B. Senha Incorreta {Caminho Alternativo}.

B.1 Se no passo A.5, a senha não puder ser validada, o servidor envia ao cliente que a senha é incorreta.

B.2 O cliente apresenta ao usuário uma mensagem "Senha incorreta!".

B.3 Volta ao passo A.2.

B.4 Após cinco tentativas fecha conexão.

A seguir, na Figura 38 é apresentado o diagrama de robustez da Solicitação de alteração da senha e o diagrama de seqüência encontra-se na Figura 75 (**apêndice B).**

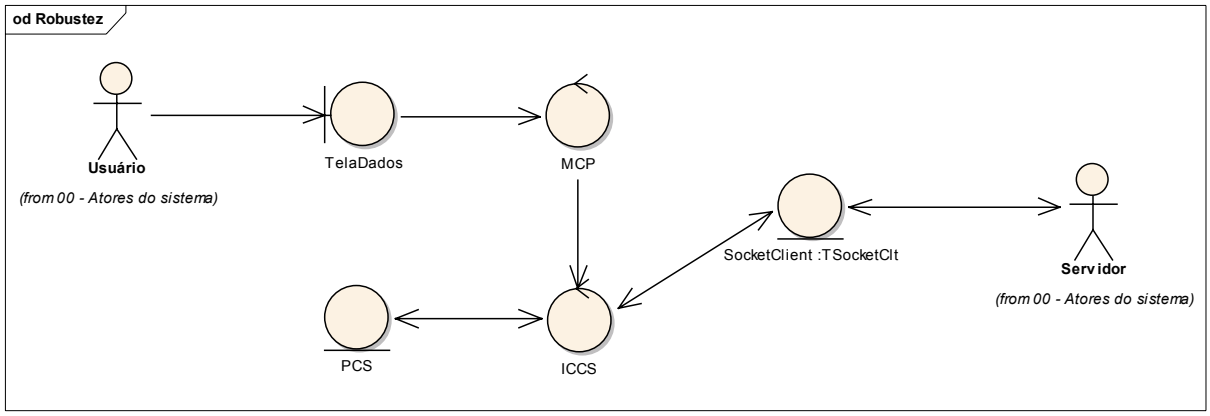

Figura 38 : Diagrama de Robustez - Solicita alteração de senha

5.5.8 USC-04.01 Salvar LOG em Arquivo

Seqüências de passos referentes a manipulação de arquivos

Requisitos:

- REF-010 O sistema deve permitir o registro dos dados capturados em um arquivo (log).

Cenário:

A. Inicia a Gravação dos dados {Caminho Principal}.

A.1 O usuário pressiona o botão de gravação de arquivos.

A.2 O APTI apresenta a tela de arquivos FORM_ARQUIVOS.

A.3 O usuário seleciona o local onde deseja salvar o arquivo.

A.4 O usuário entra com o nome do arquivo.

A.5 O usuário pressiona o botão de Criar Arquivo.

A.6 O sistema cria e abre o arquivo selecionado.

A.7 O AP libera o botão de GRAVA PDU.

A.8 O usuário pressiona o botão de GRAVA PDU.

A.9 O sistema armazena as PDUs recebidas no arquivo criado.

A.10 O usuário pressiona o botão de PARADA de gravação dos dados no arquivo.

A.11 O sistema fecha o arquivo.

B. Pausa a Gravação dos dados {Caminho Alternativo}.

B.1 Depois do passo A.8, o usuário pode optar por pausar a gravação das PDUs em arquivos.

B.2 O usuário pressiona o botão de PAUSA de gravação dos dados.

B.3 O sistema não grava mais as PDUs no arquivo.

B.4 Retorna ao passo A.8 ou vai ao passo A.10.

Na Figura 39 a seguir, é apresentado o diagrama de robustez da funcionalidade Salvar LOG. O diagrama de seqüência é apresentado no **Apêndice B** na Figura 76.

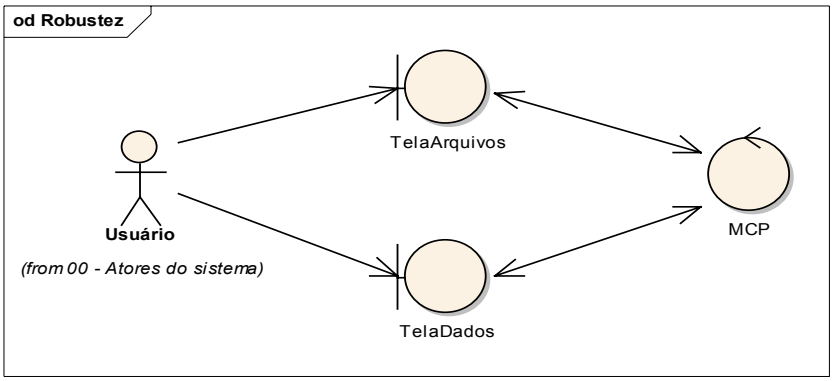

Figura 39 : Diagrama de Robustez - Salvar LOG em arquivo.

5.5.9 USC-04.02 Carregar LOG dos Arquivos

Requisitos:

REF-011 O sistema deve permitir a consulta das informações dos dados armazenados em arquivo.

Cenário:

A. Abre Arquivos {Caminho Principal}.

A.1 O usuário pressiona o botão de Abre arquivos.

A.2 O APTI apresenta a tela de arquivos FORM_ARQUIVOS.

A.3 O usuário seleciona a pasta onde o arquivo se encontra armazenado.

A.4 O usuário entra com o nome ou seleciona o arquivo.

A.5 O usuário escolhe abrir todo o Arquivo.

A.6 O usuário pressiona o botão Abrir Arquivo.

A.7 O sistema abre o arquivo e apresenta na tela de dados FORM_TELADADOS, o histórico das PDUs registradas.

B. Consulta por Períodos {Caminho Alternativo}.

B.1 No passo A.5, o usuário pode optar por carregar as PDUs gravadas em um determinado período.

B.2 O sistema apresenta uma Tela, solicitando a data inicial e final a ser recuperada.

B.3 O usuário preenche as datas.

B.4 Vai para o passo A.6.

O diagrama de seqüência da funcionalidade Carregar LOG é apresentado na Figura 77 do **Apêndice B,** e o diagrama de robustez é apresentado na Figura 40 a seguir.

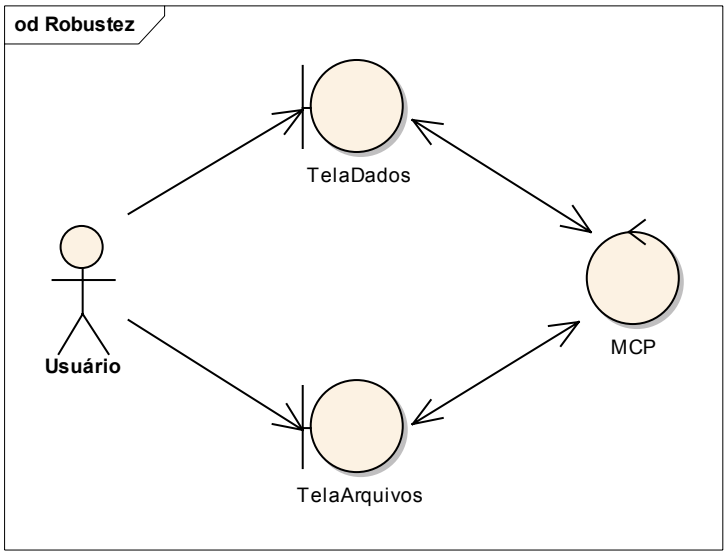

Figura 40 : Diagrama de Robustez - Carregar LOG dos arquivos.

5.5.10 USC-04.03 Salvar Configurações em Arquivo

Requisitos:

- REF-012 O sistema deve permitir o registro das configurações em um arquivo.

Cenário:

A. Grava as Configurações {Caminho Principal}.

A.1 O usuário pressiona na barra de ferramentas Arquivo e em Salvar Config.

A.2 O sistema cria o arquivo com nome "APTI.cfg" caso ainda não exista esse arquivo.

A.3 O sistema armazena as configurações no arquivo criado.

A.4 O sistema fecha a arquivo.

A seguir, na Figura 41 é apresentado o diagrama de robustez da funcionalidade salvar configurações em arquivo e na Figura 78 (**apêndice B)** encontra-se o diagrama de seqüências**.**

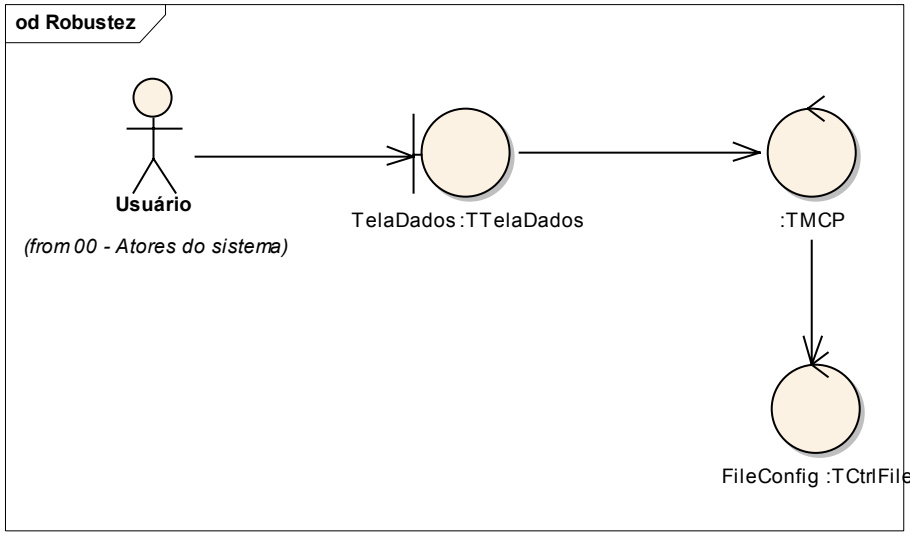

Figura 41 : Diagrama de Robustez - Salvar configurações em arquivo.

5.5.11 USC-04.04 Carregar Configurações dos Arquivos

Requisitos:

- REF-013 O sistema deve permitir que as configurações armazenadas em arquivo sejam carregadas.

Cenário:

A. Carrega as configurações {Caminho Principal}.

A.1 O usuário pressiona na barra de ferramentas Arquivo e em Carregar Config.

A.2 O sistema abre o arquivo com nome " APTI.cfg".

A.3 O sistema carrega as configurações que estavam armazenadas no arquivo.

A.4 O sistema fecha a arquivo.

A seguir, na Figura 42 é apresentado o diagrama de robustez da funcionalidade Carregar configurações do Arquivo e o diagrama de seqüência é apresentado na Figura 79 (**apêndice B).**

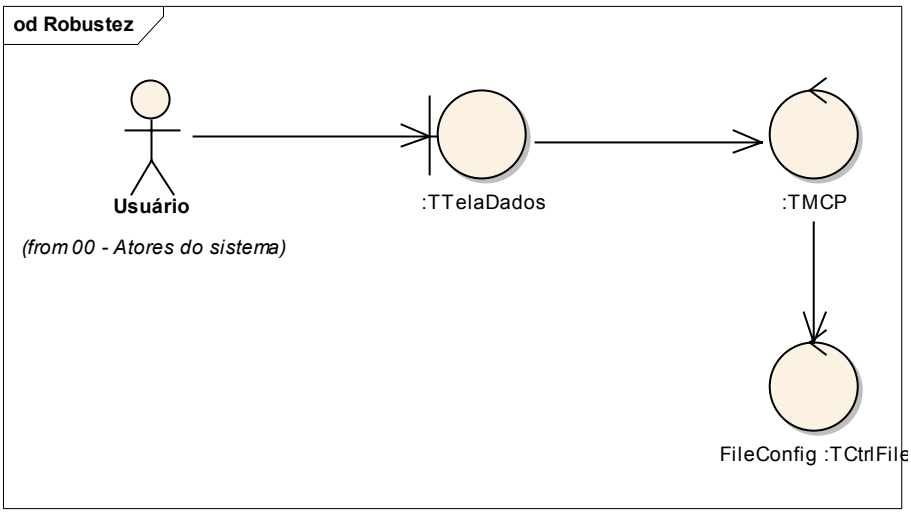

Figura 42 : Diagrama de Robustez - Carregar configurações dos arquivos.

5.5.12 USC-05.01 Configura Portas

Requisitos:

- REF-014 O sistema deve permitir que o usuário configure as portas seriais.

Cenário:

A. Configura Porta {Caminho Principal}.

A.1 O usuário pressiona na barra de ferramentas Configurações e em Configurações de Porta.

A.2 O sistema apresenta a tela de configurações da porta.

A.3 O operador seleciona as configurações da porta que ele deseja e confirma.

A.4 O sistema grava estas configurações.

O diagrama de seqüência da configuração de portas é apresentado na Figura 80, já o diagrama de robustez pode ser consultado a seguir na Figura 43.

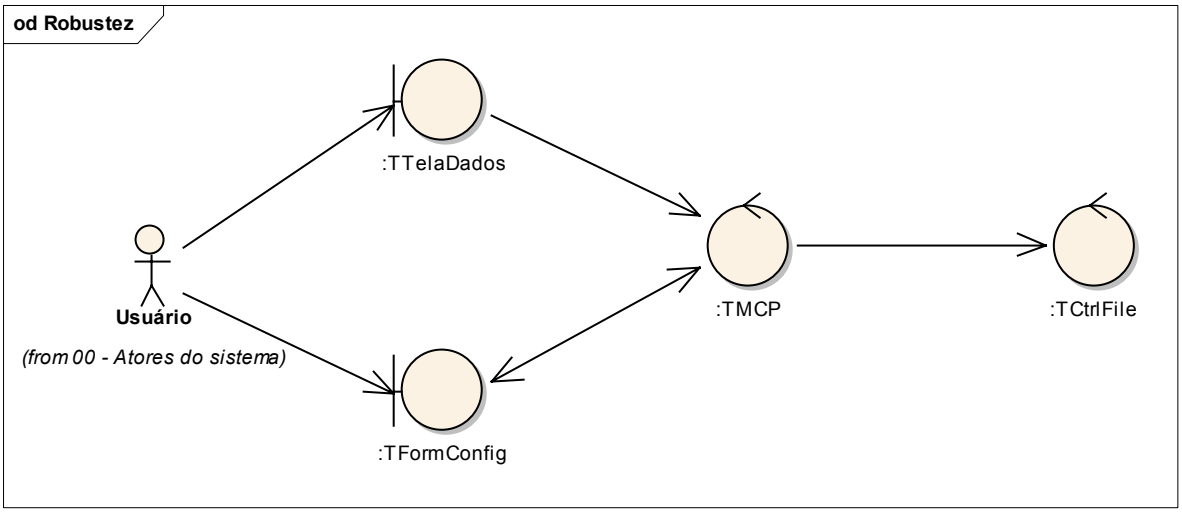

Figura 43 : Diagrama de Robustez - Configura portas.

5.5.13 USC-05.02 Configura Senha

Requisitos:

- REF-016 O sistema deve permitir a alteração da senha de acesso tanto no cliente como no servidor.

Cenário:

A. Muda Senha {Caminho Principal}.

A.1 O usuário pressiona na barra de ferramentas Configurações e em Configura Senhas de Acesso.

A.2 O sistema apresenta a tela de configuração da senha.

A.3 O usuário entra com a senha atual, a nova senha e confirma.

A.4 O sistema valida a senha e efetua a alteração.

A.5 O sistema grava estas configurações.

Na Figura 44 a seguir, é apresentado o diagrama de robustez da configuração da senha e o diagrama de seqüência pode ser consultado no **Apêndice B** na Figura 81.

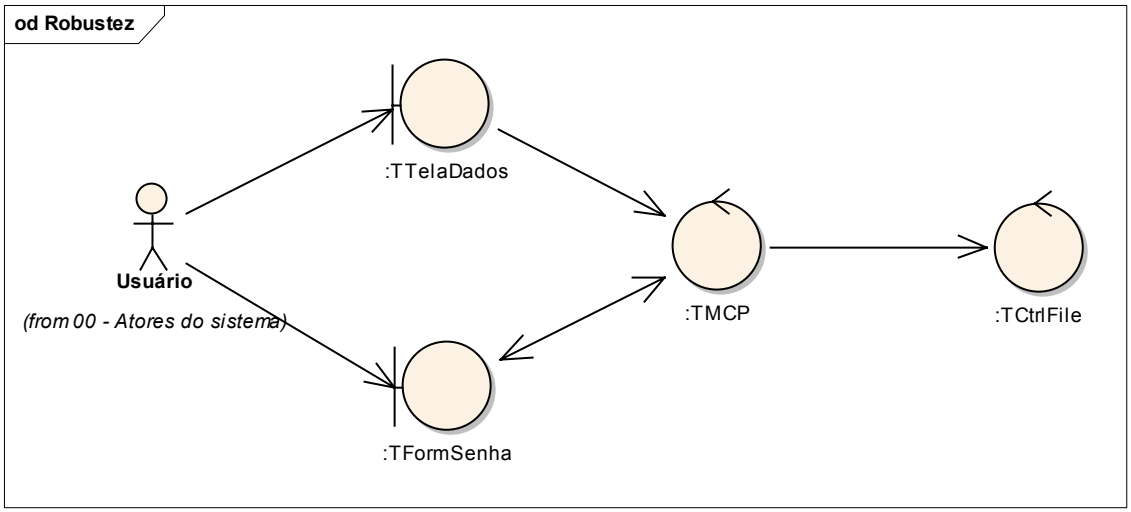

Figura 44 : Diagrama de Robustez - Configura senha.

5.5.14 USC-05.03 Configura TCP/IP

Requisitos:

- REF-015 O sistema deve permitir que o usuário configure as portas TCP/IP e endereço IP que o sistema usará para se comunicar entre cliente e servidor.

Cenário:

A. Configurações Porta TCP/IP - SERVIDOR {Caminho Principal}.

A.1 O usuário pressiona na barra de ferramentas Configurações e em Configura Portas Cliente/Servidor.

A.2 O sistema apresenta a tela de configuração das portas TCP/IP.

A.3 O usuário configura a porta que ele deseja que o servidor reserve para a comunicação com o cliente e confirma.

A.4 O sistema grava estas configurações.

B. Configura Porta e endereço TCP/IP - CLIENTE {Caminho Principal}.

B.1 O usuário pressiona na barra de ferramentas Configurações e em Configura portas Cliente/Servidor.

B.2 O sistema apresenta a tela de configuração das portas TCP/IP.

B.3 O usuário configura a porta que ele usará para se comunicar com o servidor, o endereço deste servidor e confirma.

B.4 O sistema grava estas configurações.

O diagrama de robustez da configuração TCP/IP pode ser visto a seguir na Figura 45. No **Apêndice B** na Figura 82 é apresentado o diagrama de seqüência desta funcionalidade.

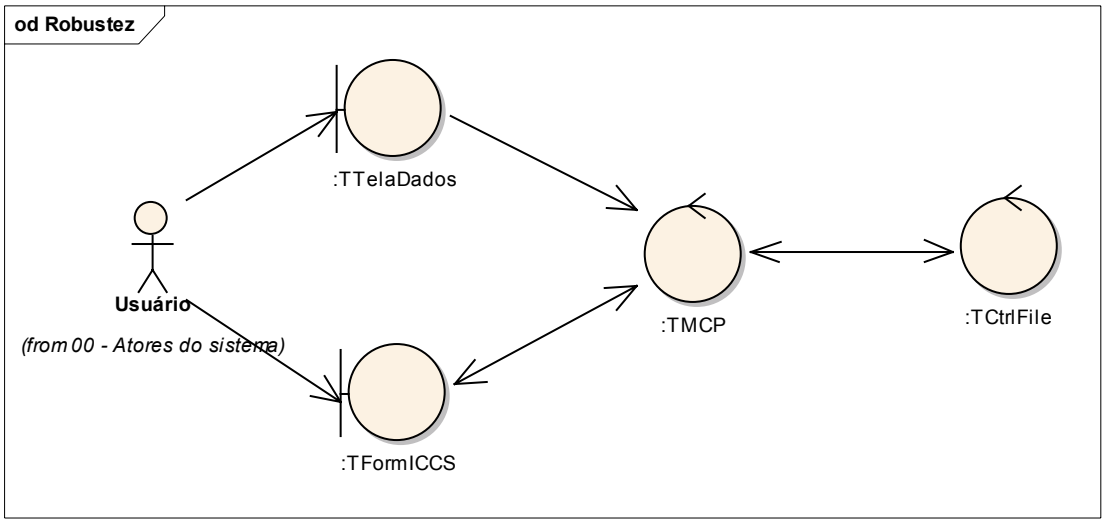

Figura 45 : Diagrama de Robustez - Configura TCP/IP.

5.5.15 USC-05.04 Editor de Relatório

Requisitos:

- REF-017 O sistema deve apresentar um editor para que o usuário possa editar (recortar, colar, copiar, apagar) os *logs* de mensagens recebidos ou qualquer outro arquivo de formato texto(.txt). Deve permitir ainda carregar e gravar os dados em arquivos.

Cenário:

A. Editor {Caminho Principal}.

A.1 O usuário pressiona o botão Editor.

A.2 O sistema apresenta a tela de Edição de PDUs.

A.3 O usuário carrega e edita as PDUs de sua preferência.

A.4 O usuário pode salvar estas alterações em arquivo.

A.5 O usuário fecha o editor.

O diagrama de robustez do editor de relatório é apresentado na Figura 46 a seguir, e na Figura 83 do **Apêndice B,** encontra-se o diagrama de seqüência.

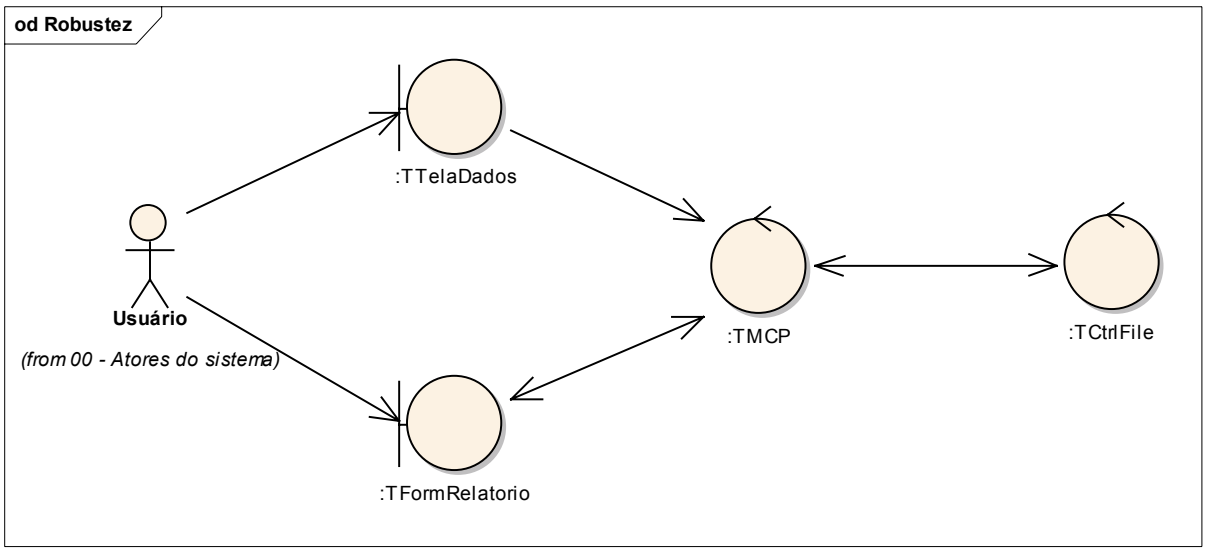

Figura 46 : Diagrama de Robustez - Editor de relatório.

5.6 OPERAÇÃO BÁSICA DO *SOFTWARE*

Esta seção mostra como utilizar o *software* AP, descrevendo suas operações básicas por meio do fluxo normal de operação e ilustrando estas operações com base em suas interfaces. Este capítulo também descreve detalhes da operação do AP, como por exemplo, as funcionalidades, seqüência de inicialização (servidor-cliente), sempre salientando as principais vantagens do uso do *software* em relação a como era feito anteriormente sem este recurso. Também é apresentado o requisito funcional que foi atendido com a função apresentada. Estes requisitos já foram detalhados no item 5.2.

5.6.1 Iniciando o AP em modo Local

Ao iniciar o programa este apresenta a tela principal (Figura 47) para que o usuário possa escolher em que modo este deseja trabalhar, modo local ou modo remoto.

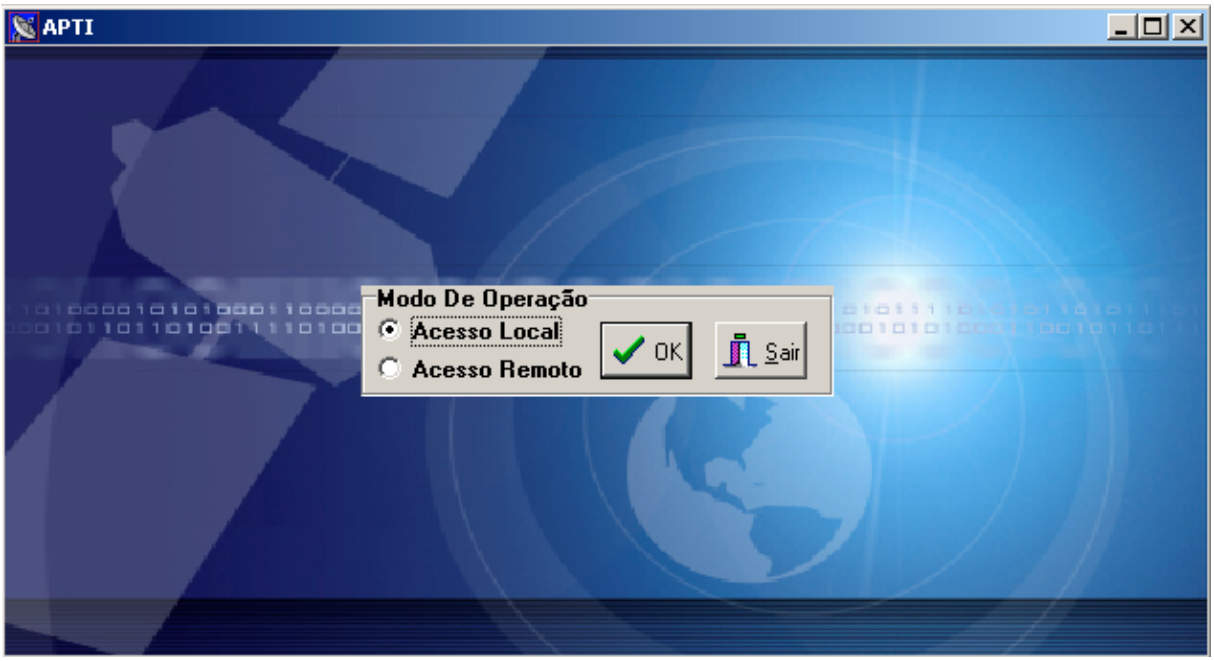

Figura 47 : Tela Principal.

Neste modo, os dados são capturados localmente, por meio da conexão de um cabo ligado à porta serial do computador e a placa PADI que é a responsável em capturar os dados de comunicação entre o PABX e a TI. A tela de dados do modo local é apresentada na Figura 48,

onde o usuário tem o acesso a maioria das funções do AP, como controles na captura de dados, acesso a arquivos de log, configurações diversas e edição de logs.

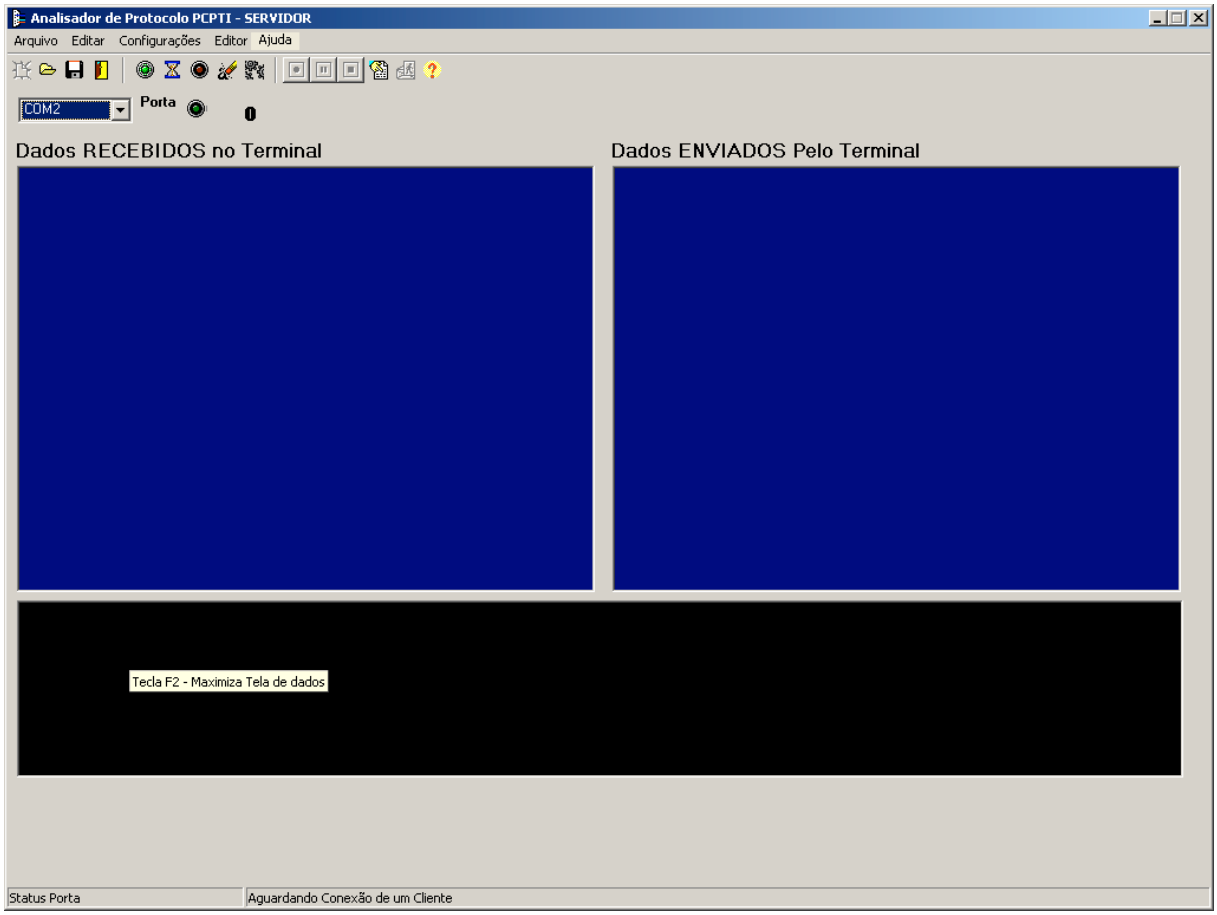

Figura 48 : Tela de dados do modo local (Servidor).

Antes da implementação deste projeto, não era possível ver e analisar o tráfego de mensagens entre TI e PABX, o que dificultava a resolução de problemas quando acontecia alguma falha de comunicação. Nestes casos, os desenvolvedores do PABX e do TI tinham que procurar em seus respectivos softwares para tentar identificar o que estava ocorrendo e onde estava o problema, o que causava perda de tempo na tentativa de descobrir o defeito e envolvendo muitas pessoas, que na maioria das vezes, não seria necessário caso fosse utilizado um AP como este.

5.6.2 Iniciando o AP em modo Remoto

A fim de atender os requisitos REF-001[15] e REF-002[16], a seguir uma descrição das funcionalidades implementadas no AP.

Para o REF-002, o usuário pode escolher trabalhar com o AP no modo remoto ou local. Neste caso, é apresentada a tela de dados do modo remoto(Figura 49), com a maioria das funções e controles do *software* bloqueadas, pois estas só estão disponíveis após conexão com o servidor.

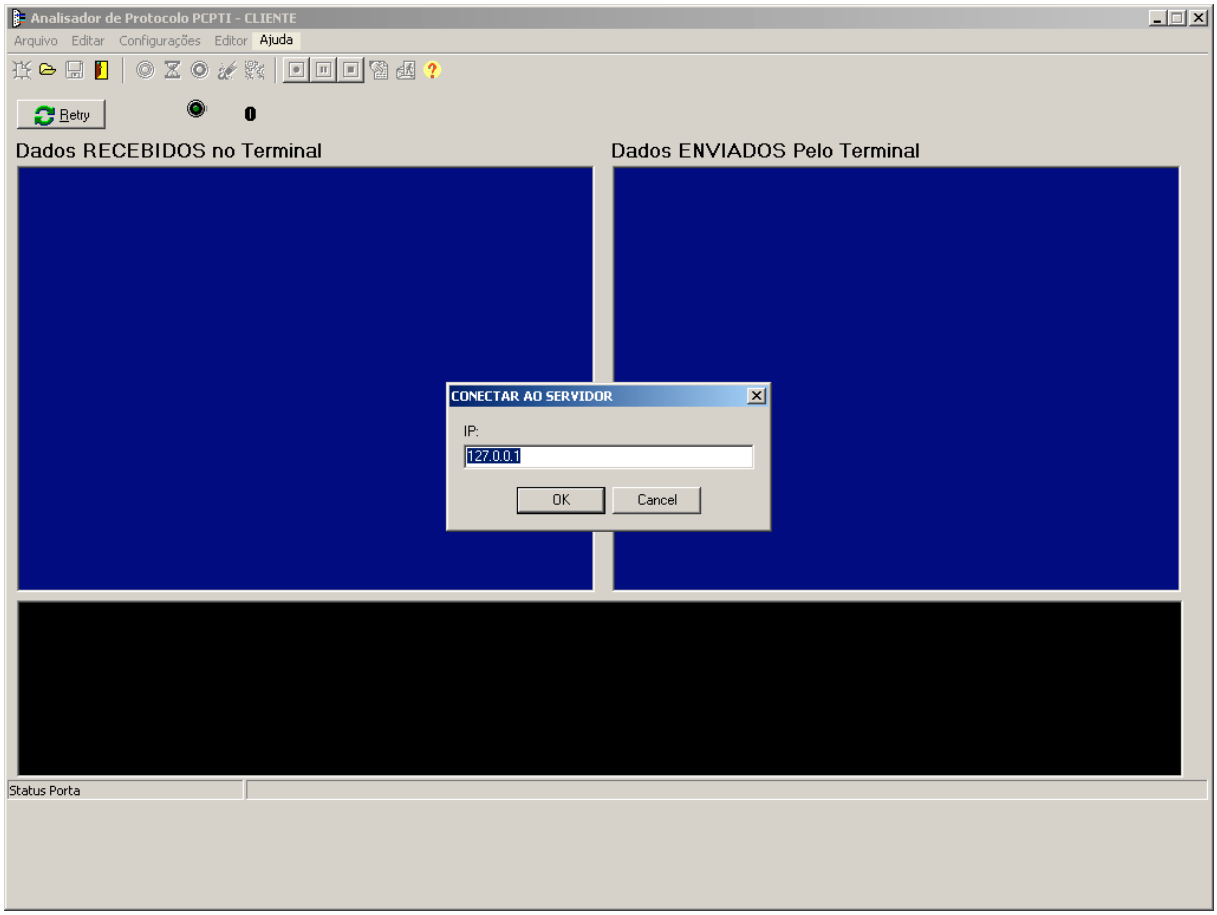

Figura 49 : Tela de dados do modo remoto (Cliente).

[15] REF-001: O sistema deve permitir que um usuário possa efetuar o *login* no sistema.
[16] REF-002: O sistema deve permitir o acesso remoto a um servidor que captura os dados.

A tela de dados apresenta uma segunda interface para que o usuário entre com o endereço IP do servidor que este deseja se conectar. Neste caso, o servidor já deve estar executando e aguardando a conexão de um AP cliente. Após a confirmação do endereço IP o cliente conecta ao servidor que aceita a conexão e solicita ao cliente uma senha. No AP cliente, é apresentada a interface de senha (Figura 50), solicitando que o usuário entre com a senha de acesso. O usuário fornece a senha válida e o servidor, recebendo esta senha, valida e confirma para o cliente. A partir deste momento, a conexão é efetivada e o AP cliente libera todos os comandos e funções da ferramenta para o usuário.

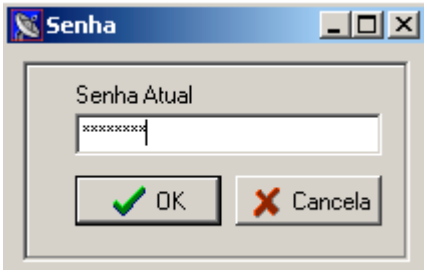

Figura 50 : Interface de controle de acesso.

Neste modo, não é possível a captura dos dados por meio de portas seriais do computador onde está executando o AP cliente, somente por meio da conexão TCP/IP com o servidor.

Antes do desenvolvimento deste projeto, a equipe técnica da INTELBRAS não tinha condições, caso necessitasse, analisar remotamente os dados deste tipo de comunicação. Nestes casos, o técnico deveria se deslocar até o local para analisar o tráfego de mensagens. Com a funcionalidade de acesso remoto, ganhou-se tempo de resposta mais rápido para resolver problemas de comunicação com um menor custo.

5.6.3 Selecionando uma Porta para a captura dos Dados

Na tela de dados, após o usuário ter escolhido o modo local, este deve selecionar uma porta serial disponível para a captura dos dados. Caso o *software* não encontre nenhuma porta serial disponível, o campo porta pode ser editado. Isto pode acontecer em computadores que tenham o registro do Windows (REGEDIT) bloqueado.

Após ter feito esta escolha, o usuário pode dar início à captura dos dados pressionando o botão de captura de dados localizado na barra de tarefas superior, sendo o quinto botão da esquerda para a direita, como apresentado na Figura 51 a seguir.

Figura 51 : Botões de captura dos dados.

O usuário também pode parar momentaneamente (PAUSA) a captura dos dados. Neste caso, a partir deste momento os dados recebidos não são registrados, ou seja, estes dados são perdidos.

O usuário pode parar definitivamente a captura dos dados, fechando a porta de captura, neste caso todas as PDUs são limpas da tela de dados.

Sem esta facilidade, seria difícil a identificação das portas seriais usadas para a captura dos dados por meio da PADI, que estão disponíveis no PC. A Figura 52 apresenta a tela de seleção de portas seriais.

Figura 52 : Tela de seleção das portas seriais.

Os dados são manipulados segundo as regras do protocolo PCPTI e apresentados na janela de dados, de acordo com os filtros selecionados previamente. O item 5.6.4 descreve estes filtros. A Figura 53 apresenta a janela de dados com as PDUs capturadas.

Caso o usuário necessite analisar todos os dados da PDU, basta selecionar a mensagem desejada e pressionar o botão direito do mouse, então é apresentada uma segunda tela de informação contendo a PDU sem formatação (Figura 53).

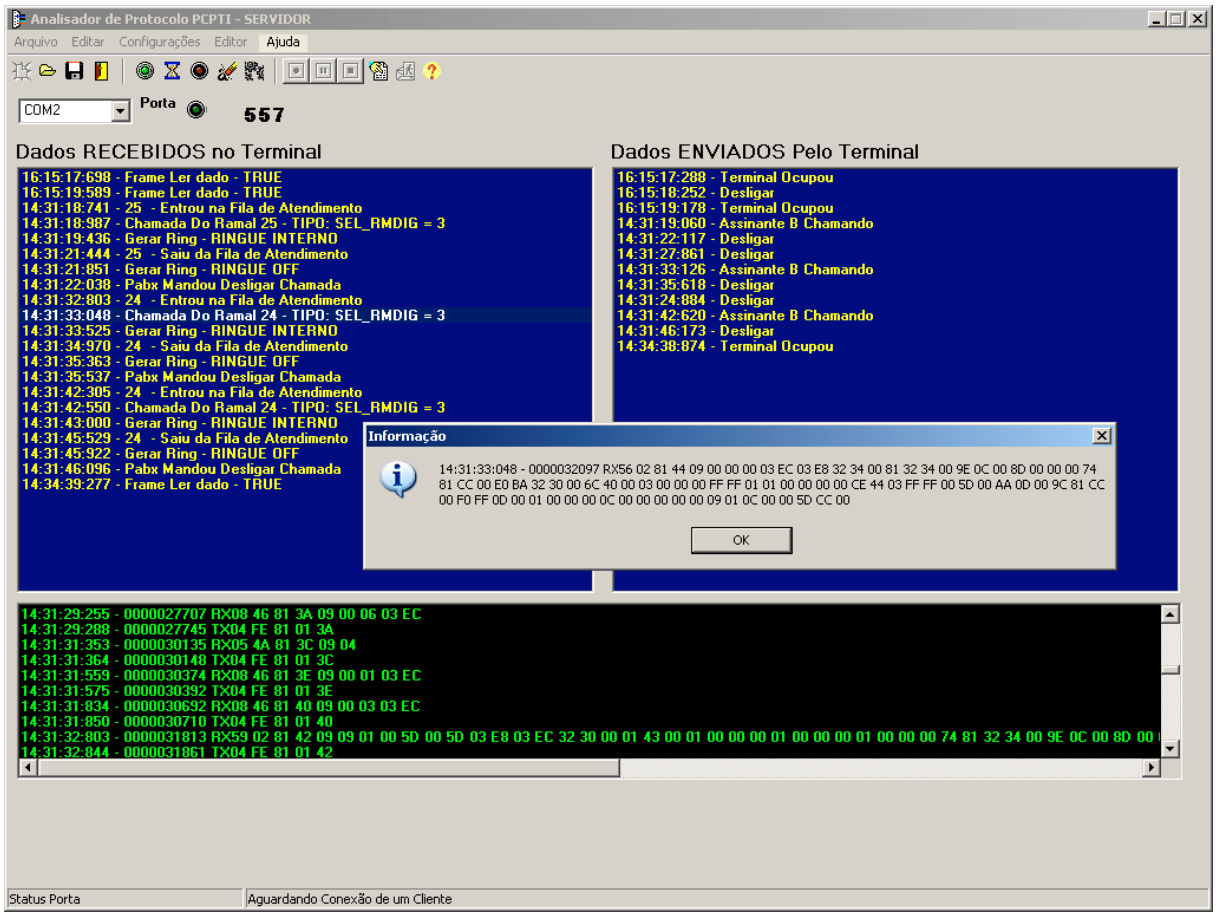

Figura 53 : Tela de dados com PDUs capturadas.

O procedimento para o início da captura de dados, no modo remoto, é o mesmo do modo local, mas não sendo necessário selecionar a porta por onde os dados são recebidos. Neste modo, o usuário tem que escolher entre receber os dados desde o início da captura das PDUs, pelo servidor, ou somente a partir do pedido de PDUs pelo cliente, como apresenta a Figura 54.

Esta facilidade de escolher quando se deseja receber os dados oferece uma maior flexibilidade para o usuário que pode não desejar receber todos os dados capturados pelo servidor, pois este já pode estar capturando as PDUs durante mais tempo que o desejado, por exemplo, uma semana antes do início da análise desejada.

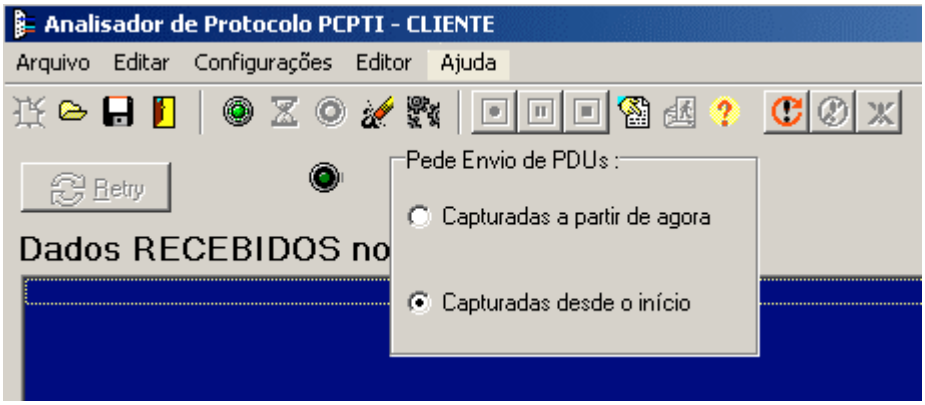

Figura 54 : Escolha do início de captura de dados.

A tela de dados é dividida basicamente em três partes:

- janela da esquerda que apresenta PDUs recebidas no TI, ou seja, enviadas pelo PABX;
- janela localizada à direita que apresenta as PDUs enviadas pelo TI para o PABX;
- janela localizada na parte inferior que é onde são apresentadas todas as mensagens, sem formatação, trocadas entre PABX e TI, capturas pela PADI. Esta janela pode ser maximizada para tela inteira, como apresenta a Figura 55, utilizando a tecla F2, da mesma forma pode ser feito para minimizar esta tela.

Esta divisão em janelas facilita a análise e identificação do sentido das mensagens, ou seja, quem é o responsável pelo envio da PDU: ou o PABX ou o TI. A apresentação da PDU recebida sem formatação é importante, pois nestas PDUs existem muito mais informação do que aparece na janela com formatação.

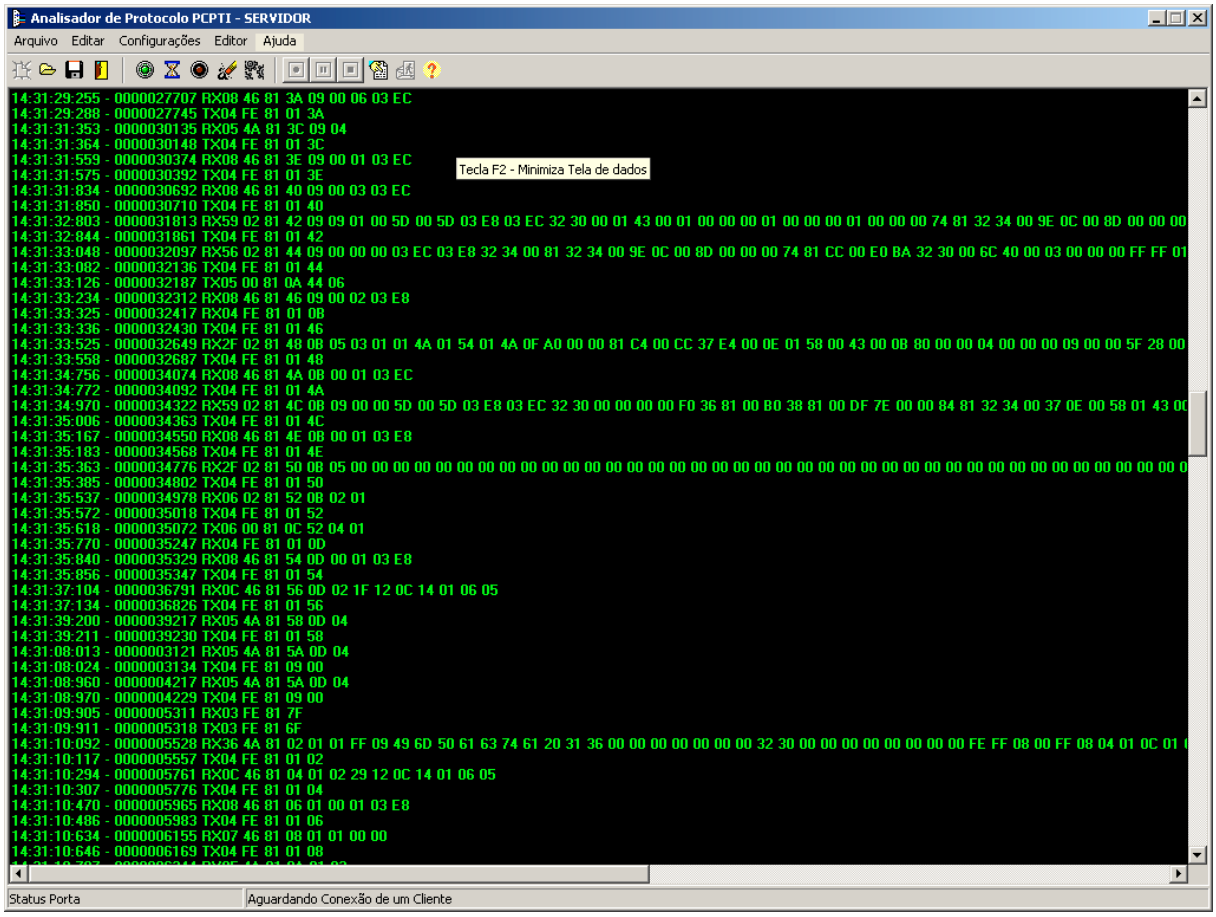

Figura 55 : Tela de dados sem formatação Maximizada.

Na parte da tela dos dados sem formatação, são permitidas funções de edição, tais como: recortar, copiar, colar e selecionar tudo.A maximização desta tela facilita a análise caso o técnico queira verificar os dados puramente sem formatação, sendo uma funcionalidade muito usada na etapa de desenvolvimento.

Neste item, foram descritas as funcionalidades que atendem os requisitos REF-003[17], REF-004[18], REF-005[19] e REF-007[20].

---

[17] REF-003: O sistema deve permitir o acesso às portas seriais para a captura dos dados.

[18] REF-004: O sistema deve ser capaz de analisar as PDUs recebidas, tratá-las segundo o Protocolo PCPTI e apresentá-las na tela de dados para o usuário.

[19] REF-005: O sistema deve permitir que o usuário possa consultar os logs de dados capturados

5.6.4 Configurações de Mensagens (Filtros)

Esta funcionalidade foi desenvolvida para atender o requisito REF-006[21] e permite que o usuário possa escolher os tipos de mensagens que serão apresentadas na tela. Para isto, o usuário deve pressionar na barra de ferramentas a opção "Configurações", e escolher "Configura Filtros de Mensagens". Neste caso, é apresentada a tela de Filtros conforme Figura 56.

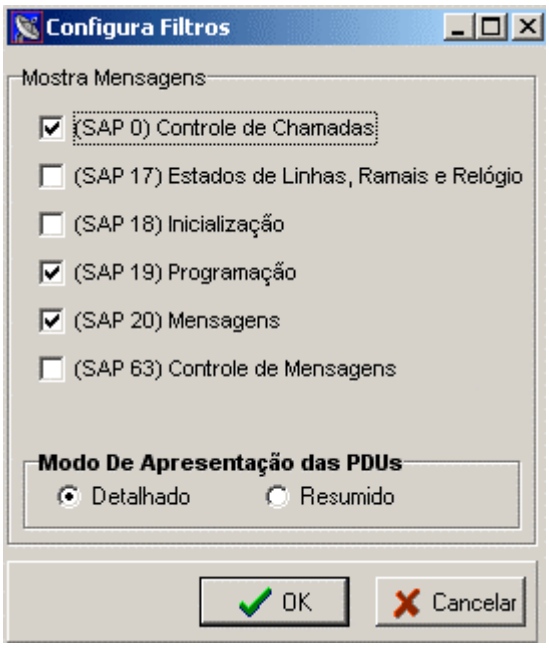

Figura 56 : Tela de Configuração de Filtros de Mensagens.

Tipos de Filtros:

SAP 0: são apresentadas todas as mensagens referentes ao controle de chamadas.

SAP 17: são apresentadas todas as mensagens referentes aos estados de linhas e ramais e mensagens de calendário e relógio.

---

[20] REF-007: O sistema deve permitir que o usuário possa consultar remotamente (no Cliente) e em tempo real, os logs de dados capturados pelo servidor.

[21] REF-006: O sistema deve permitir que o usuário configure qual tipo de mensagens ele deseja visualizar (Filtros de mensagens).

SAP 18: apresenta as mensagens de inicialização com tipo e versão de PABX, número do TI e outras informações e ainda mostra mensagem de sincronismo da camada física.

SAP 19: apresenta as mensagens pertinentes a programações como senha de ramal desvios de chamadas entre outras.

SAP 20: apresenta as PDUs referente à troca de mensagens de texto entre os TIs, serviço semelhante às mensagens SMS de celular.

SAP 63: controle de mensagens são tipos de mensagens de nível físico usado para o controle de mensagens trocadas entre PABX e TI. Sempre que uma mensagem é recebida, é retornada uma resposta para quem enviou esta Mensagem.

DETALHADO: Apresenta a PDU de uma forma mais detalhada e de fácil entendimento.

RESUMIDO: Apresenta o tipo de quadro trocado entre PABX e TI. Usado mais em tempo de desenvolvimento.

Sem estes filtros todas as mensagens são apresentadas na tela de dados e acabando dificultando a análise dos PDUs que realmente interessam.

5.6.5 Configurações do *software*

O AP permite várias configurações, como por exemplo, das portas seriais, configurações de portas e endereço TCP/IP e senha de acesso ao servidor.

Para configurar a porta serial o usuário deve pressionar, na barra de ferramentas, a opção “Configurações” e escolher “Configura Porta”, neste momento é apresentada a tela de configuração da porta serial como apresenta a Figura 57. Nesta tela, as seguintes opções são apresentadas:

velocidade da porta: é a taxa de transmissão, em bits por segundo, usada entre os dispositivos;

bts de dados: o número de bits que cada dado (byte) pode ter;

paridade: usada para controle de erros. Pode ser programado como sem paridade ou ainda se paridade par ou ímpar;

stop Bit: Quantos bits de parada são usados para identificar final de cada dado.

Figura 57 : Tela de Configuração de Portas Seriais.

A configuração TCP/IP deve ser programada tanto no servidor quanto no cliente. No servidor, é programada a porta que o servidor reservará para que o cliente se conecte a este. Já no cliente é programada a porta que este se conectará ao servidor e o endereço IP deste servidor. As Figura 58 e Figura 59 apresentam as telas de configurações das portas TCP/IP do servidor e cliente, respectivamente.

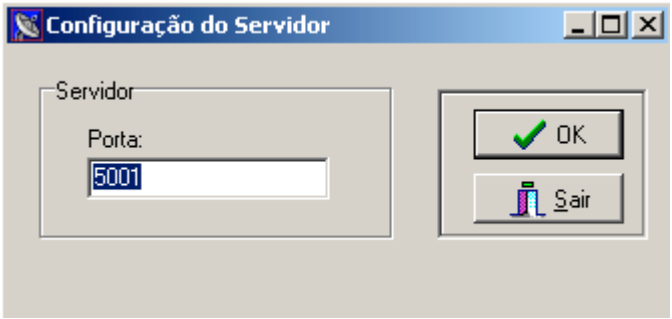

Figura 58 : Tela de configuração TCP/IP do servidor.

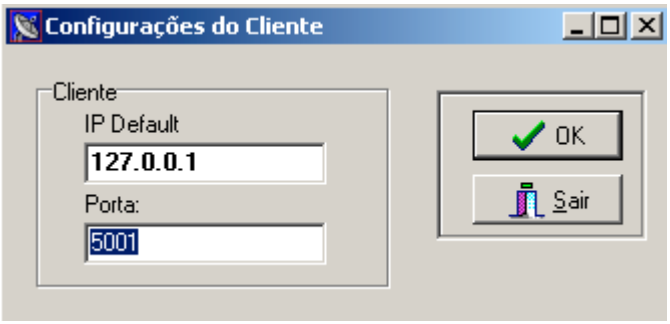

Figura 59 : Tela de Configuração de TCP/IP do Cliente.

Tanto no cliente como no servidor é possível acessar esta configuração, pressionando em "Configurações" a opção "Porta cliente /servidor" na barra de ferramentas.

Outra configuração possível é a alteração da senha de acesso. A Figura 60 apresentam a tela de configuração da senha onde no primeiro campo o usuário fornece a senha atual e no segundo a nova senha de acesso.

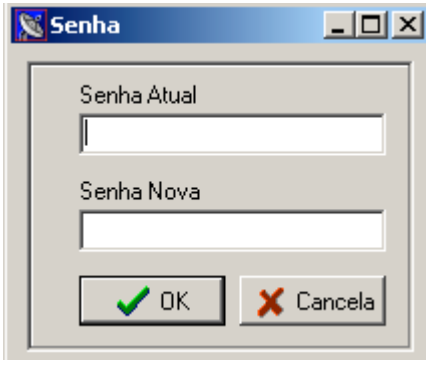

Figura 60 : Tela de Configuração de Senha de Acesso.

Todas as configurações podem ser gravadas em arquivo. Para isto, basta o usuário pressionar em "Arquivos" e "Salvar Config". As configurações são armazenadas em um arquivo com nome fixo de "APTI.cfg" e sempre fica armazenada na mesma pasta onde o APTI.exe está instalado. Caso este arquivo ainda não exista, este será criado.

Estas configurações também podem ser carregadas a qualquer momento, bastando para isso que o usuário clique em "Arquivos" e "Carregar Config".

Todas estas configurações contemplam os requisitos funcionais REF-014[22], REF-015[23], REF-016[24], REF-009[25], REF-012 [26]e REF-013[27].

5.6.6 Registro em arquivos das PDUs capturadas

O *software* possui uma funcionalidade que atende o requisito REF-010[28], que permite que todas a PDUs recebidas possam ser gravadas em arquivo. Não é necessário usar nenhum tipo de banco de dados para se armazenar as PDUs, pois a quantidade de dados armazenados não é muito grande e o editor de relatório disponibiliza funcionalidades para se procurar determinadas mensagens capturadas em determinadas datas. O AP também poderá ser instalados em PCs que podem não suportar determinado banco de dados. Outro motivo foi que, se notou que na maioria dos APs estudados, as PDUs eram armazenadas em arquivos de log e não em banco de dados.

Para gravar as PDUs em arquivo o usuário tem que seguir os seguintes passos:

- Pressionar o botão "Abrir Arquivo" para gravação de PDUs, neste instante é apresentada a tela de criar arquivo de *log* (Figura 61), então o usuário escolhe o local e o arquivo que será criado para armazenar as PDUs recebidas.
- Após ter sido criado o arquivo de *log* o AP libera os botões de gravação de PDUs, com isso o usuário tem total controle da gravação da PDU em arquivo e a qualquer momento este pode parar a gravação momentaneamente ou finalizar a gravação das PDUs recebidas.

---

[22] REF-014 O sistema deve permitir que o usuário configure as portas seriais.

[23] REF-015: O sistema deve permitir que o usuário configure as portas TCP/IP e endereço IP que o sistema usará para se comunicar entre cliente e servidor.

[24] REF-016: O sistema deve permitir a alteração da senha de acesso tanto no cliente como no servidor.

[25] REF-009: O sistema deve permitir a alteração da senha de acesso remotamente.

[26] REF-012: O sistema deve permitir o registro das configurações em um arquivo.

[27] REF-013: O sistema deve permitir que as configurações armazenadas em arquivo sejam carregadas.

[28] REF-010: O sistema deve permitir o registro dos dados capturados em um arquivo (log).

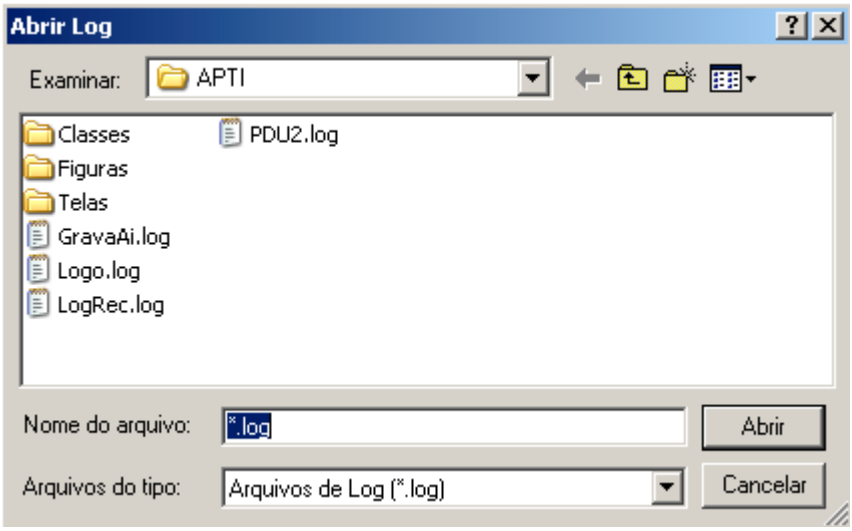

Figura 61 : Tela de criação e abertura de arquivos de log.

O AP permite ainda, quando estiver sendo executado em modo remoto, que o cliente solicite para o servidor gravar localmente as PDUs recebidas da PADI. Neste caso, o cliente também tem controle da gravação das PDUs por meio de botões de gravação de PDUs remotamente. Os botões utilizados são os três últimos situados à direita na barra de ferramentas. A limitação, neste caso, é que o usuário tem que digitar todo o caminho do local de gravação do arquivo de PDUs, caso seja escolhido somente o nome do arquivo, este é gravado na pasta onde o APTI.EXE do servidor estiver instalado.

5.6.7 Consulta de PDUs armazenadas em arquivo

Como o AP permite a gravação das PDUs em arquivo, este permite também carregar estes dados na tela de dados seguindo os passos:

- Pressionar o botão “Carregar Log” na tela de dados, neste instante é apresentada a tela de procura de arquivo de log (Figura 61), então o usuário escolhe onde se encontra o arquivo.
- Após ter sido localizado o arquivo o usuário pressiona “OK", neste momento é apresentado uma tela de calendário (Figura 62), para que o usuário escolha uma data. Então todas as PDUs armazenas a partir desta data são apresentadas na tela. As PDUs são apresentadas nas telas de acordo com os filtros de mensagem selecionados.

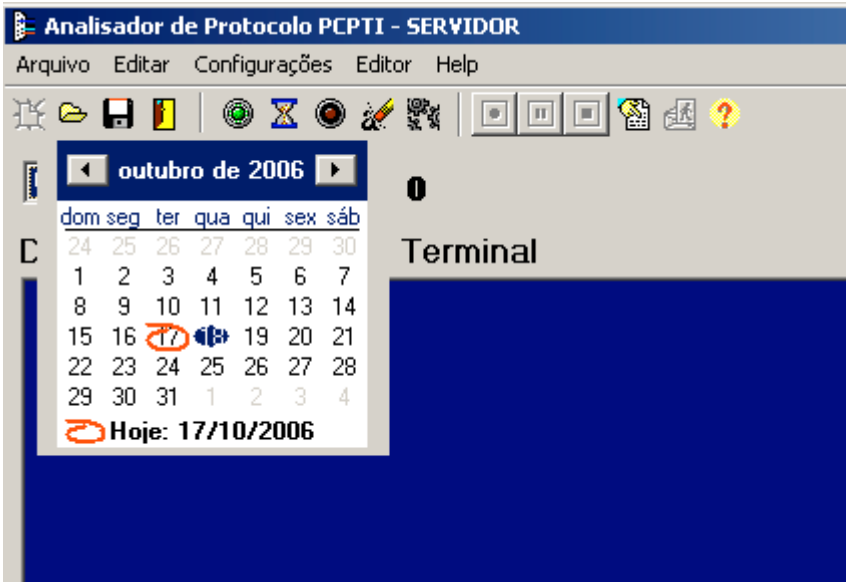

Figura 62 : Tela de calendário.

Esta facilidade de poder carregar as PDUs do arquivo a qualquer momento é muito útil quando se está recebendo os dados, determinados filtros estão selecionados e aconteceu um problema qualquer e as mensagens não estão aparecendo na tela porque o filtro referente aquele tipo de mensagem não estava selecionado. Neste caso, o usuário pode parar a captura, limpar a tela, selecionar corretamente os filtros e carregar o arquivo com o período correspondente que se pretende analisar. Com isso, as mensagens desejadas podem ser analisadas sem problema. Os filtros só funcionam na apresentação dos dados na tela e não na gravação dos logs. Mas para isto ser possível, é preciso ativar a gravação dos dados. Esta funcionalidade descrita atende dois requisitos: o REF-011[29] e o REF-008[30].

[29] REF-011: O sistema deve permitir a consulta das informações dos dados armazenados em arquivo.

[30] REF-008: O sistema deve permitir a consulta remotamente das informações dos dados armazenados em arquivo.

5.6.8 Editor de Relatório

Para facilitar a manipulação dos arquivos de *log* e atender ao requisito REF-017[31], foi implementado um editor de relatório que permite uma maior flexibilidade do que ter, simplesmente, uma função de impressão de relatórios. Com isto, o usuário pode manipular e editar os *logs* de PDUs. Este editor permite ainda a manipulação do arquivo, imprimindo o texto editado, realizando buscas por palavra-chave, mudança de tipo e tamanho de fontes, negrito, itálico e sublinhado. A Figura 63 apresenta a tela de edição de relatório.

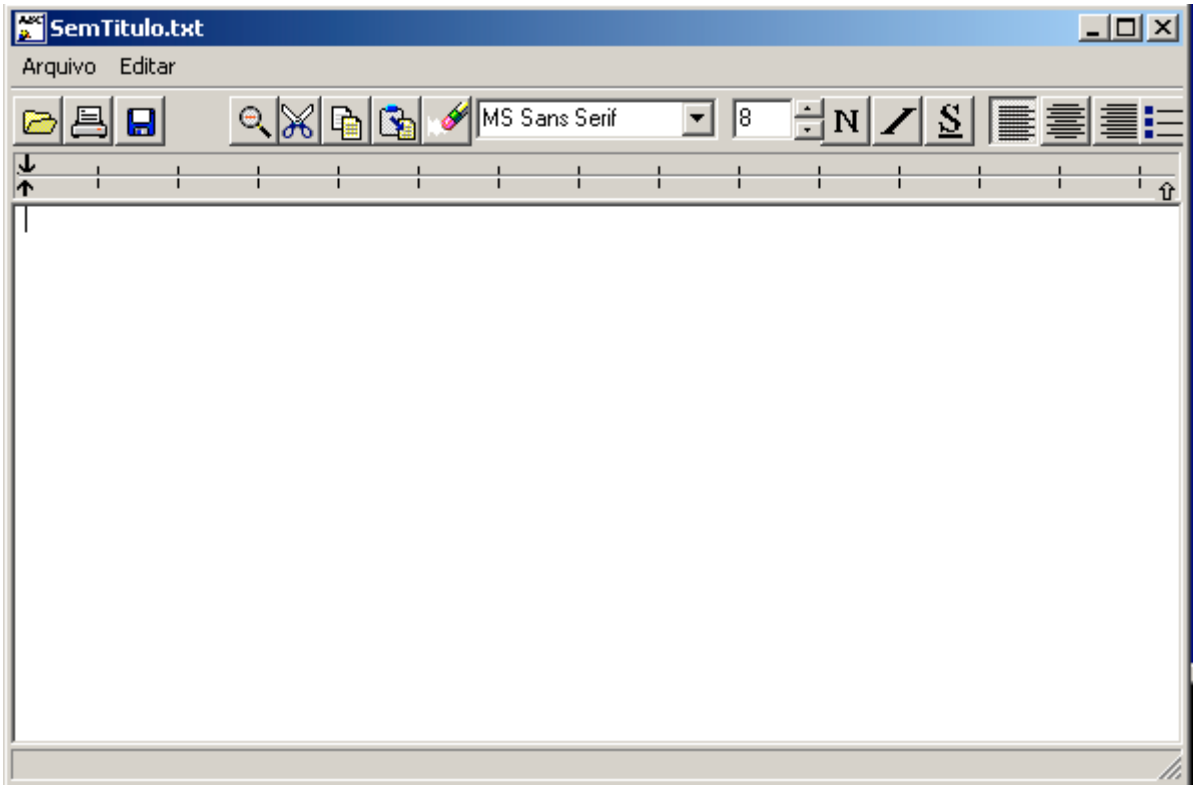

Figura 63 : Tela do editor de relatório.

Quando estiver em modo remoto o editor possui uma função que permite solicitar ao servidor que envie um arquivo de *logs*, fornecendo o nome do arquivo, a data inicial e a data final. A

[31] REF-017: O sistema deve apresentar um editor para que o usuário possa editar (recortar, colar, copiar, apagar) os logs de mensagens recebidos ou qualquer outro arquivo de formato texto(.txt). Deve permitir ainda carregar e salvar os dados em arquivos.

Figura 64 apresenta a tela de pedido de envio de arquivo ,o servidor recebendo este pedido envia todas as PDUs gravadas naquele arquivo solicitado e entre as datas solicitadas.

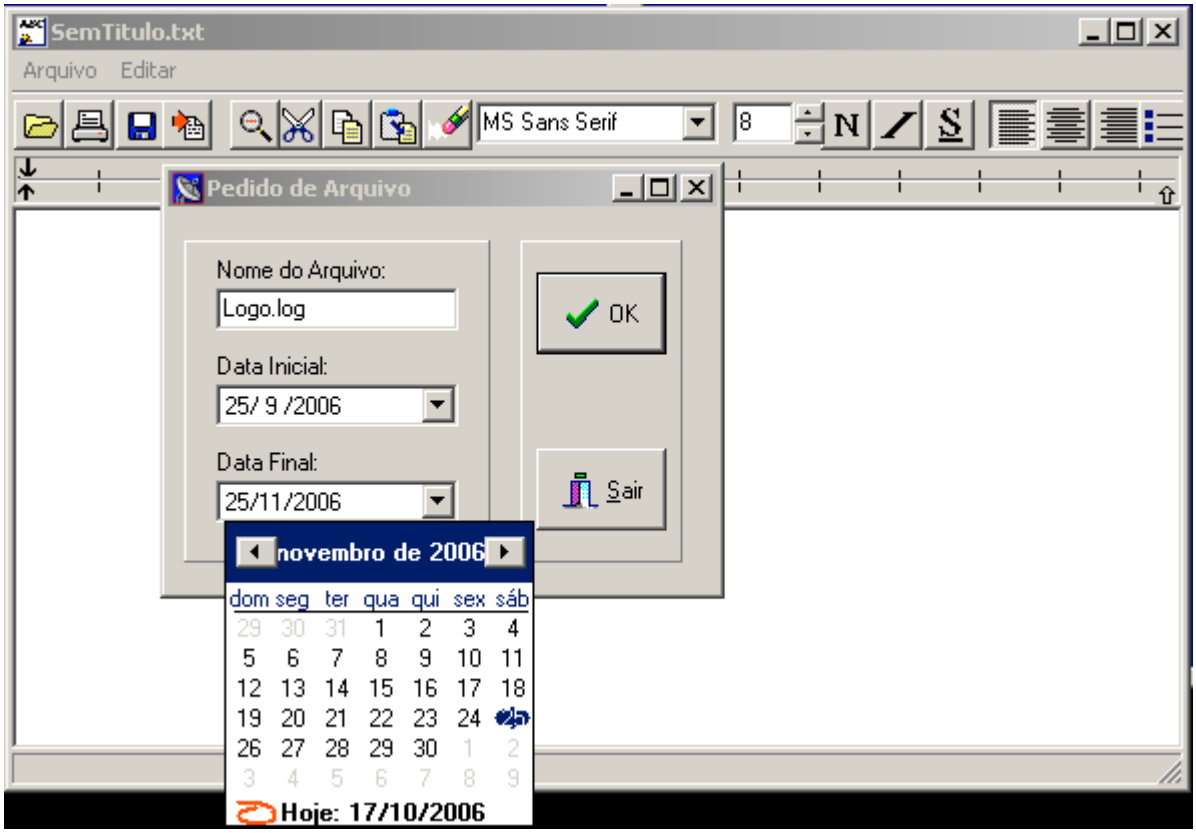

Figura 64 : Tela de pedido de relatório Remoto.

A edição dos dados é muito usada quando se está procurando um determinado dado ou uma seqüência de mensagem, com a finalidade de se identificar algum erro no protocolo, já que a ferramenta não tem esta função de mostrar que ocorreu um erro de protocolo.

6 CONCLUSÃO

Neste capítulo, são apresentados os principais resultados obtidos com este trabalho. Também são apresentadas as dificuldades encontradas durante o desenvolvimento do TCC. Por fim são apresentados os planos futuros para a evolução do *software*.

6.1 RESULTADOS OBTIDOS

Antes que o projeto fosse desenvolvido, tentou-se contratar uma empresa de desenvolvimento de *software* para desenvolver este *software* AP. Esta empresa já havia desenvolvido outros projetos semelhantes para a INTELBRAS, mas o custo financeiro e o tempo de desenvolvimento orçado tornaram o projeto proibitivo. Ao contrário disto, observou-se que se o projeto fosse desenvolvido internamente tornar-se-ia viável e com um custo mais baixo do que o orçado pela empresa.

O trabalho foi desenvolvido e foram obtidos todos os resultados esperados. O *software* resultante deste projeto está ajudando muito no desenvolvimento e testes dos sistemas de PABX e TI da INTELBRAS, permitindo um ganho de tempo de desenvolvimento e de confiabilidade nos equipamentos envolvidos, permitindo descobrir defeitos que possivelmente só seriam encontrados após a comercialização, causando um desgaste a imagem do produto.

Como mencionado no item 1.6 o *software* é uma ferramenta de auxílio para o diagnóstico de problemas e não tem funções que o habilite a identificar problemas de forma autônoma. A detecção e a resolução de problemas depende da competência e experiência do técnico ou engenheiro que estiver utilizando a ferramenta.

As principais facilidades do AP são a captura e apresentação das mensagens que trafegam entre PABX e TI e também o modo remoto que permite ao usuário acompanhar o que está acontecendo com os equipamentos, mesmo não estando no local, onde estes estão instalados. Antes da implementação deste projeto, como já mencionado no capitulo 5, não era possível analisar o tráfego de mensagens entre TI e PABX, dificultando a resolução de problemas de comunicação e causando um atraso no projeto. Sem o modo remoto, a equipe técnica não tinha condições de analisar os dados remotamente, necessitando o deslocamento do técnico

até o local para analisar o tráfego de mensagens. Com o acesso remoto, ganhou-se tempo de resposta para resolver problemas de comunicação, com um custo menor. Mas todas as outras funcionalidades são importantes, pois facilitam o uso da ferramenta.

A ferramenta foi e está sendo usada pela equipe de desenvolvimento de terminais da INTELBRAS para desenvolver e ajustar o novo terminal NKT 2165 e o NKT 4245. Finalizada esta fase de desenvolvimento o equipamento vai para a fase de testes. Durante os testes, tanto em laboratório quanto em campo, o AP está sendo usado para identificar falhas e erros de *software* destes produtos. Em especial nos testes de campo, foi usada a facilidade "Modo Remoto", não sendo necessário o técnico se deslocar várias vezes até o local de teste. E por fim a ferramenta será utilizada pelo setor de assistência técnica da INTELBRAS para detectar possíveis erros de comunicação entre sistemas PABX e o TI, caso eles ocorram, após a comercialização dos produtos.

Sem dúvida nenhuma o que fez a diferença na realização do trabalho foi a especificação clara e objetiva das tarefas a serem realizadas e também o modo que elas foram orientadas, com uma presença assídua do orientador para que o trabalho não tomasse um rumo errado, não sobrecarregando as épocas de entrega e também não deixando que surpresas se apresentassem durante a realização do projeto.

6.2 DIFICULDADES ENCONTRADAS

Na primeira parte do trabalho, referente ao TCC1, foram encontradas várias dificuldades. Foi também, uma parte muito trabalhosa, mas necessária para deixar o documento claro e de fácil entendimento, para que não surgisse nenhuma dúvida quanto à elaboração e objetivos do trabalho.

No aspecto metodológico, surgiu uma certa dificuldade em se conseguir classificar o tipo de trabalho proposto, pelo fato da literatura existente tratar este assunto de uma forma superficial, não apresentando detalhes dos tipos de trabalhos e pesquisas já realizados, o que dificultou em se tentar encaixar o trabalho proposto em um tipo específico.

A falta de literatura sobre analisadores de protocolo para PABX foi um dos maiores problemas encontrados. O material existente geralmente aborda superficialmente o assunto e,

na maioria das vezes, contém informações puramente comerciais disponibilizadas nos *sites* dos fabricantes. Literatura sobre PABX não é muito comum no meio acadêmico e o que se encontra é muito superficial. Isto ocorre, possivelmente por não haver um padrão no desenvolvimento de PABX e também pelo fato de os fabricantes não exporem suas tecnologias. Isto dificulta muito o entendimento de como funciona um PABX. Até mesmo os fabricantes têm dificuldades para encontrar ferramentas que os auxiliem no desenvolvimento de novos equipamentos.

Com relação ao protocolo PCPTI, desenvolvido internamente na INTELBRAS, para a comunicação entre PABX e TI, também não é vasta a documentação e a única referência existente era o próprio código fonte dos programas.

Outra dificuldade encontrada foi o envolvimento da equipe de desenvolvimento da INTELBRAS no desenvolvimento do AP, pois ela seria responsável pelo desenvolvimento da PADI e qualquer alteração a que se referia na captura dos dados. Quando era necessária alguma alteração nesta parte de captura dos dados, muitas vezes não era possível ou foi feito pelo desenvolvedor do AP. Isto ocorreu pelo fato da equipe estar muito envolvida com projetos da empresa e não dispor de tempo para auxiliar o desenvolvimento do AP.

Na fase de implementação e testes do AP, esta foi facilitada pela disponibilidade de equipamentos para a realização dos testes de funcionalidade e captura dos dados. Estes equipamentos foram cedidos pela INTELBRAS.

Dentre os módulos do *software*, a implementação cliente/servidor foi a que mais exigiu esforço durante o desenvolvimento do AP. Por se tratar de programação distribuída, a complexidade inerente deste modelo de desenvolvimento exige maior atenção em todas as etapas desde a análise ao teste dos componentes de *software* relacionados à este módulo.

6.3 FUTUROS TRABALHOS

Como futuros trabalhos, os quais visam melhorar a funcionalidade do produto de *software* desenvolvido, abaixo estão listadas aquelas funcionalidades que já fazem parte da nova lista de requisitos para as futuras versões.

6.3.1 Captura de dados Pela USB

A captura de dados via porta USB era uma das facilidades propostas neste trabalho, só que não foi possível devido ao fato de que a parte que deveria ser desenvolvido do lado da PADI não foi possível ficar pronta em tempo. Neste caso, esta facilidade por ser de extrema importância para o projeto e também para INTELBRAS será desenvolvida após a conclusão do TCC. A USB é importante para o projeto, pois a porta serial apresenta limitações que não existem quando usada uma interface USB, tais como ser necessário acrescentar um circuito adicional, circuito óptico, por exemplo, para evitar que a porta serial do PC não seja danificada e também a dificuldade de encontrar uma porta serial disponível nos computadores modernos.

6.3.2 Analisador de Protocolo do PABX

Atualmente, não se conhece nenhuma ferramenta de análise de protocolos para PABX com as características desenvolvidas neste projeto. Com esta ferramenta seria possível analisar o protocolo usado pelo PABX para a comunicação de suas partes internas, placas e circuitos que formam o PABX.

Portanto, uma das sugestões para futuros trabalhos é estender as funcionalidades do AP para que este funcione para o PABX, bastando para isto que seja alterado somente a parte que faz a análise dos protocolos. Esta tarefa se torna de uma complexibilidade baixa, pelo fato que o *software* do AP foi desenvolvido de forma modular.

6.3.3 Testador

O projeto futuro mais interessante e mais complexo será transformar o AP, que hoje só captura os dados que passam na linha, ou seja, se comporta de uma forma passiva, para que este tenha um comportamento ativo, se tornando um testador tanto de PABX quando de TI. Poderá simular um Terminal ou um PABX dependendo da necessidade. Isto é importante no caso de se desejar simular determinada situação que seja muito difícil de ocorrer na prática, como por exemplo, o TI receber várias ligações ao mesmo tempo. Isto ajudaria muito no desenvolvimento e principalmente nos testes de validação dos equipamentos.

REFERÊNCIAS BIBLIOGRÁFICAS

ALVES, Luiz, **Comunicação de Dados**. 2. ed. São Paulo: Makron Books, 1994.

ALMEIDA, Altieni Rodrigues de, **Apostila sobre Microprocessadores,** janeiro de 2006

BELLAMY, John, **Digital Telephony**, Second Ed. John Wiley & Sons, Inc, 1991.

CANZIAN, Edmur, **Comunicação Serial - RS232** Conceitos Básicos Sobre Comunicação Serial: CNZ Engenharia e Informática LTDA. São Paulo: S.P. 2000. Disponível em: <http://arquivos.coinfo.cefetpb.edu.br/~leonidas/irc/apostilas/comun_serial.pdf >. Acesso em 13 de abr. 2006.

CISCO. PABX IP – A evolução do sistema telefônico. Disponível em: < http:// http://www.cisco.com/br/IPT/index.shtml>. Acesso em: 05 maio 2006.

DUQUE, Carlos Augusto, **Comunicação Serial**. Juiz de Fora: Minas Gerais 2001 Disponível em: <http://www.mestradoeletrica.ufjf.br/professores/duque/microp_cap10.pdf>. Acesso em 11 de abr. 2006.

ETHEREAL, **The world's most popular network protocol analyzer** 2001. Disponível em: <http://www.ethereal.com>. Acesso em 25 de mai. 2006.

GONÇALVES, Flávio de Andrade, **Asterisk PBX:** Como construir e configurar um PABX com *Software* Livre. 2005.

IEPM, ***Monitoring with tcpdump***, Revised 17 August 1999. Disponível em: <http://www-iepm.slac.stanford.edu/monitoring/passive/tcpdump.html> .Acesso em: 26 mai. 2006

INFINEON, **SCOUT-DX PSB 21373 Version 1.1** ***Data sheet***, München, Germany, maio 2002.

INTELBRAS, **Impacta Manual Técnico** 1 Ed. São José Santa Catarina, 2006

IPTRAF, **IP Network Monitoring** ***Software***. September 2005 Disponível em: < http://iptraf.seul.org/index.html> Acesso em 23 de mai. 2006.

JORDAN, Larry; CHURCHILL Bruce**: Comunicações e redes com o PC; tradução Ernesto Veras. - 5. ed. atual. - Rio de Janeiro: Axcel Books, 1994.**

MARTINS, Roberto - **A Fundamentação da Telefonia através da História** -Parte 1: Da Invenção ao Início do Século XX (pesquisa realizada para a Fundação Telefônica, em 2002). Disponível em: <http://www.museudotelefone.org.br/sistemas1.htm>. Acesso em 20 de mar. 2006

NETSAFE, ***InfiniStream Network Management***. 2006. Disponível em: http://www.netsafe.com.br. Acesso em 25 de mai. 2006.

NETWORK GENERAL, **Product Overview**. 2006. Disponível em: http://www.networkgeneral.com/Product_Home.aspx. Acesso em 25 de mai. 2006.

NETWORK INSTRUMENTS, **Observer® 11**. 2006. Disponível em: http://www.networkinstruments.com. Acesso em 20 de mai. 2006.

NTOP.ORG, **What is ntop? Overview**. 2006. Disponível em: http://www.ntop.org/overview.html. Acesso em 25 de mai. 2006.

PINTO, Laércio José Gonzaga. **Analisador de Protocolo ARP**. Goiania: CEFET-GO, 2004.

Disponível em <http://www.redes.cefetgo.br/gl_downloads/tcc/pdf/tcc_012.pdf>Acesso em: 20 mai. 2006

SATO, Alberto Mitsuo. **Banda larga e VOIP**. Tutorial Teleco. Publicado em 18 de nov. 2004. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialpabx/default.asp>. Acesso em: 03 abril 2006.

SANTOS, Raimundo dos, **Metodologia Científica**: a construção do conhecimento.3. ed. Rio de Janeiro: Rio de Janeiro,2000.

SILVA, Edna Lúcia da; MENEZES, Estera Muszkat. **Metodologia da pesquisa e elaboração de dissertação**. 3ª ed. Ver. atual. Florianópolis: Laboratório de Ensino a Distância da UFSC, 2001. Disponível em: <http://www.ppgep.ufsc.br>. Acesso em 20 de mar. 2006.

SOARES L. F. G.; LEMOS, G.; COLCHER, S. **Redes de Computadores : Das LANs e WANs às Redes ATM**, 2ª edição. Ed. Campus. Rio de Janeiro, Rio de Janeiro, 1997.

SPECIALSKI, Elizabeth, **Arquitetura de redes de Computadores**, INE - UFSC, Mar., 2000, Florianópolis.

SPITZ, Rejane et al, **Projeto e Planejamento de Interfaces.** In: Workshop Formação de Recursos Humanos em Tecnologia da Informação para o Estado do Rio de Janeiro, FAPERJ, 9., 2000, Rio de Janeiro. Disponível em: <http://www.rnp.br/ti-rj/final/gt6.pdf> Acesso em 31 de mar. 2006.

STMICROELETRONICS, **uPSD3234 Version 3.0** ***Data sheet***, july 2004

TANENBAUM, Andrew S. **Redes de computadores**. Tradução Vandernberg D. de Souza – Rio de Janeiro, 4. ed. 2003.

RONCERO, Valeriana Gomes et al, **Monitoramento do Protocolo RTSP (Real Time Streaming Protocol) utilizando NTop (Network Top)**, 2005. Disponível em: < http://www.rederio.br/downloads/pdf/nt00402.pdf> Acesso em 20 de mai. 2006.

TEKTRONIX, **TLA5000 Series Logic Analyzers** Disponível em: < http://www.tek.com/site/ps/58-16733/pdfs/58W_16733.pdf>. Acesso em 25 de mai. 2006

APÊNDICE (A): DIAGRAMA DE ESTADOS CLIENTE-SERVIDOR

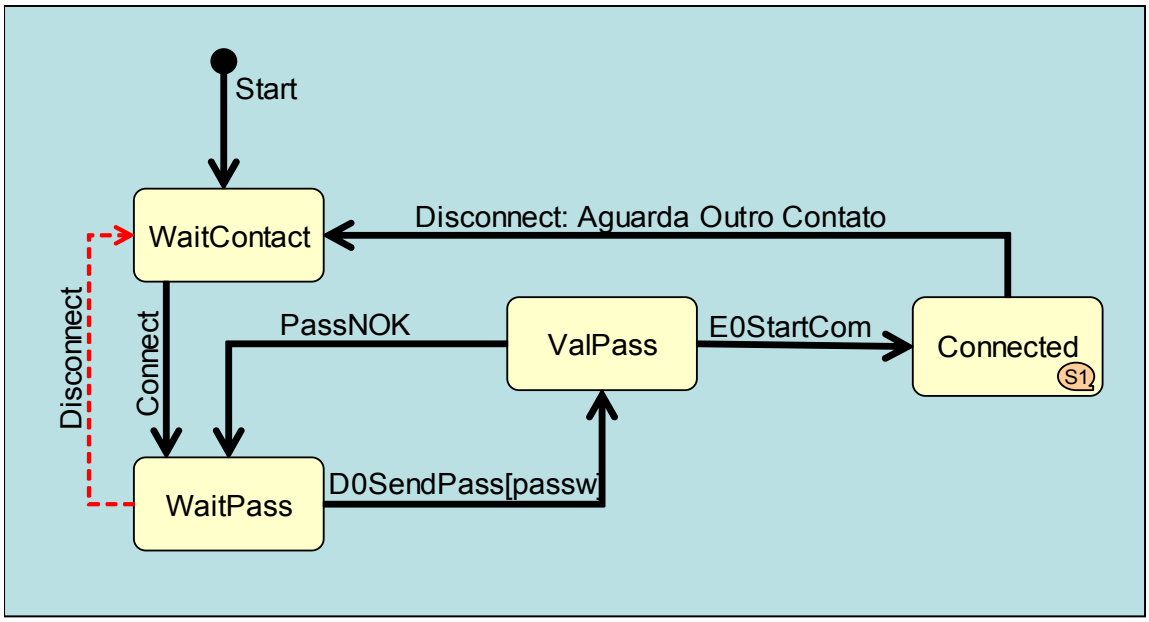

Figura 65 : Máquina de estado servidor: Início da conexão.

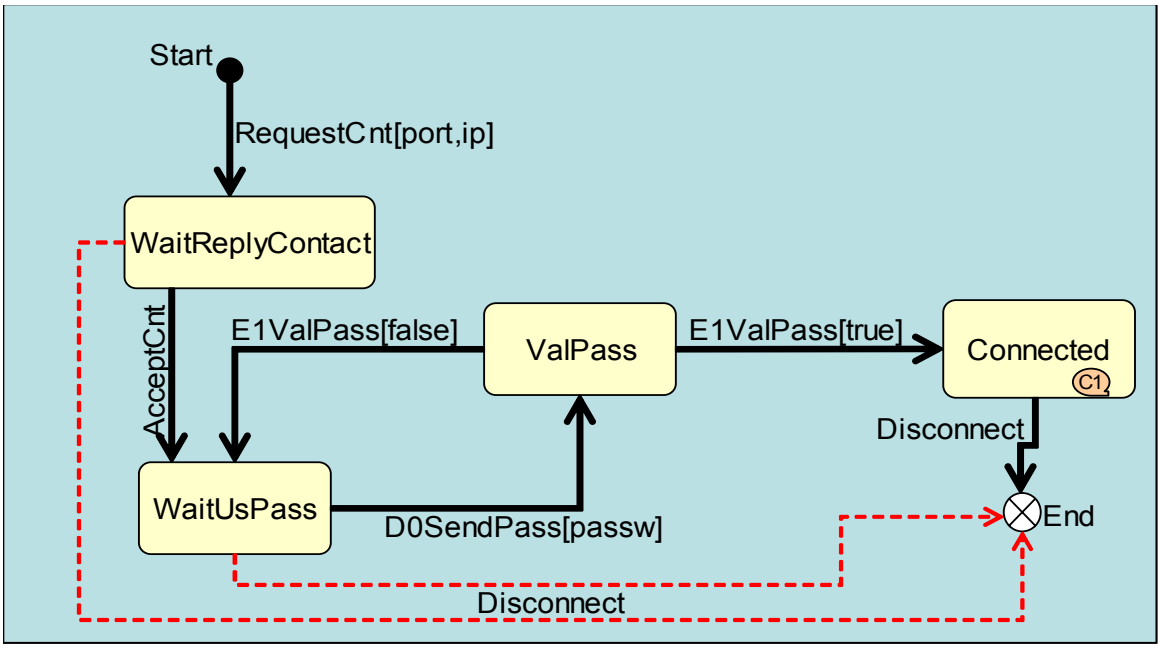

Figura 66 : Máquina de estado cliente: Início da conexão.

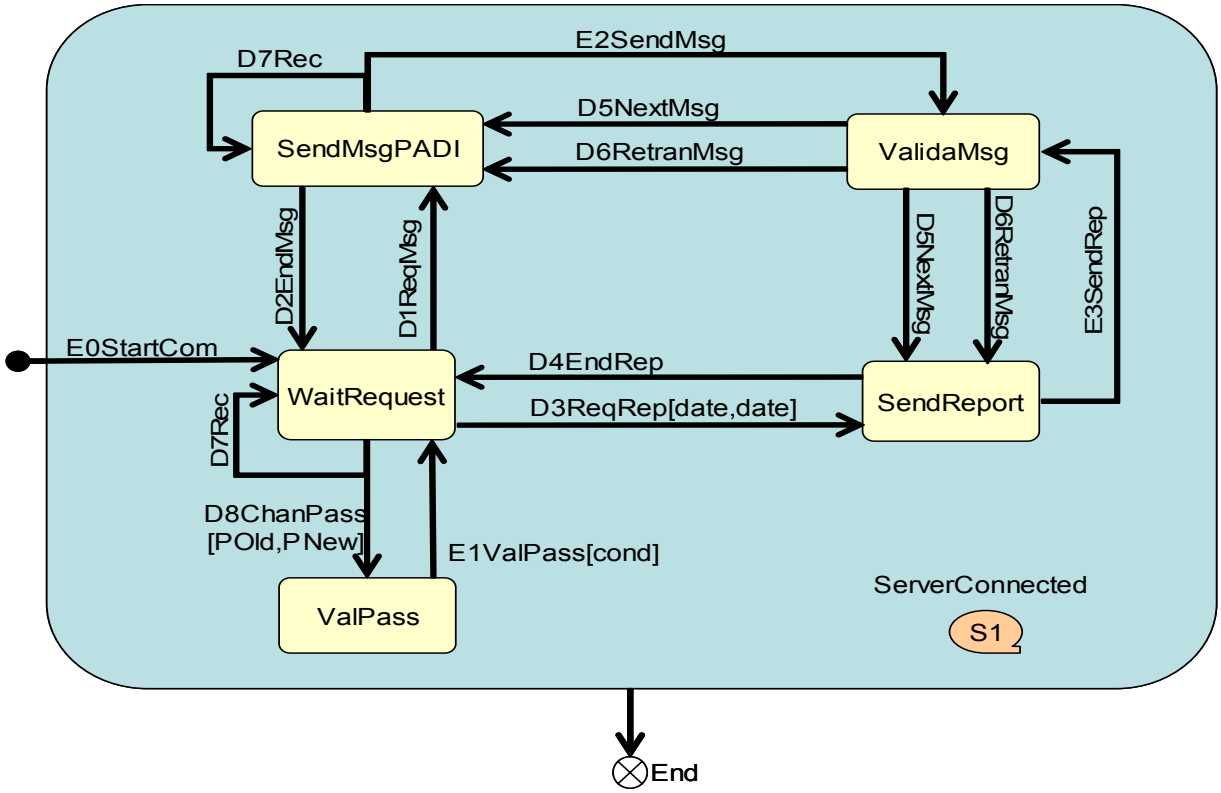

Figura 67 : Máquina de estado servidor: conectado.

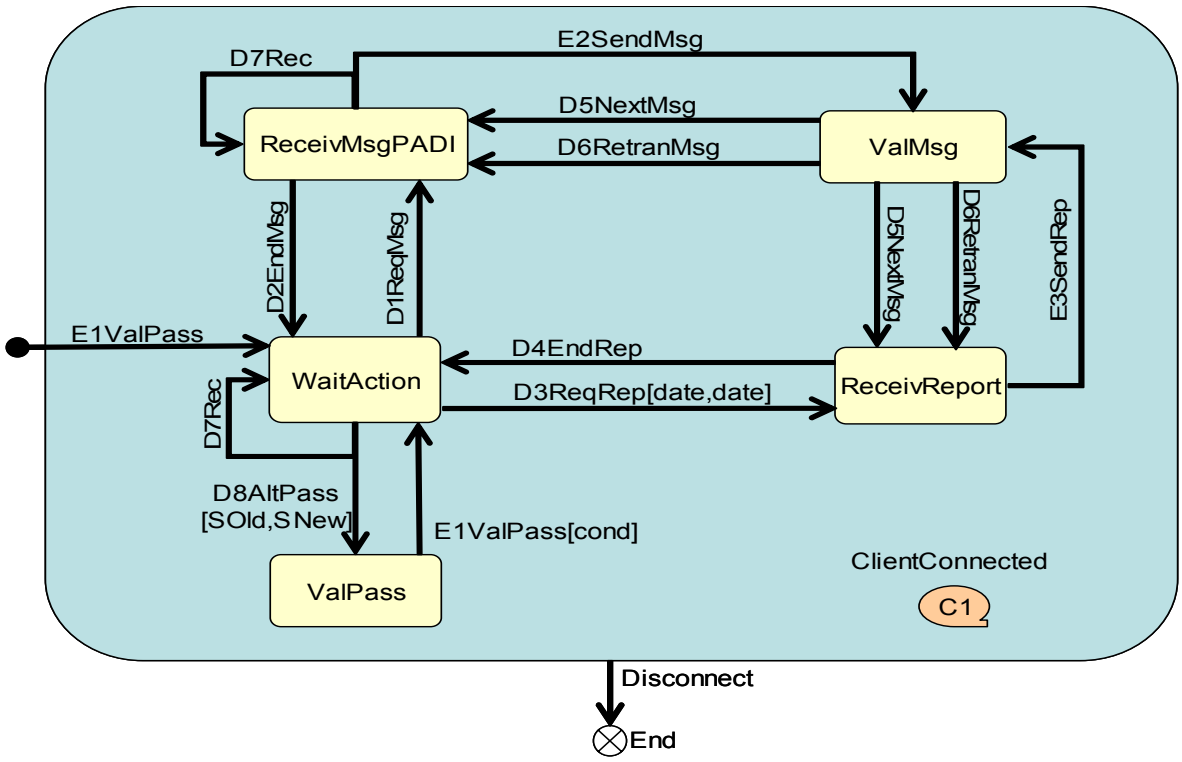

Figura 68 : Máquina de estado cliente: conectado.

A seguir a tabela 2 apresenta as mensagens trocadas entre cliente e servidor.

NOME	SIGNIFICADO	FUNÇÃO
Socket_Connect	Conectar	Inicia uma conexão.
Socket_Disconnection	Desconectar	Finaliza uma conexão.
RequestCnt[port,ip]	Solicita Conexão	Mensagem usada pelo cliente para enviar um pedido de conexão ao servidor. port é a porta onde se deve conectar e ip é o endereço ip do servidor.
FrmSendPass_CS[passw]	Mensagem D0: Envia Senha	Mensagem usada pelo cliente para enviar a senha ao servidor solicitando acesso.
FrmReqPDU_CS	Mensagem D1: Solicita Mensagem	Mensagem do cliente, usada para pedir ao servidor que envie a primeira PDU originada da PADI.
FrmEndPDU_CS	Mensagem D2: Fim de Mensagem	Mensagem usada pelo cliente, solicitando ao servidor que seja encerrada o envio de PDUs da PADI.
FrmReqRep_CS[file,date, date]	Mensagem D3: Solicita Relatório	Mensagem usada pelo cliente para pedir ao servidor que envie a primeira PDU contida no relatório armazenada em arquivo, passando o nome do Arquivo, a data inicial e data final.

FrmEndRep_CS	Mensagem D4: Fim de Relatório	Mensagem usada pelo cliente, solicitando ao servidor que seja encerrada o envio de PDUs do relatório.
FrmNextMsg_CS	Mensagem D5: Solicita Próxima mensagem	Mensagem usada pelo cliente, solicitando ao servidor que envie próxima PDU.
FrmRetranMsg_CS	Mensagem D6: Pedido de retransmissão de Mensagem	Mensagem usada pelo cliente, solicitando ao servidor que retransmita a PDU anterior, pois esta não chegou ou chegou corrompida.
FrmRecPDU_CS	Mensagem D7: Pedido de gravação de mensagens em arquivo	Mensagem usada pelo cliente, solicitando ao servidor que inicie a gravação em arquivo de todas as PDU originadas da PADI
FrmChanPass_CS [POld,PNew]	Mensagem D8: Alteração de senha de acesso	Mensagem usada pelo cliente, solicitando a alteração da senha de acesso ao servidor
FrmInval_CS	Mensagem D9: Mensagem Inválida.	Mensagem usada pelo cliente para indicar ao servidor que a mensagem Recebida é inválida.
FrmFimCnt_CS	Mensagem DAh: Fim da Conexão.	Mensagem usada pelo cliente para indicar ao servidor o fim da conexão.
FrmStartCom_SC	Mensagem E0: Inicia	Mensagem usada pelo

	Cominicação	servidor para iniciar a comunicação.
FrmValPass_SC[cond]	Mensagem E1: Valida Senha	Mensagem usada pelo servidor para validar a senha enviada pelo cliente, “cond” é a condição que pode ser False ou True
FrmSendPDU_SC	Mensagem E2: Envia Mensagem	Mensagem usada pelo servidor para enviar ao cliente as PDUs originadas da PADI
FrmSendREP_SC	Mensagem E3: Envia Mensagem	Mensagem usada pelo servidor para enviar ao cliente as PDUs gravadas em Arquivos.

Tabela 2: Mensagens trocadas entre Cliente e Servidor.

APÊNDICE (B): Diagramas de Seqüência

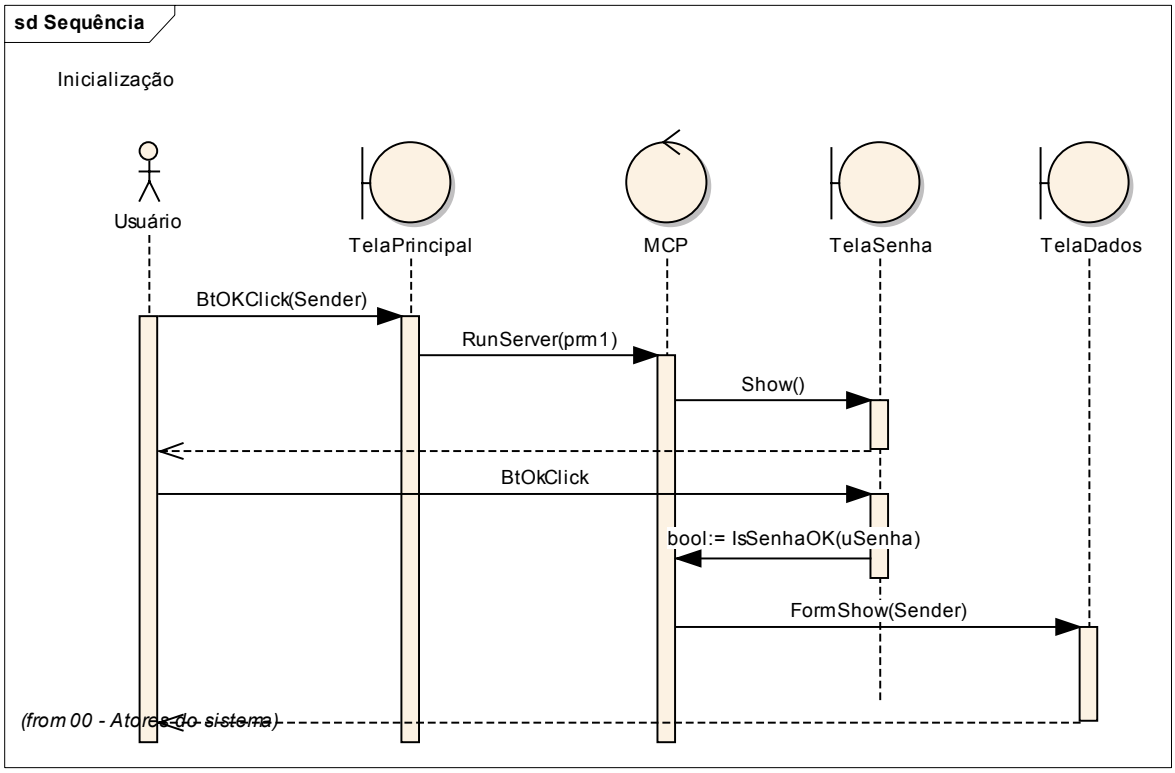

Figura 69 : Diagrama de seqüência de inicialização.

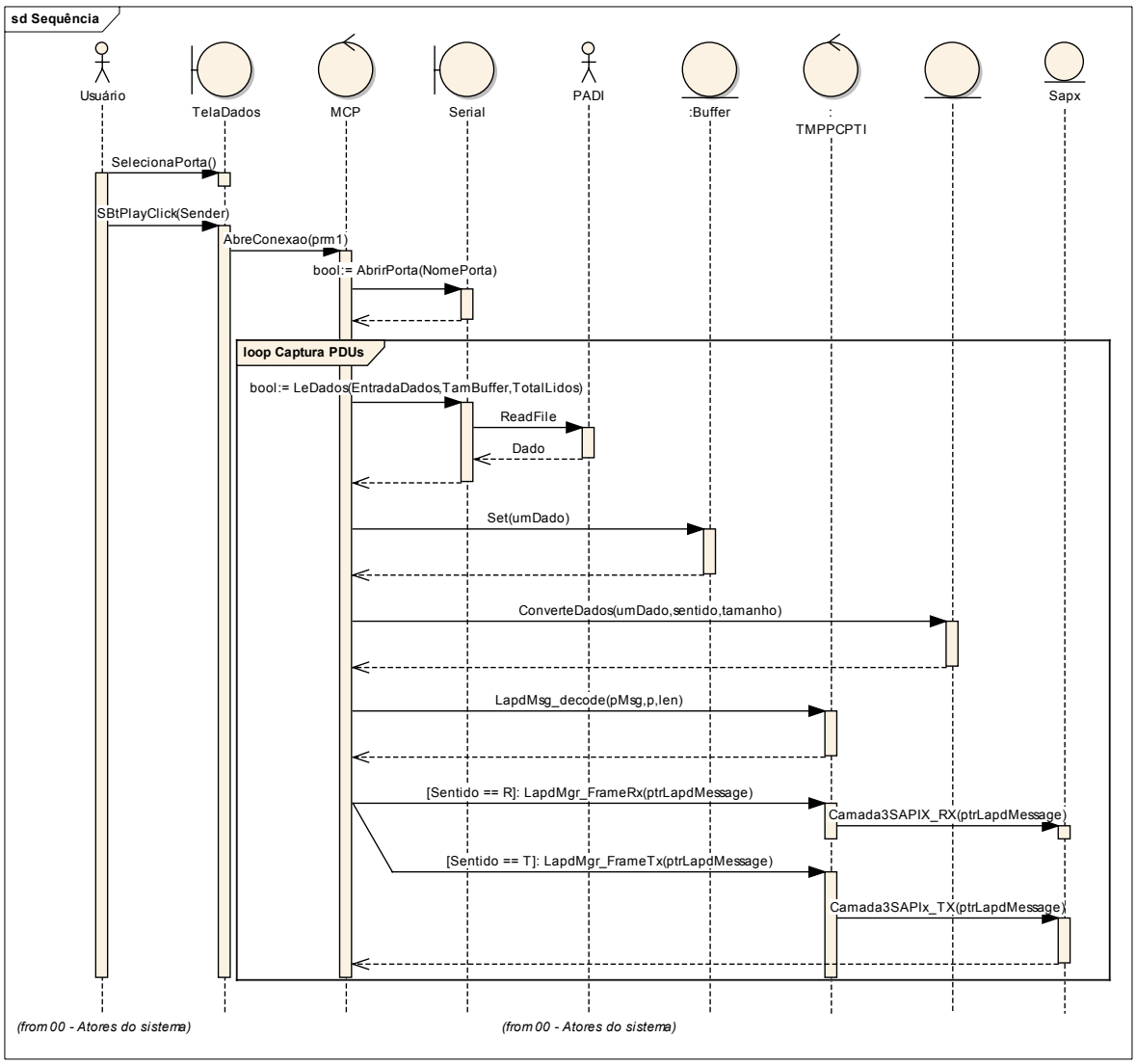

Figura 70 : Diagrama de Seqüência de captura de dados.

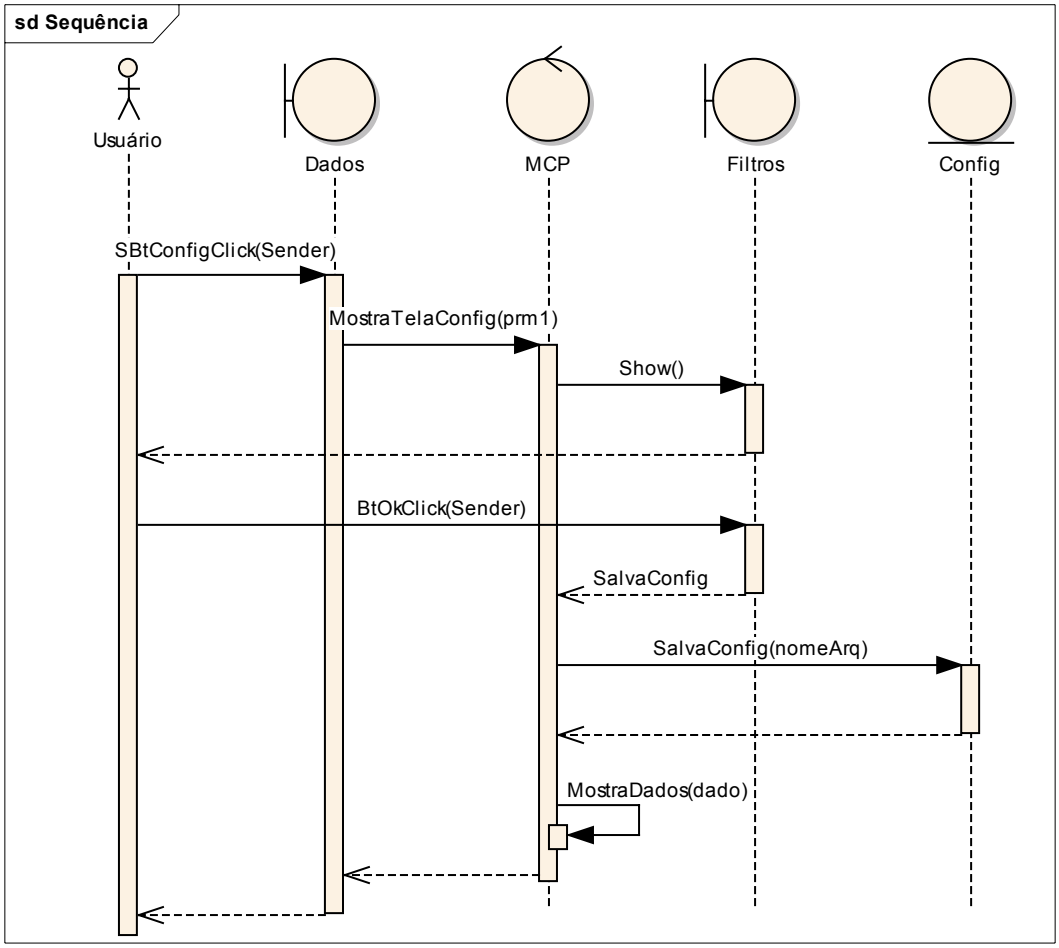

Figura 71 : Diagrama de seqüência de seleção de filtros.

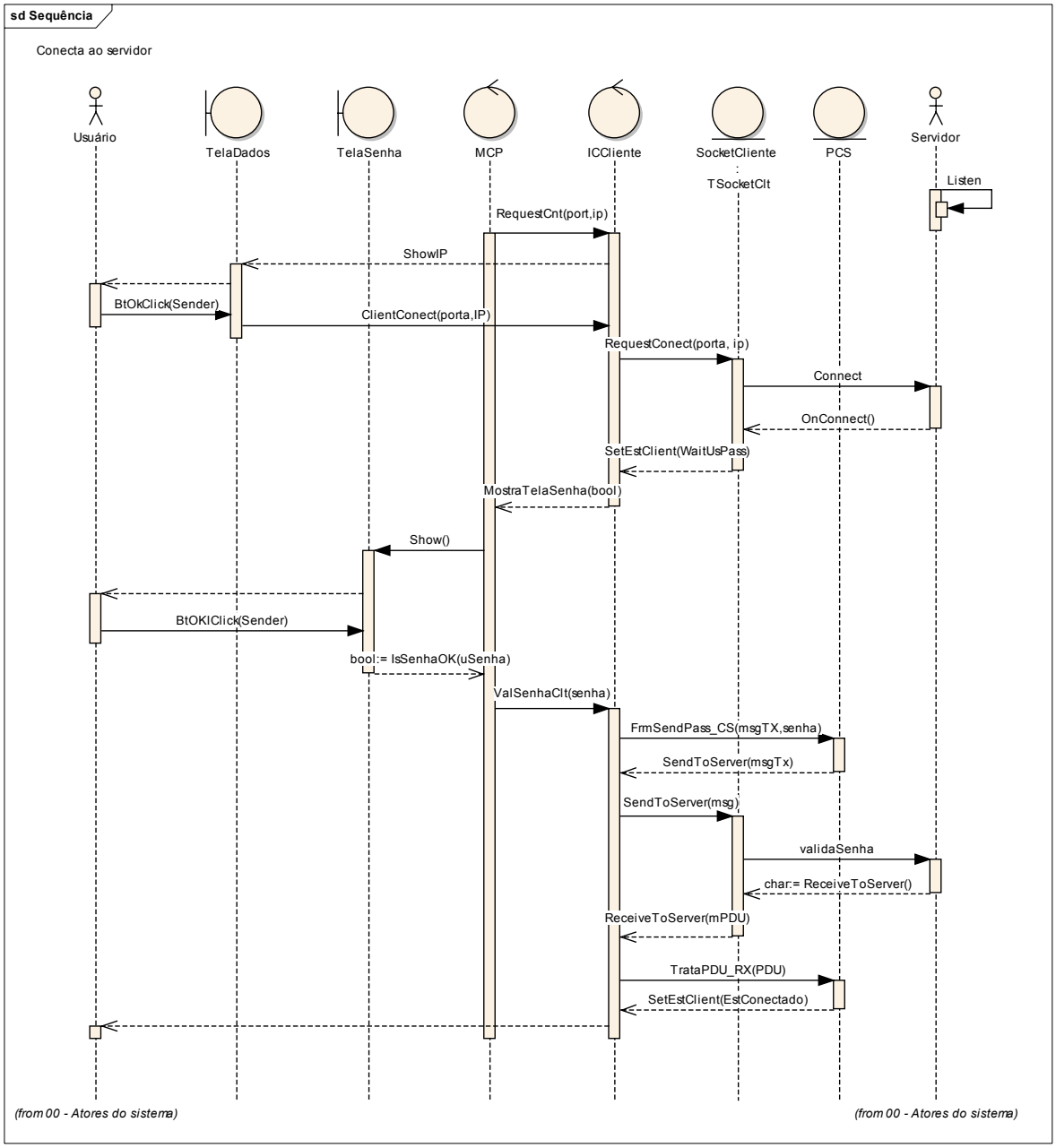

Figura 72 : Diagrama de seqüência de conexão.

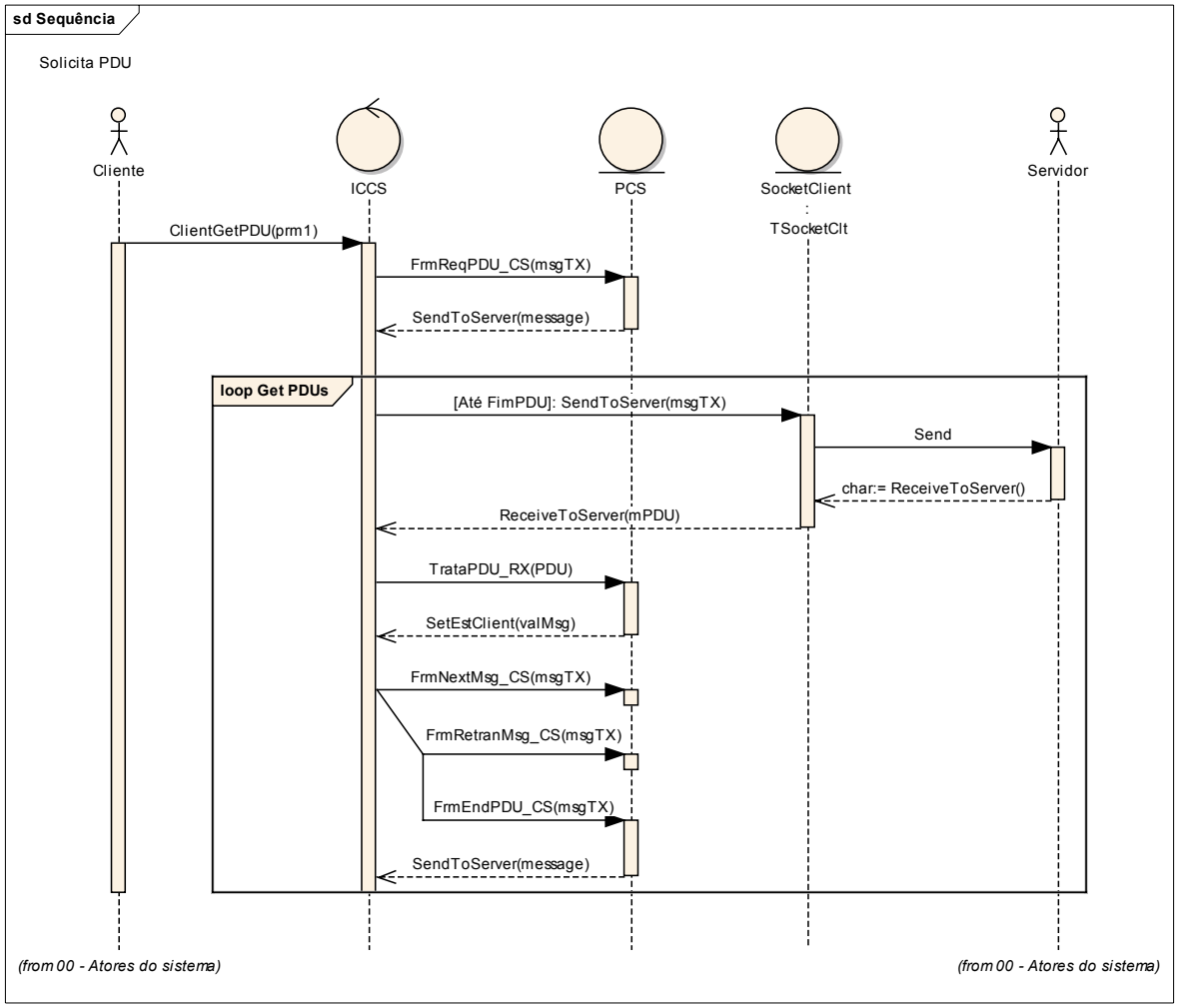

Figura 73 : Diagrama de seqüência de solicitação de PDU.

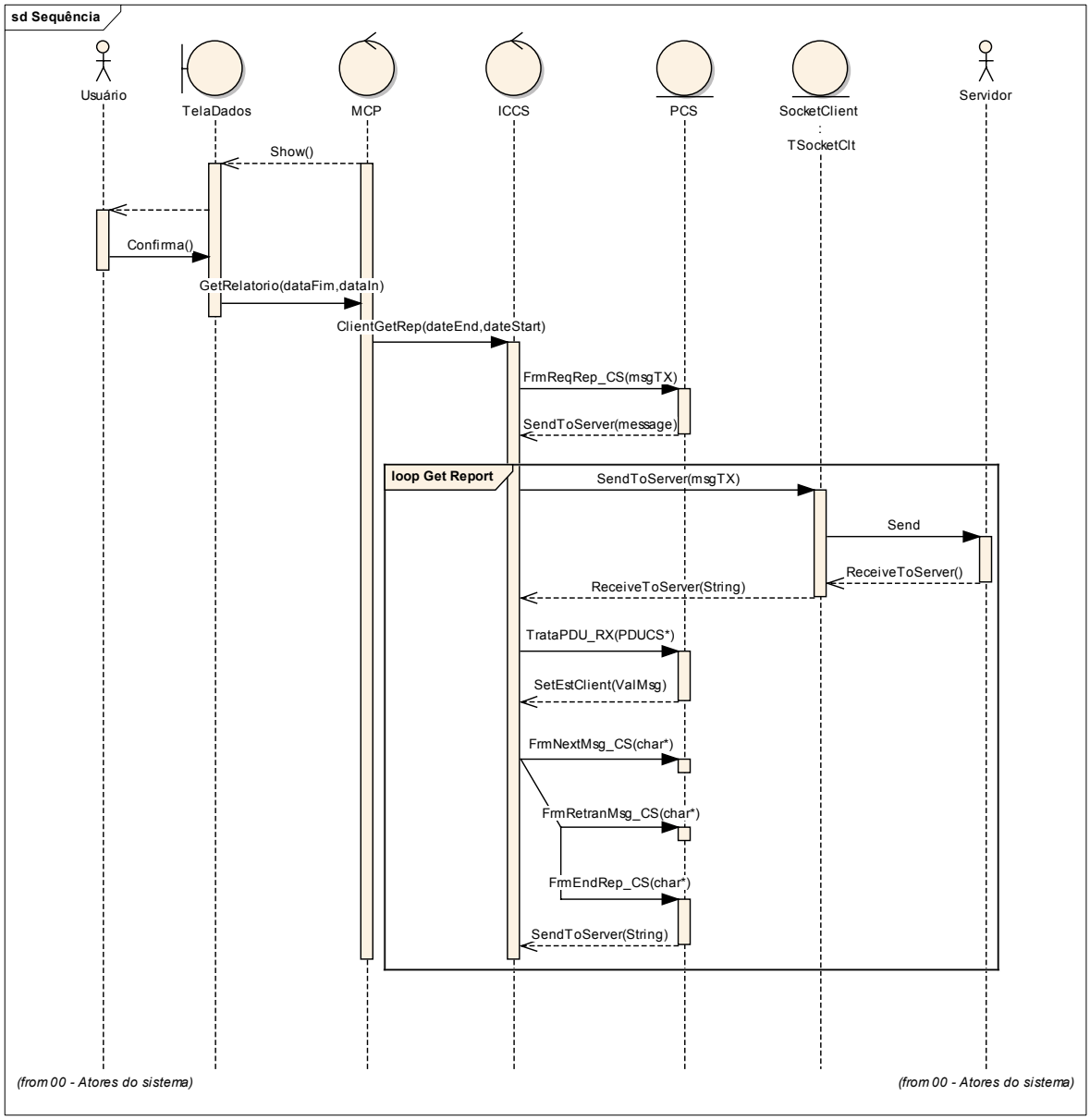

Figura 74 : Diagrama de seqüência de solicitação de relatório.

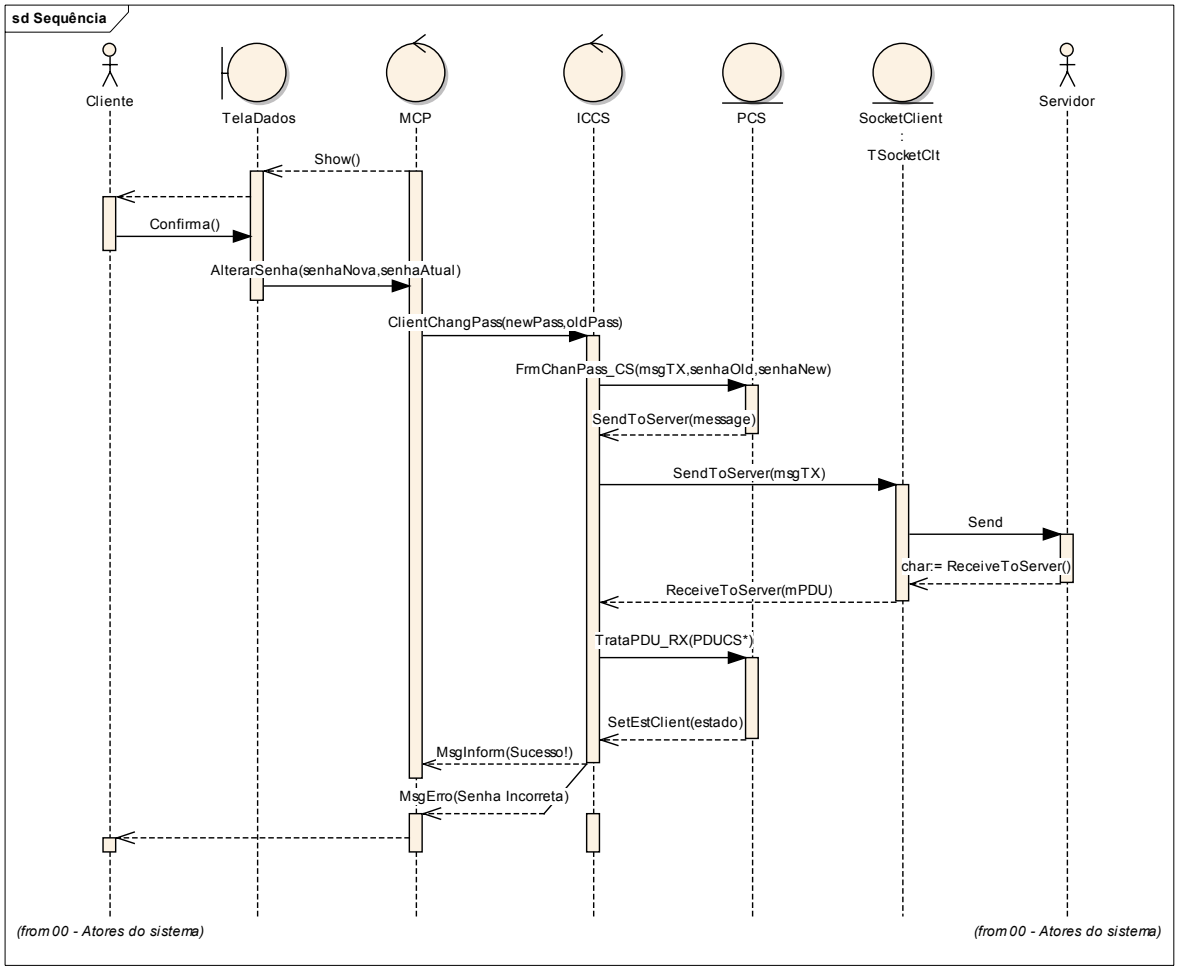

Figura 75 : Diagrama de seqüência de solicitação de alteração de senha.

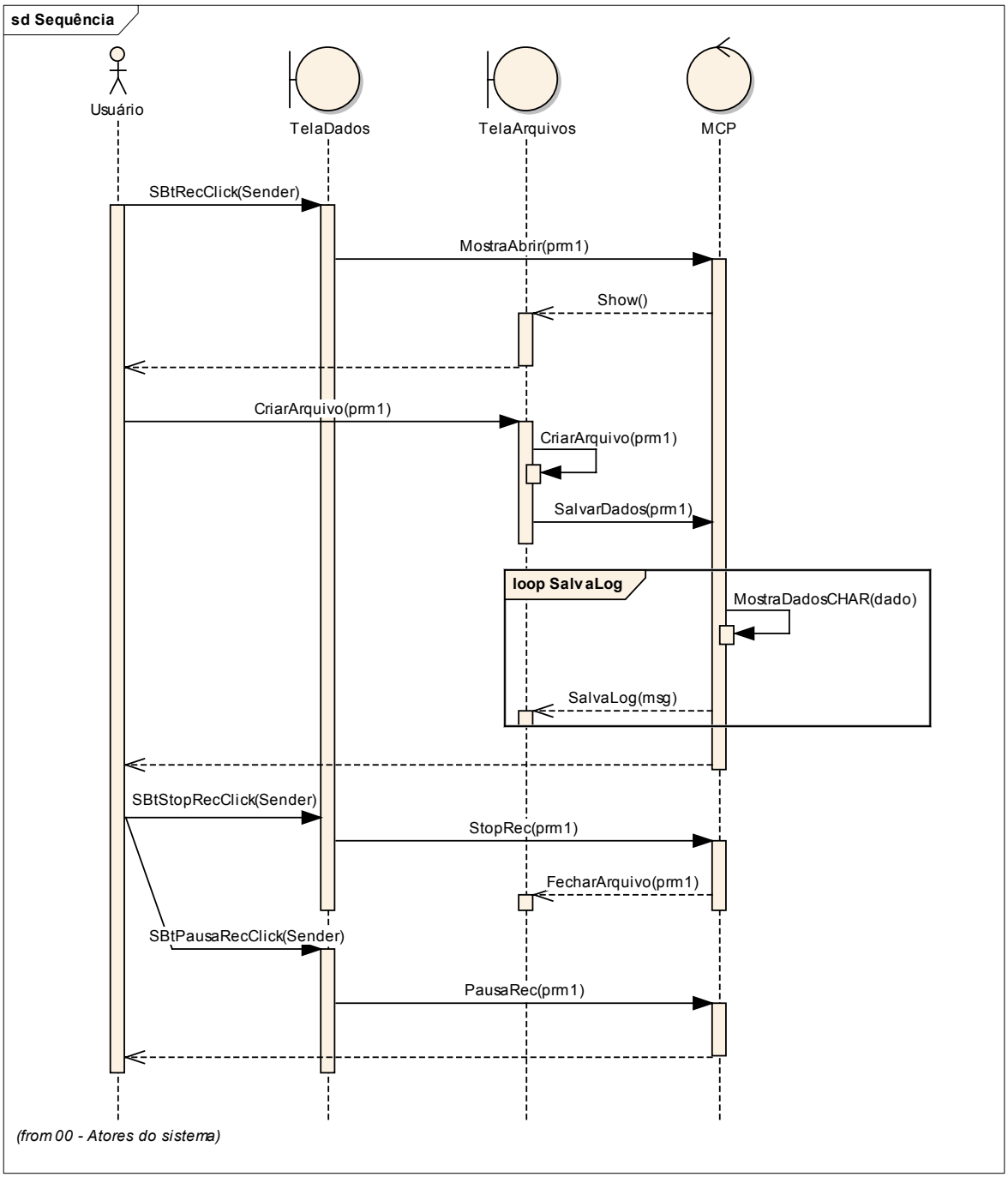

Figura 76 : Diagrama de seqüência de gravação de PDU em arquivo.

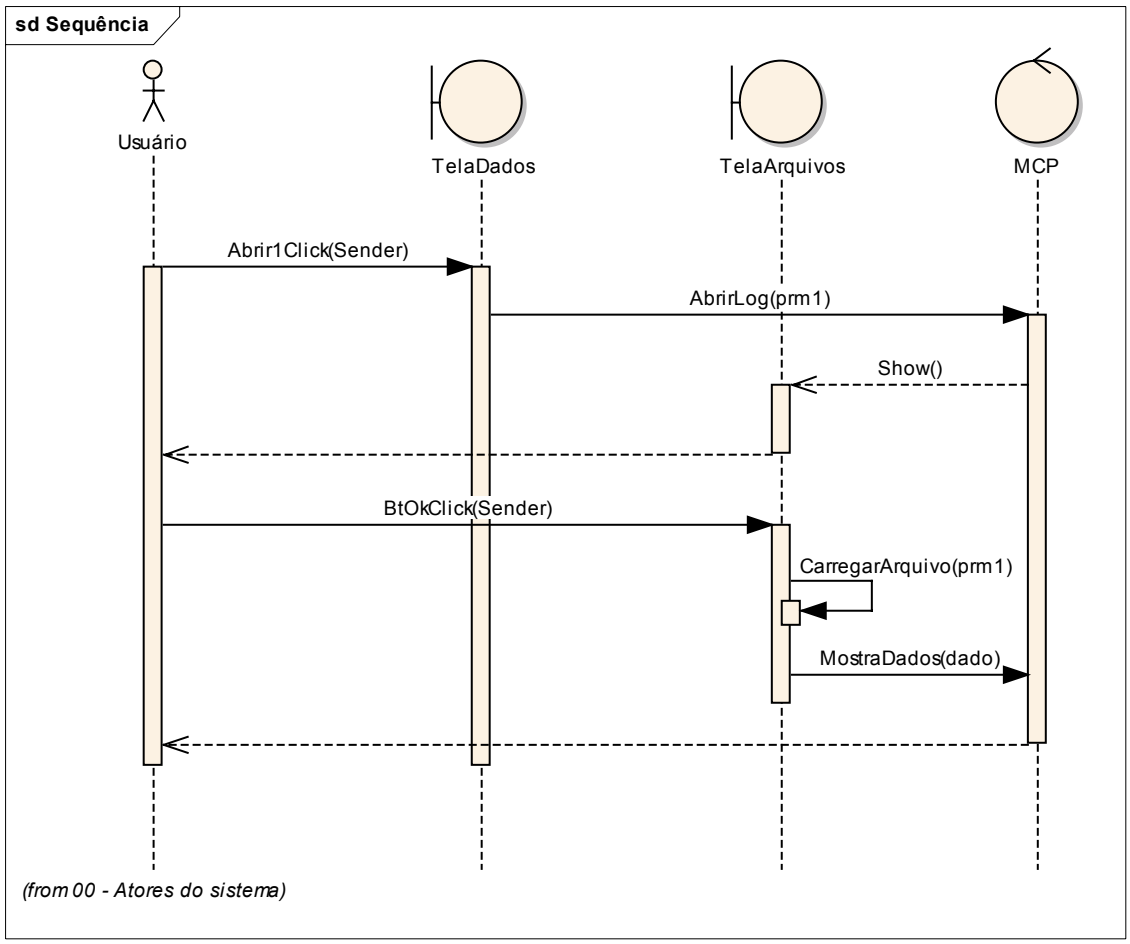

Figura 77 : Diagrama de seqüência carrega log do arquivo.

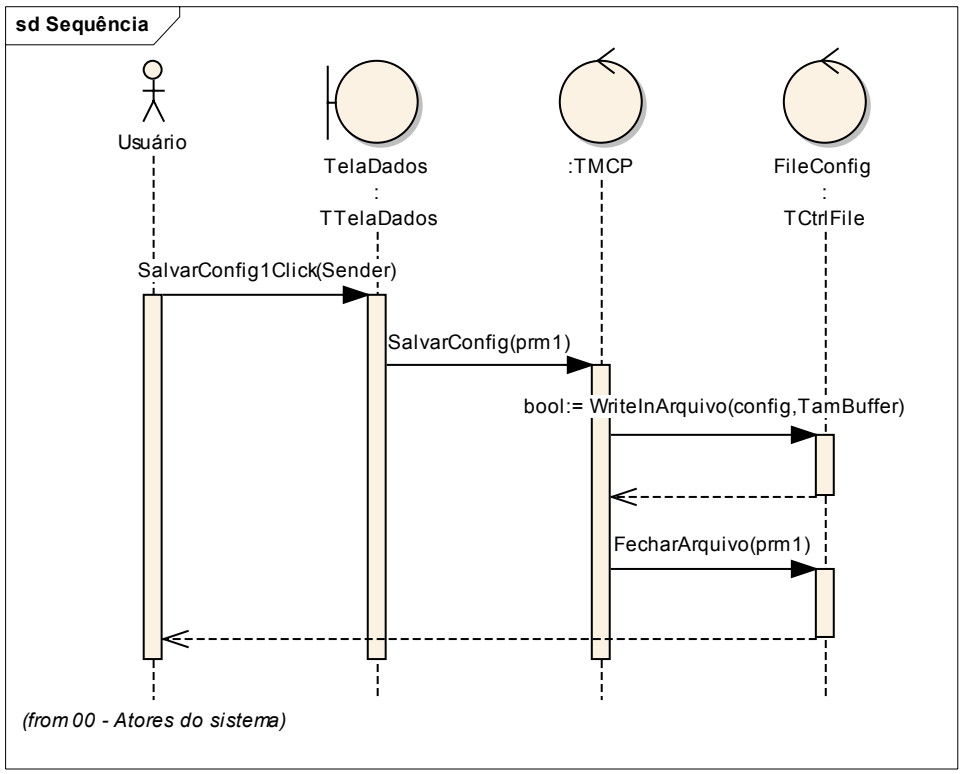

Figura 78 : Diagrama de seqüência salva configurações em arquivo.

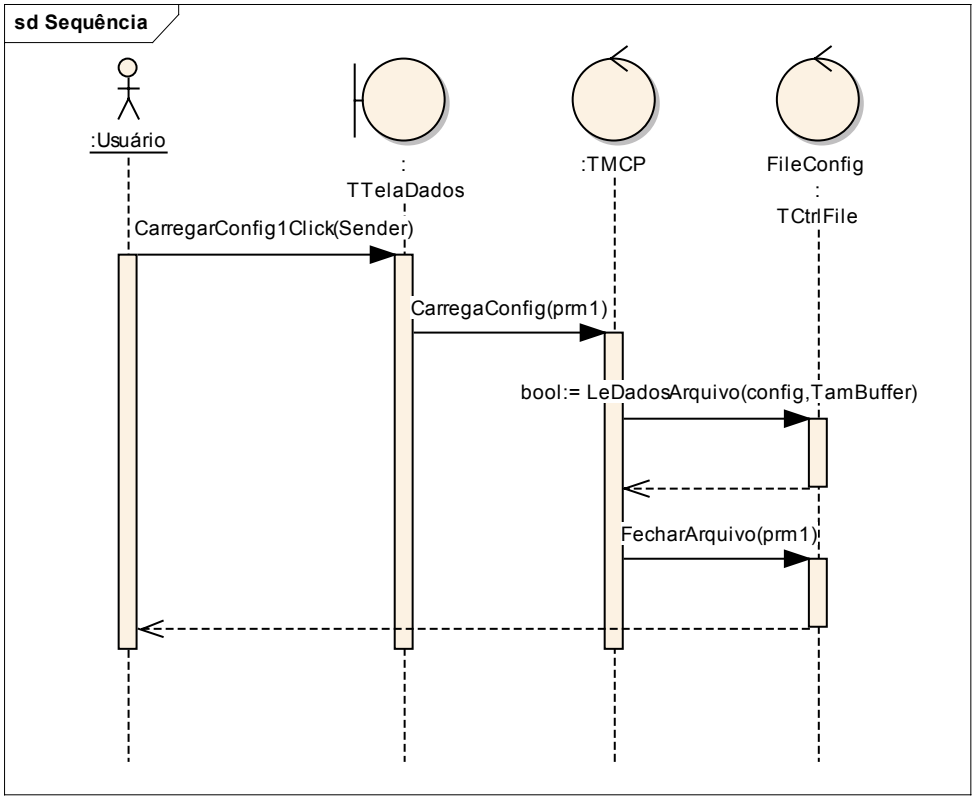

Figura 79 : Diagrama de Seqüência carrega configurações do arquivo.

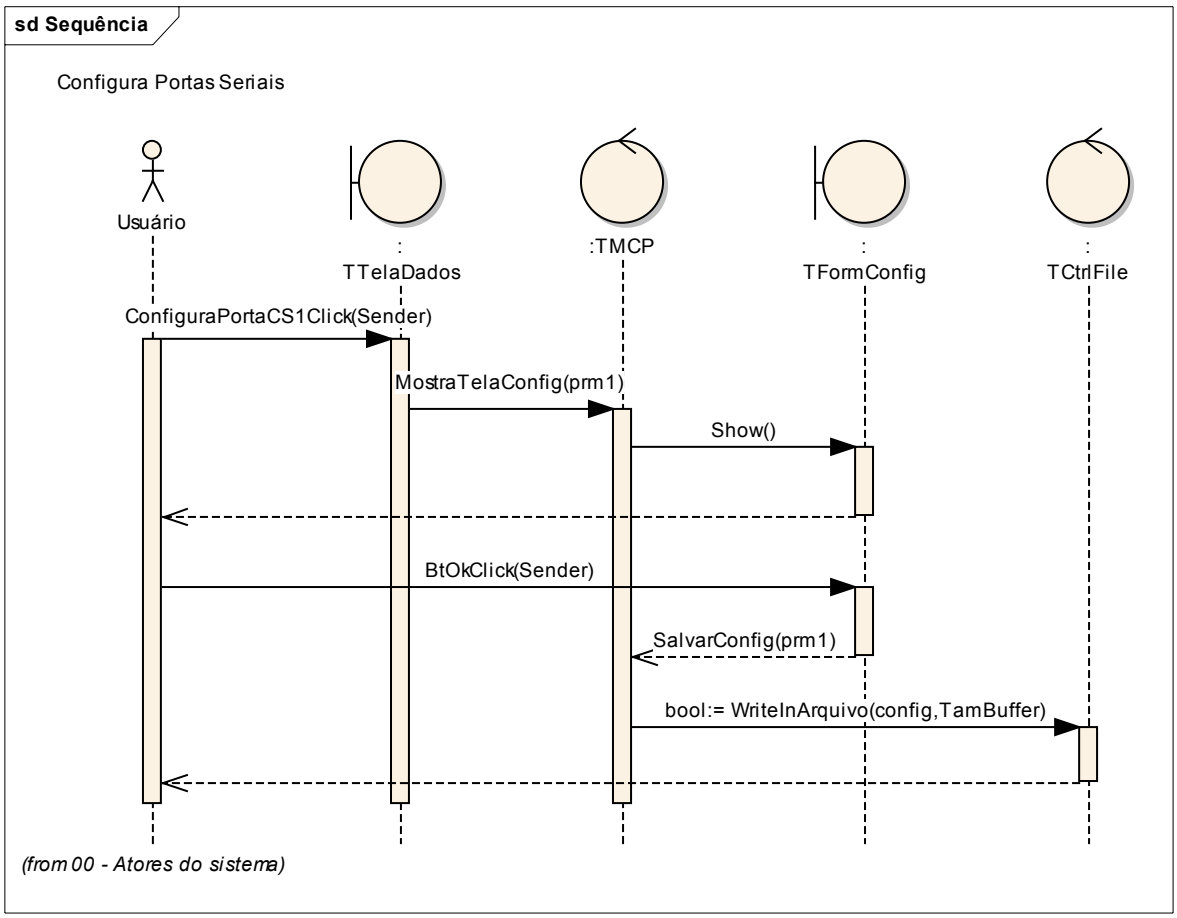

Figura 80 : Diagrama de seqüência de configuração de porta serial.

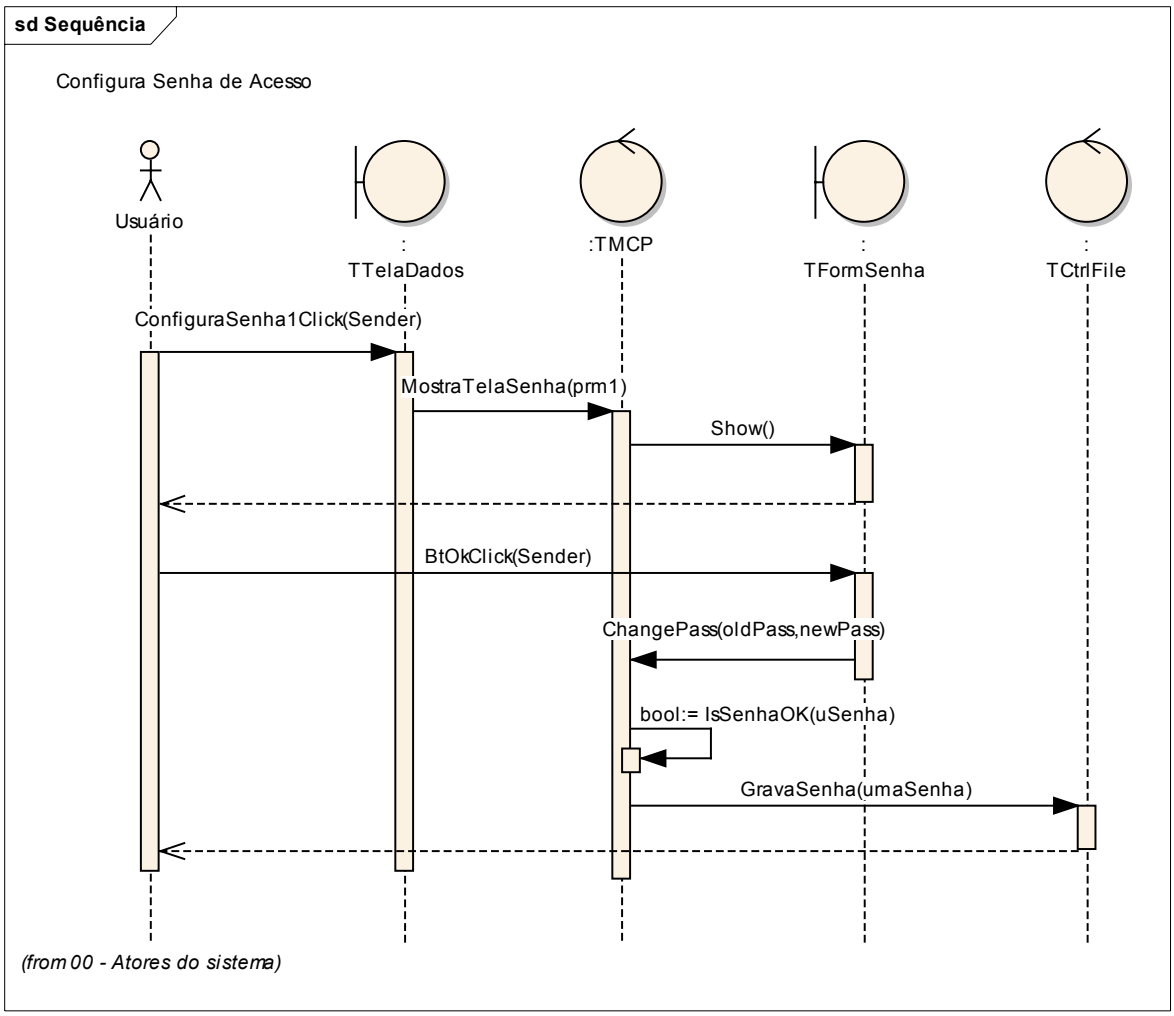

Figura 81 : Diagrama de seqüência de configuração de senha.

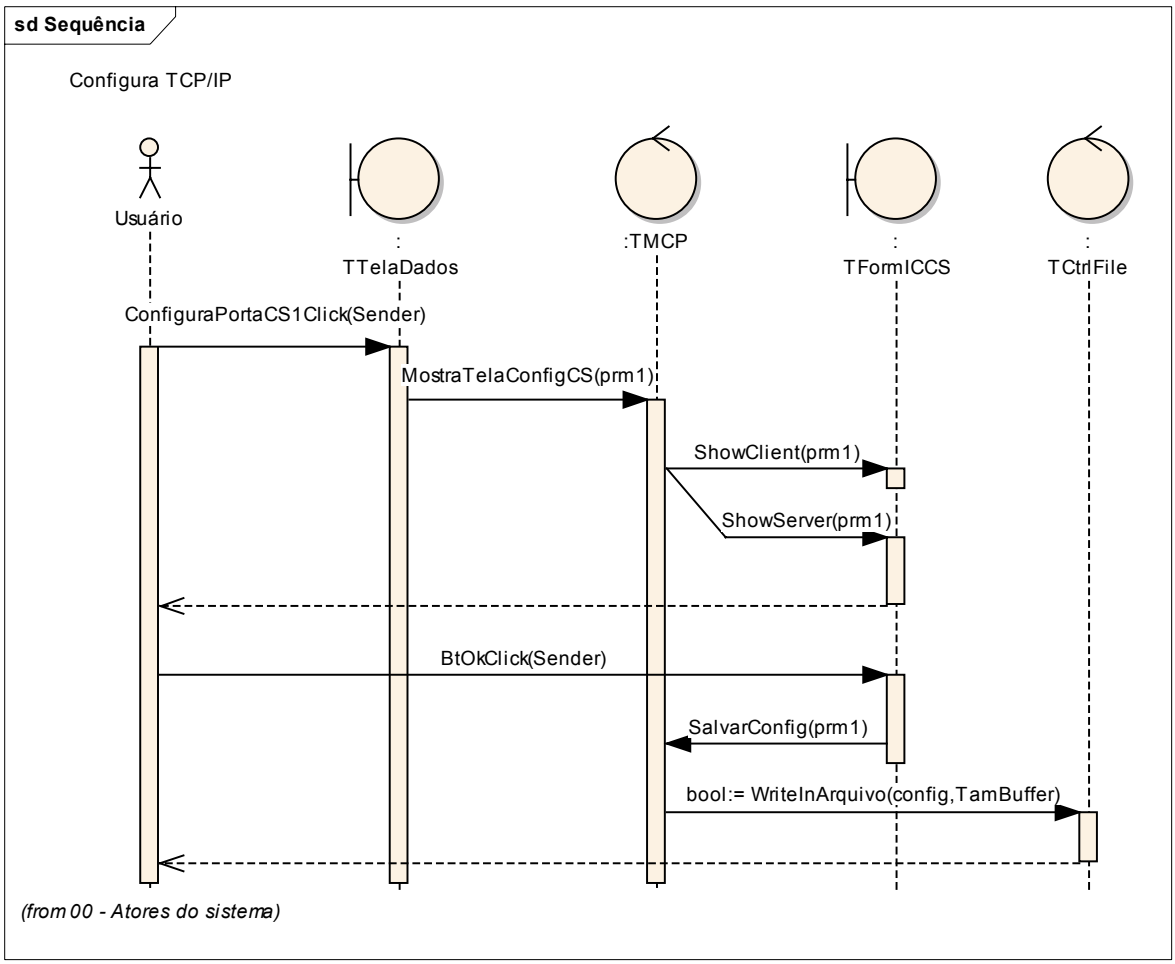

Figura 82 : Diagrama de seqüência de configurações TCP/IP.

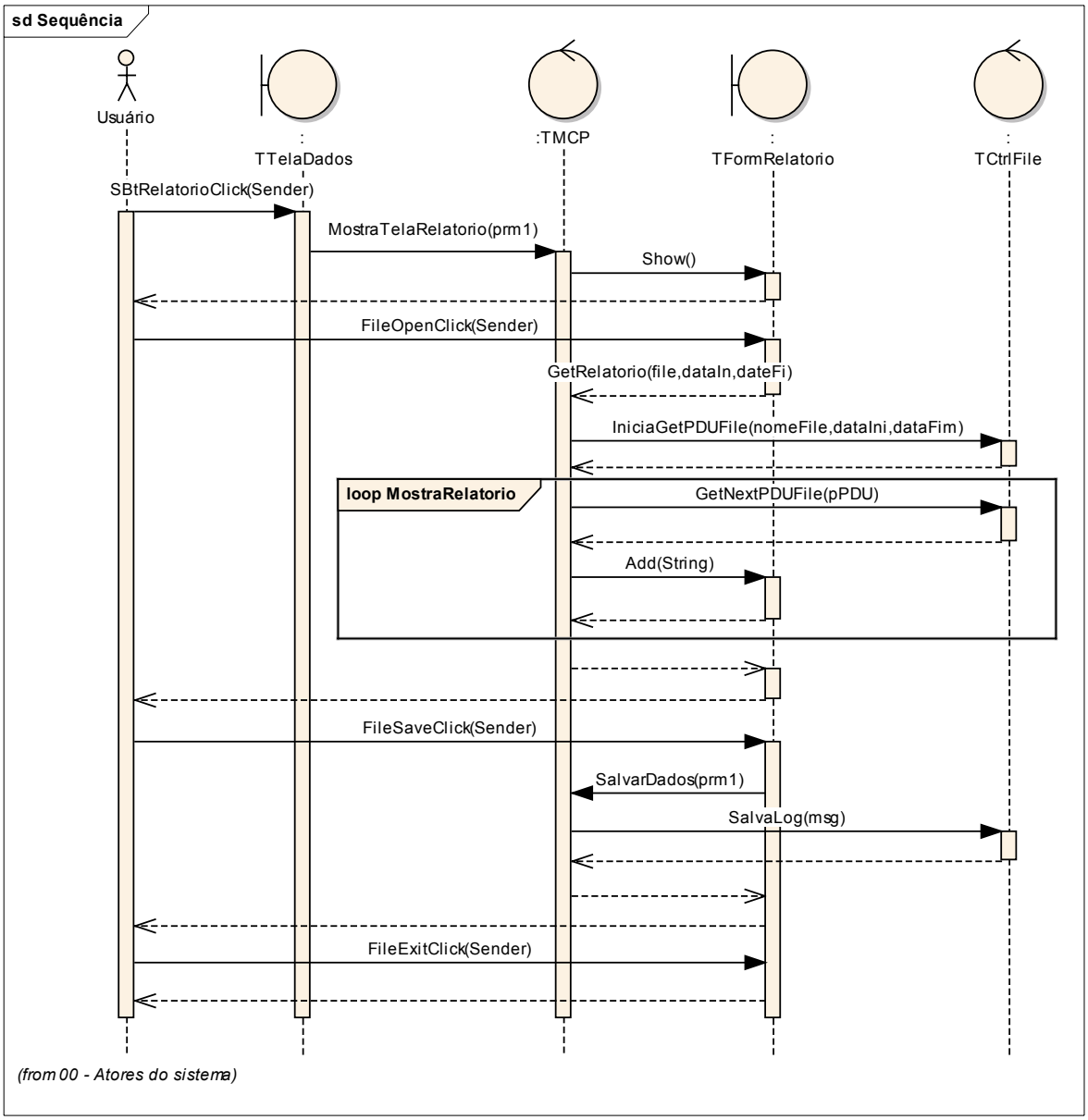

Figura 83 : Diagrama de seqüência do editor de relatório.