



**UNIVERSIDADE FEDERAL DE PERNAMBUCO**  
**CENTRO DE INFORMÁTICA**  
**GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

Um Framework de Telemetria utilizando a rede GSM

**Por**  
**Renato Bibiano de Sá Marques**  
**Trabalho de Graduação em Dispositivos embarcados**  
**e rede GSM**

**Recife**  
**Junho, 2008**

## Assinaturas

Este Trabalho de Graduação representa o esforço do aluno **Renato Bibiano de Sá Marques**, orientado pelo professor **Djamel Fawzi Hadj Sadok**, sob o título “**Um framework de telemetria utilizando a rede GSM**”. Todos abaixo estão de acordo com o conteúdo deste documento e os resultados deste trabalho de graduação.

---

Renato Bibiano de Sá Marques

---

Djamel Fawzi Hadj Sadok

*“Algo só é impossível até que alguém duvide e acabe provando o contrário.”*

Albert Einstein

## **Dedicatória**

Eu dedico este trabalho a Deus, à minha família e a todos os que conviveram comigo este período de graduação.

## Agradecimentos

Eu gostaria de agradecer a todos que me ajudaram e que acreditaram que eu poderia chegar aonde eu cheguei.

Gostaria de agradecer em primeiro lugar à minha mãe, Risoleta Bibiana de Sá Marques, e ao meu pai, Marcos Pires de Sá Marques, sem eles nunca chegaria tão longe, eles que me deram todo o suporte, companhia, educação, amor e saúde.

Aos meus companheiros de turma de Engenharia da Computação 2003-2, a melhor turma da qual eu poderia fazer parte, assim como à turma de Ciências da Computação do mesmo período.

Às pessoas que fazem parte do GPRT (Grupo de Redes de Pesquisa e Telecomunicações), em especial a Djamel Fawzi Hadj Sadok, meu orientador, que me ajudou neste trabalho, a Judith Kelner, professor responsável do GPRT que também me deu muito apoio desde o momento em que ingressei ao grupo, e a Luis Eduardo Melo Corrêa de Oliveira, meu gerente e amigo no grupo, que me ajudou em várias etapas do projeto.

Por fim, gostaria de fazer um agradecimento especial a Mayara Marques de Oliveira e Silva e sua família por estarem ao meu lado durante grande parte desta jornada.

## Resumo

Este trabalho consiste no estudo e desenvolvimento de um *framework* de telemetria utilizando a tecnologia GSM/GPRS. O projeto é dividido em dois módulos distintos. O primeiro consiste em uma aplicação embarcada escrita em JME® a ser executada num módulo TC65 da *Siemens*®. Essa aplicação terá a responsabilidade de se comunicar com os dispositivos e/ou equipamentos a serem monitorados disponibilizando as informações coletadas através de comunicação serial e/ou GPRS. O segundo módulo por sua vez trata-se de uma aplicação *desktop* implementada em Java® e tem como principal função gerenciar as informações enviadas pela aplicação embarcada e efetuar possíveis configurações na mesma.

Outras tecnologias consideradas neste *framework* são: o envio de mensagens SMS, permitindo aos usuários o recebimento de alertas; e alarmes e a administração remota das funções do módulo através de GPRS, e OTAP, que possibilita a atualização remota da aplicação embarcada do módulo através da rede de telefonia celular.

Como estudo de caso, foi desenvolvido um protótipo para monitoramento de *Nobreaks*. Entretanto o *framework* desenvolvido foi construído de forma genérica e com interfaces bem definidas favorecendo sua adaptação futura a qualquer aplicação de telemetria, não estando seu uso restrito apenas a leitura de informações de *nobreaks*.

**Palavras-chave:** GSM, J2ME, Java, GPRS.

## **Abstract**

This work consists of a study and development of a telemetry framework using the GSM/GPRS technology. The project is divided in two distinct modules. The first consists of an embedded application developed in JME® to be executed on a TC65 module from *Siemens*®. That application is responsible for the communications with the devices and/or equipments to be tracked, providing the collected information from the serial and/or GPRS communication interface. The second module, in turn, is a desktop application, implemented in Java®. Its main functions are to manage the information sent by the embedded application and to make possible remote configurations.

Other technologies developed on this work are the SMS exchange, allowing the users to receive alarms and warnings and to remotely manage the module functions through GPRS, and OTAP, which allows the remotely embedded module application update through the mobile phone network.

As a case study, our system was used to track nobreaks (Uninterruptable Power Supply or UPS) data. However, the developed framework was built in a generic way and with well defined interfaces, favoring their adaptation to any future telemetry application, not letting its use being restricted to information reading from nobreaks.

**Keywords:** GSM, J2ME, Java, GPRS.

# Sumário

Assinaturas .....	2
Dedicatória.....	4
Agradecimentos.....	5
Resumo .....	6
Abstract.....	7
Sumário.....	8
Lista de figuras .....	<a href="#">9</a>
Lista de tabelas .....	10
1. Introdução.....	11
1.1. Motivação .....	12
1.2. Objetivos.....	13
1.3. Organização do documento .....	14
2. Trabalhos Relacionados.....	15
3. Arquitetura e Tecnologias para Monitoramento Remoto .....	18
3.1. Arquitetura do <i>Framework</i> .....	18
3.2. Módulo TC65 .....	20
3.3. SMS .....	24
3.4. GPRS .....	26
3.5. OTAP.....	28
4. Desenvolvimento do <i>Framework</i> para controle remoto .....	30
4.1. Plataforma Embarcada.....	32
4.1.1. Timer .....	32
4.1.2. Timer do log .....	33
4.1.3. GPRS .....	33
4.1.4. OTAP.....	33
4.1.5. Serial.....	34
4.1.6. Mensagens SMS .....	34
4.2. Software de Controle .....	35
4.2.1. Tela Configurações Gerais .....	37
4.2.2. Tela Configurações da Serial.....	38
4.2.3. Tela Configurações GPRS.....	39
4.2.4. Tela Configurações OTAP .....	41
4.2.5. Tela de Configuração de Alarmes .....	41
4.2.6. Tela Configurar Usuários .....	43
4.2.7. Habilitando e desabilitando o Autostart .....	44
4.2.8. Terminal .....	44
4.2.9. Conexões GPRS e Serial .....	46
4.2.10. Enviar Arquivo de Configuração ao modem.....	47
5. Conclusão .....	49
5.1. Dificuldades Encontradas .....	49
5.2. Trabalhos Futuros .....	50
Referências .....	52
Apêndice A – Comandos AT.....	54
Apêndice B – Arquivo de Configuração .....	57
Apêndice C - Interfaces .....	60

## Lista de figuras

Figura 1. Arquitetura do <i>Framework</i> .....	18
Figura 2. Configuração de um cabo ideal para o sistema.....	21
Figura 3. Módulo TC65 (vista superior traseira).....	22
Figura 4. Módulo TC65 (vista superior frontal).....	22
Figura 5. Modem com entrada de alimentação e antena GPRS.....	22
Figura 6. Tensão negativa à esquerda, tensão positiva a direita.....	23
Figura 7. Resumo esquemático do módulo TC65.....	24
Figura 8. Antena GPRS.....	26
Figura 9. Pilha de protocolos GPRS.....	27
Figura 10. OTAP no TC65.....	28
Figura 11. Arquitetura do sistema.....	30
Figura 12. Tela inicial do Software de Controle.....	37
Figura 13. Tela Configurações Gerais.....	38
Figura 14. Tela Configurações da Serial.....	39
Figura 15. Tela Configurações GPRS.....	39
Figura 16. Abrindo a tela de log.....	40
Figura 17. Tela Configurações OTAP.....	41
Figura 18. Tela de Configuração de Alarmes.....	42
Figura 19. Configurando o programa para <i>nobreaks</i> trifásicos.....	42
Figura 20. Abas de alarmes para <i>nobreaks</i> trifásicos.....	43
Figura 21. Tela Configurações de Usuários.....	43
Figura 22. Habilitar ou Desabilitar o <i>Autostart</i> .....	44
Figura 23. Terminal para controle remoto via GPRS.....	45
Figura 24. Tela para funcionamento de Ponte do modem.....	46
Figura 25. Imagens do ícone de conexão GPRS.....	47
Figura 26. Enviando um novo Arquivo de Configuração ao modem.....	48
Figura 27. Imagens do ícone de alterações no Arquivo de Configuração.....	48
Figura 28. Esquemático de um cabo com 9 pinos paralelos.....	61

## Lista de tabelas

Tabela 1. Classes GPRS .....	27
Tabela 2. Mensagens SMS aceitas pelo modem. ....	35
Tabela 3. Atributos para conexão GPRS de cada operadora. ....	40
Tabela 4. Descrição do parâmetro para o comando AT+ICF.....	55
Tabela 5. Descrição do parâmetro para o comando AT\Q .....	56

# 1. Introdução

O constante crescimento das redes de telefonia móveis no mundo atual permite que cada vez mais aplicações sejam desenvolvidas utilizando a tecnologia GSM<sup>1</sup>/GPRS<sup>2</sup>. Segundo [1], GSM é a tecnologia de comunicação com maior crescimento de todos os tempos, sendo utilizada por 29% da população mundial. Esse crescimento faz com que todas as tecnologias associadas a ela também cresçam, na mesma proporção. Hoje em dia, quase todos os celulares já podem acessar a internet graças à tecnologia GPRS, por exemplo, e, com o aumento de seu uso, a tendência é diminuir ainda mais o seu custo.

Fazendo uma análise comparativa, de acordo com [3], enquanto as taxas de transferência de tecnologias anteriores eram de 12Kbps, a do GPRS pode ultrapassar a marca dos 170Kbps, sendo que na prática chega-se apenas a 64Kbps. Os custos de troca de dados são menores, pois com a tecnologia GPRS as tarifas são cobradas pela quantidade de dados transmitidos, antes a cobrança se dava pelo tempo de conexão.

Em [2] o autor é um pouco mais inflexível e informa que o padrão GPRS pode substituir por completo as demais tecnologias atuais, CSD<sup>3</sup> e SMS<sup>4</sup>, pois tem várias vantagens sobre as mesmas. Também segundo [2], a tecnologia GPRS facilita conexões instantâneas e novas aplicações através da rede GSM.

Com GPRS, a informação é dividida em pacotes, sendo eles comutados e enviados, um processo idêntico ao utilizado pelos protocolos da internet, o que possibilita o acesso à internet através de GPRS. Essa comutação não significa que a troca de dados vá ocupar o canal de comunicação por inteiro, com GPRS o recurso será compartilhado concorrentemente entre vários usuários.

Com o crescimento da rede GSM, abrem-se portas para o desenvolvimento de aplicações embarcadas em celulares ou dispositivos que a utilizem. Entre todas as linguagens possíveis para programar tais aplicações, uma que merece destaque é a plataforma J2ME<sup>5</sup>, sendo a mais utilizada no mundo.

---

<sup>1</sup> GSM (*Global System for Mobile Communications*) – Sistema global para comunicações móveis, é o padrão de celular mais usado no mundo.

<sup>2</sup> GPRS (*General Packet Radio Service*) – Tecnologia utilizada na rede GSM que aumenta a taxa de transferência de dados.

<sup>3</sup> CSD (*Circuit Switched Data*) – Tecnologia de troca de dados.

<sup>4</sup> SMS (*Short Message Service*) – Protocolo de troca de mensagens.

<sup>5</sup> J2ME(*Java 2, Micro Edition*) – Plataforma Java para dispositivos com recursos de hardware e software limitados.

As tecnologias JME® contêm um JRE<sup>6</sup> bastante aperfeiçoado e especialmente desenvolvido para o mercado de grande consumo. Essas tecnologias abrangem uma ampla gama de produtos com tamanhos reduzidos e habilitam programas úteis, de segurança e conectividade em *smart cards*, *paggers*, conversores de sinal digital, celulares e outros aparelhos de pequeno porte. Trabalhos como [4] e [5] mostram que a tecnologia JME é bastante utilizada e está difundida nos dispositivos eletrônicos.

O sistema aqui apresentado trata de um *framework* de telemetria que utiliza as tecnologias anteriormente citadas para realizar o monitoramento e controle de dispositivos eletrônicos, no nosso estudo de caso, *nobreaks*. O trabalho é bastante flexível, podendo ser direcionado facilmente para outros fins, de acordo com a necessidade de cada projeto.

Um sistema de telemetria é um sistema de monitoramento com diversas aplicações, que normalmente funcionam via transmissão sem fio (sinal à rádio), telefonia ou satélite. Sistemas sem fio são bastante utilizados para o monitoramento de ambientes, recolhimentos de dados meteorológicos, em vários esportes automobilísticos (inclusive na fórmula 1), entre outros.

## 1.1. Motivação

Sistemas de telemetria estão cada vez mais presentes em empresas, estabelecimentos, lojas, etc. Os seus gerentes buscam uma maior segurança para seu negócio e também de estarem cientes de tudo que acontece no ambiente. Às vezes essa vontade de monitoramento constante não pode ser suprida com os recursos que esses têm em mãos (funcionários ou equipamentos), e é por isso que cada vez mais estes usuários ou simples proprietários adquirem sistemas de telemetria para serem implantados em seu ambiente.

O monitoramento de ambientes não é o único tipo de sistema de telemetria existente. Em alguns casos, almeja-se o monitoramento de um equipamento eletro-eletrônico específico, como é o caso dos *nobreaks* ou UPS (*Uninterruptible Power Supply*). O que se deseja neste caso é ser informado caso o equipamento esteja funcionando de modo incorreto ou fora de funcionamento, para que se possam ser tomadas as providências necessárias. No caso de um *nobreak* ou de um gerador, geralmente eles estão

---

<sup>6</sup> JRE(Java Run-Time Environment) – Ambiente de execução Java.

fornecendo energia para algo importante onde mesmo que haja uma perda de energia, o equipamento necessita continuar funcionando.

Os sistemas de telemetria modernos estão usando cada vez mais as tecnologias da rede GSM em suas aplicações. Há alguns anos, a grande maioria dos sistemas fazia o monitoramento através da internet e o usuário precisava navegar na *web* para obter as informações desejadas e os alarmes só poderiam ser enviados por e-mail. Com GSM, tudo pode ser feito através do celular, facilitando a vida dos usuários e tornando o processo mais rápido e flexível, sem contar que o usuário pode se comunicar com o sistema de qualquer lugar onde haja cobertura da rede GSM.

A tendência é que todos os sistemas adotem este tipo de comunicação, pois com a modernização dos aparelhos celulares (já há celulares com navegadores *web* integrados), não se fará necessário um computador com internet para acessar os dados e receber alarmes sobre o ambiente ou equipamento monitorado.

## 1.2. Objetivos

Os principais objetivos deste trabalho são:

- Desenvolver uma aplicação em JME®, a ser executada num módulo TC65 da Siemens®, que se comunicará com um *nobreak*, obtendo seu status e monitorando o seu funcionamento. O módulo, ou modem, enviará informações sobre possíveis falhas no equipamento a usuários que forem feitos responsáveis por ele. Inicialmente a aplicação terá funcionalidades básicas de comunicação e envio de alarmes;
- Criar um sistema que se comunicará com o módulo, para um acesso remoto ao mesmo, permitindo a um usuário definir padrões e perfis de usuários para a comunicação e monitoramento;
- Desenvolvimento do formato ou protocolo de comunicação SMS com o modem, através do qual o usuário poderá fazer requisições ao mesmo;
- Utilização da tecnologia GPRS para troca de dados entre o módulo e o software de monitoramento a distância;
- Implantação do serviço OTAP (*Over-the-air provisioning*) no modem para atualização automática da aplicação embarcada à distância.

### 1.3. Organização do documento

Este documento é composto por cinco capítulos, cada um apresentado da seguinte forma:

Capítulo 1 – Mostra uma introdução aos conceitos e tecnologias da rede GSM, além de um breve histórico do desenvolvimento da mesma. Este capítulo também demonstra a motivação do aluno para desenvolver tal projeto e os objetivos finais alcançados.

Capítulo 2 – Apresenta um estudo sobre o estado da arte do projeto, mostrando como as tecnologias utilizadas são empregadas hoje em dia e mostrando também sistemas similares ao desenvolvido.

Capítulo 3 – Nesta seção a arquitetura do *framework* será introduzida de maneira genérica. Aqui também serão discutidas detalhadamente as tecnologias utilizadas no projeto, como GPRS, SMS, OTAP, e o terminal TC65 da *Siemens*®, utilizado no projeto para comunicação direta com os *nobreaks*, mostrando seus dados técnicos e também suas limitações.

Capítulo 4 – Nesta seção é apresentada a aplicação embarcada que roda no terminal se comunicando com o *nobreak*. Também serão mostradas as dificuldades encontradas nesta etapa. Este capítulo também apresenta o *software* de controle desenvolvido para monitoramento e controle remoto do modem, também mostrando as dificuldades encontradas.

Capítulo 5 – No último capítulo, a monografia será concluída, falando sobre as dificuldades gerais encontradas no projeto, os resultados obtidos e os trabalhos futuros.

## 2. Trabalhos Relacionados

O avanço das redes GSM e da tecnologia GPRS permite que cada vez mais sistemas e aplicações sejam desenvolvidos, objetivando um maior controle sobre ambientes, equipamentos e veículos. Após uma pesquisa mais aprofundada, foram encontradas algumas soluções que utilizam estas tecnologias para telemetria.

Para realizar a comparação entre os sistemas encontrados e o projeto desenvolvido, foram utilizados critérios como: as funcionalidades de cada sistema, o custo de mercado, utilização de tecnologias, desempenho do sistema, etc.

A maioria dos sistemas e pesquisas encontrados trata do monitoramento e rastreamento de veículos ou ambientes. Entretanto, os sistemas [6], [7], [8] e [9] demonstraram certa semelhança com o desenvolvido neste trabalho, especialmente o [6], o [7] e o [8], que são sistemas de telemetria voltados ao monitoramento de *nobreaks*.

O foco do sistema apresentado em [6] é o monitoramento patrimonial, nesse caso o produto é conectado a uma central de alarmes. Entretanto o equipamento pode ser vendido com softwares específicos para outras áreas como, por exemplo, monitoramento de geradores, *nobreaks* e monitoramento de presença.

O programa da central [6] possui uma tecla Auto/Manual, utilizada para habilitar o envio de SMS pelo módulo para telefones previamente cadastrados. Os telefones programados são de técnicos preparados para assistência técnica dos equipamentos (geradores e *nobreaks*). Quando a chave está em Manual, as mensagens não são enviadas. Quando a chave está em Auto, as mensagens são enviadas.

A mensagem é formada por um identificador do cliente e um diagrama de estado do equipamento, com o qual o técnico tem condições de analisar o que ocorreu no equipamento. Um relatório é gerado a cada período de tempo previamente determinado pelo cliente, entre outras coisas esse relatório possui a somatória de horas trabalhadas de cada equipamento.

Neste sistema, o usuário pode se comunicar com o módulo que faz o monitoramento apenas através de mensagens SMS, que é apenas um formato de comunicação apresentados aqui. Realizando um comparativo, o projeto dessa monografia, além de possibilitar mais tipos de comunicação com o módulo, permite também um maior controle sobre o dispositivo eletrônico monitorado, já que esse sistema só possibilita a requisição de informações e a geração de informação por eventos do *nobreak*. Uma

vantagem desse sistema em relação ao apresentado é a possibilidade de desligar o envio de mensagens, já que nem sempre elas precisam ser enviadas.

O produto descrito em [7] também consiste num sistema de monitoramento e gerenciamento remoto para *nobreaks* a partir de uma central de serviços. O gerenciamento externo do *nobreak* não interfere nas operações da rede interna nem depende da mesma, oferecendo aos administradores controle de forma ininterrupta sobre seus equipamentos. O dispositivo tem como principais características: um servidor *web* integrado, interfaces RS232 e *Ethernet*, permitem acesso através da *web*, relógio interno auto-ajustável com servidores locais ou servidores de tempo localizados na internet, atualização de *firmware* através da internet e a disponibilidade em versão analógica (linha telefônica) ou versão GSM.

As vantagens que esse sistema apresenta são todas através da interface *Ethernet*, que permite a comunicação do módulo com a Internet. Por outro lado, isso aumenta o custo final da solução, tornando-a mais cara.

O sistema mostrado em [8], é muito semelhante com o definido em [7], também utiliza a internet e faz o monitoramento remoto de *nobreaks*, sendo que este permite a notificação de falhas via e-mail e através de um modem GSM. Uma desvantagem é que ele só permite o monitoramento dos dispositivos, não havendo o seu controle remoto.

Já o sistema descrito em [9], consiste em um equipamento que executa a função de transmissão e recepção de dados de telemetria, estando conectado à rede de telefonia celular digital (GSM/GPRS/SMS), permitindo também um acompanhamento on-line (com um uso de equipamentos extras). Este produto é mais voltado para monitoramento de pontos fixos (residências, indústria, comércio, etc.).

O produto [9] consiste em um módulo celular com “chip” GSM, acoplado a uma unidade inteligente de processamento. Assim as informações capturadas de sensores (de presença, movimento, portas, janelas, temperatura, impacto, etc.) são codificadas (criptografadas) e enviadas em tempo real para uma central. Com estes dados, os computadores através de software específico e mapeamento digitalizado, identificam o local e o evento.

As tecnologias utilizadas por [9] são as mesmas do trabalho proposto nesta monografia, mas não permite o controle, apenas o monitoramento, tendo, um custo um pouco maior do que [9], por causa do *software* de controle.

Outros sistemas como [10], [11] e [12] também utilizam a tecnologia GSM/GRPS, mas são mais voltadas para o rastreamento de veículos ou aplicações parecidas, o

projeto descrito em [12], por exemplo, pode ser usado na área de segurança da informação. Os sistemas que realizam o rastreamento de veículos geralmente utilizam também a tecnologia GPS<sup>7</sup>.

Um fato interessante é que nenhum dos sistemas apresentados faz o uso da tecnologia OTAP para atualização remota das aplicações. Não se encontram muitos trabalhos que utilizem essa tecnologia, especialmente na área de telemetria. Alguns trabalhos que a utilizam foram encontrados, como, por exemplo, [15] e [16], ambos os trabalhos utilizam a tecnologia OTAP para atualização das aplicações em seus sistemas.

Várias aplicações utilizam as tecnologias aqui descritas, mas para propósitos diferentes. GPRS, por exemplo, pode ser encontrada em trabalhos como [13] e [14]. Em [13], a tecnologia foi utilizada para monitoramento da poluição atmosférica, também utilizando alarmes para avisar quando o nível de poluição está alto, já em [14], é apresentado uma rede de sensores para monitoramento de containers de navios, que se comunicam com um controlador informando sobre qualquer problema.

Como exemplo de uso do módulo TC65, tem-se [17], que apresenta uma plataforma de rede de sensores sem-fio com um alcance muito maior, consumindo a mesma quantidade de energia de redes comuns. Ela faz uso de redes *ethernet*, WLAN e GPRS (usando o TC65) e usa energia solar para diminuir o seu consumo.

É possível notar que as tecnologias utilizadas neste trabalho já estão bastante difundidas em várias aplicações existentes, sendo que este número tende a aumentar à medida que as tecnologias crescem. Deste estudo sobre o estado da arte, é possível mostrar que o trabalho apresentado engloba várias funcionalidades de diferentes produtos, se mostrando ser um trabalho mais abrangente e geral, fornecendo funcionalidades ao usuário que só seriam atingidas se o mesmo utilizasse dois ou mais dos produtos mostrados.

Nos capítulos que seguem, serão apresentadas as funcionalidades do *framework* desenvolvido, assim como algumas explicações necessárias.

---

<sup>7</sup> GPS (*Global Positioning System*) – sistema de posicionamento por satélite, determina a posição de um receptor na Terra.

### 3. Arquitetura e Tecnologias para Monitoramento Remoto

Nesta seção será apresentada a arquitetura do *framework* desenvolvido neste trabalho, apresentando seus módulos suas comunicações, assim como as tecnologias envolvidas no projeto, dando uma breve explicação sobre como e onde as mesmas foram utilizadas. Neste capítulo também será apresentado o módulo TC65 utilizado.

#### 3.1. Arquitetura do *Framework*

O *framework* proposto tem como objetivo principal realizar o monitoramento e o controle de certo dispositivo. Para tal, foram desenvolvidos três módulos, o módulo controlador, o módulo de comunicação e o módulo *web*. A Figura 1 apresenta um diagrama da arquitetura do sistema proposto.

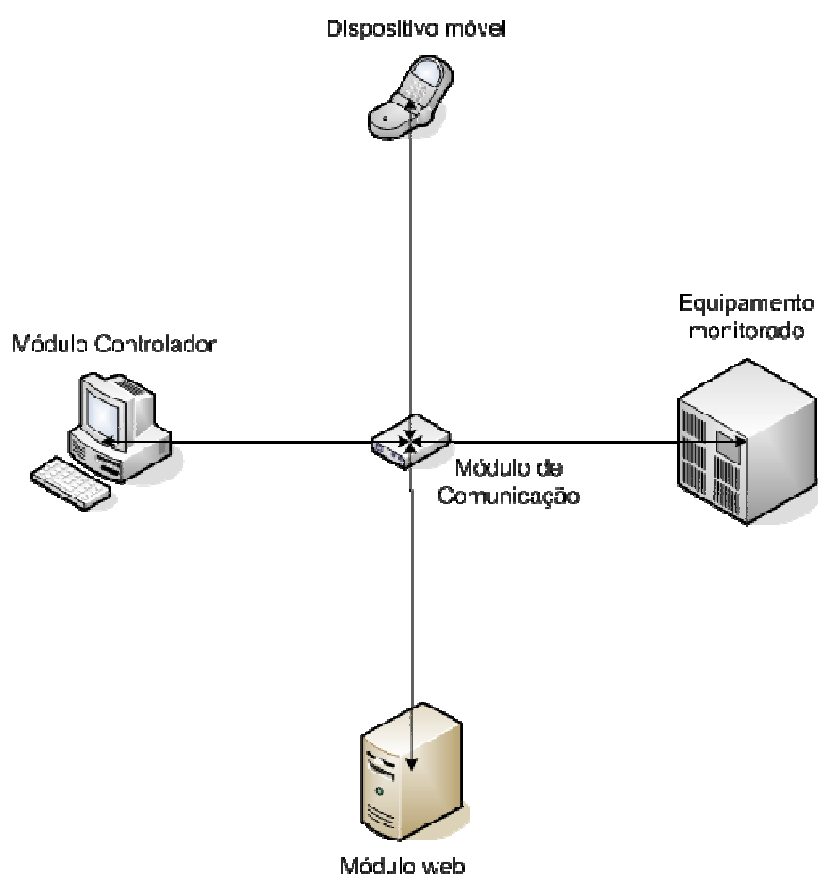


Figura 1. Arquitetura do *Framework*

Nesta figura é possível notar que o módulo de comunicação tem ligações com todas as outras entidades, sendo o principal módulo do *framework*. Ele é responsável por se comunicar com o equipamento monitorado obtendo dados que tornam possível a verificação do seu funcionamento.

O módulo de controle tem a função exclusiva de coordenar o módulo de comunicação. Eles se comunicam para troca de informações onde o módulo de controle envia informações ao módulo de comunicação de como ele deve funcionar. Em outras palavras, o módulo de controle define alguns atributos do módulo de comunicação para que este realize sua função. Ambos os módulos utilizam os mesmos atributos, só que para fins diferentes. Enquanto o módulo de comunicação utiliza os atributos como um guia para realizar suas operações, o módulo de controle guarda estes atributos apenas para apresentar ao usuário.

Outras formas de comunicação ainda são possíveis entre estes dois módulos principais. O módulo de controle é capaz de se comunicar com o equipamento monitorado através do módulo de comunicação. Para isto, o módulo de comunicação recebe uma requisição de controle e se prepara para intermediar a comunicação entre controle e equipamento.

O módulo de controle ainda é capaz de requisitar ao módulo de comunicação os seus atributos e também os dados que este recebe do equipamento monitorado. Caso os atributos do módulo de comunicação tenham sido obtidos, o módulo de controle é capaz de carregá-los em seus próprios atributos.

Já o módulo *web* tem uma função simples no *framework*. Ele é responsável apenas por armazenar aplicações que são executadas pelo módulo de comunicação. Este, por sua vez, inicia uma comunicação com o módulo *web* quando requisitado, através de um dispositivo móvel, obtendo a aplicação nele armazenada e atualizando seu sistema. Como este módulo é utilizado apenas para configurar a aplicação a ser executada no módulo de comunicações, ele é acessível apenas à programadores do módulo. Usuários comuns do *framework* não têm acesso a tal serviço. Usuários comuns se referem aos usuários que serão responsáveis por alarmes, que irão interagir com o módulo de controle. A única interação entre os usuários comuns e o sistema é dada através do módulo de controle (tratando-se de um software com interface gráfica) e através de envios de mensagens pelos dispositivos móveis.

Através de um dispositivo móvel (um celular, um PDA, um *Tablet*, etc.), também é possível realizar outras operações. Com ele é possível requisitar ao módulo de

comunicação os dados obtidos do equipamento monitorado. O módulo, então, enviará os últimos dados que foram obtidos ao dispositivo. Um dispositivo móvel também é necessário para iniciar algumas comunicações entre módulos.

As próximas subseções deste capítulo apresentam os equipamentos e as tecnologias utilizadas no desenvolvimento de um sistema baseado na arquitetura do *framework* proposta.

### 3.2. Módulo TC65

O modem (ou módulo) utilizado no trabalho foi o módulo TC65 da *Siemens*®. Este foi escolhido por se tratar de um modem apresentado em várias aplicações e por permitir a sua programação através da linguagem Java® (JME®).

O TC65 possui duas portas para comunicação serial, uma porta para comunicação USB<sup>8</sup>, uma entrada para a alimentação, uma entrada para antena de sinal GSM e uma entrada para SIMCARD (*chip* de celular), além de um botão de *reset* e alguns *leds* de diagnóstico.

Suas duas entradas seriais utilizam o padrão RS-232 com um detalhe importante, geralmente neste padrão os pinos ‘2’ e ‘3’ da porta se comunicam, respectivamente, com os pinos ‘3’ e ‘2’ da outra extremidade. Já, no modem em questão, os pinos ‘2’ e ‘3’ são invertidos, permitindo aos usuários o uso de simples cabos seriais paralelos (pinos ligados paralelamente, ‘1’ no ‘1’, ‘2’ no ‘2’, etc.). Um cabo ideal é apresentado na Figura 2:

---

<sup>8</sup> USB (*Universal Serial Bus*) – Tipo de conexão *plug-and-play*.

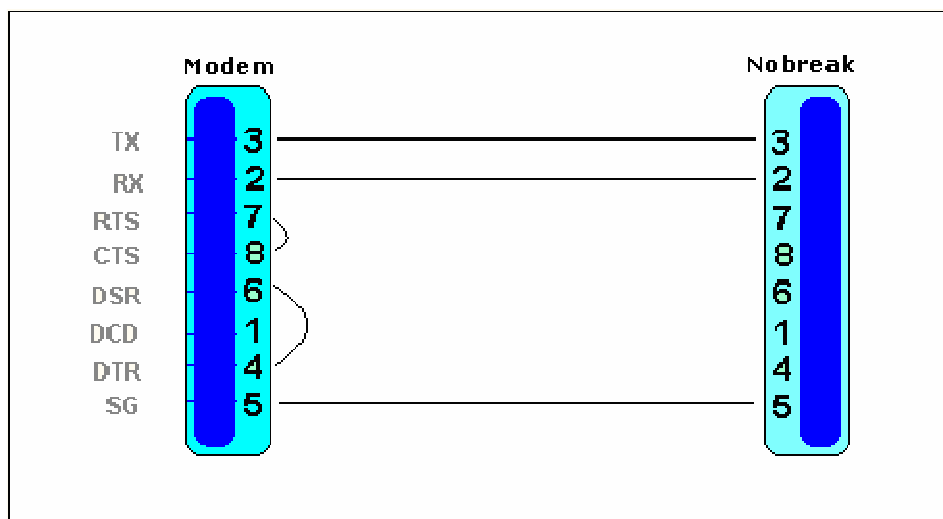


Figura 2. Configuração de um cabo ideal para o sistema

A figura mais a esquerda mostra a ponta do cabo que deve ser conectada ao módulo e a da esquerda a ponta que é ligada ao *nobreak*.

Outros cabos seriais normais podem ser utilizados, mas geralmente será necessária a troca dos pinos '2' e '3'. Para um melhor entendimento sobre os cabos utilizados, ver o Apêndice C - Interfaces.

As duas portas seriais são nomeadas de ASC0 e ASC1, a primeira é utilizada basicamente para comunicação com o PC e a segunda para qualquer tipo de comunicação necessária. É possível alterar essa ordem enviando certos comandos ao módulo.

O módulo aceita *chips* de qualquer operadora, o usuário deve se preocupar apenas em definir os atributos de comunicação GPRS de acordo com a operadora utilizada, isso será explicado na seção 3.4.

A tensão de alimentação para o TC65 varia de 8 a 30 volts, qualquer tensão DC<sup>9</sup> nessa faixa aplicada ao próprio, fará com que ele funcione perfeitamente. Tensões mais baixas aplicadas ao modem também farão com que o mesmo funcione, mas não é garantido que funcione perfeitamente. Ao serem aplicadas tensões inferiores às especificadas, pôde-se notar uma falha no funcionamento do modem, o que, em alguns casos, faz com que o mesmo se desligue.

Quanto às temperaturas de operação do modem, na sua especificação encontrasse que os limites para operação dele são de -30°C a até 65 °C. Não foram realizados testes com diferentes temperaturas, apenas seu funcionamento a temperatura ambiente. O módulo tem as dimensões 130 mm x 90 mm x 38 mm e pesa cerca de 90 gramas.

<sup>9</sup> Tensão DC – Tensão Contínua.

As figuras abaixo mostram imagens do TC65, utilizado neste trabalho:



**Figura 3. Módulo TC65 (vista superior traseira).**



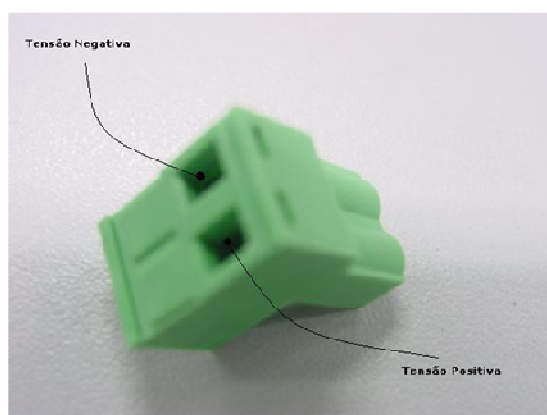
**Figura 4. Módulo TC65 (vista superior frontal).**



**Figura 5. Modem com entrada de alimentação e antena GPRS.**

A peça marcada em “1” é responsável pela alimentação do modem. É importante saber a ordem da alimentação, pois no lado esquerdo entra a tensão negativa e no direito

a tensão positiva. A troca dessa ordem provavelmente irá danificar o equipamento. A Figura 6 mostra como deve ser ligada à alimentação ao modem:



**Figura 6. Tensão negativa à esquerda, tensão positiva a direita.**

O TC65 possui um processador ARM7®<sup>10</sup> com uma memória. A memória do TC65 se divide em memória Flash<sup>11</sup> e memória RAM<sup>12</sup>. A memória Flash para aplicações Java é de 1,73MB e a RAM é de apenas 400KB. Isso mostra que a capacidade do módulo é bem inferior à dos PCs<sup>13</sup>, o que é de se esperar por se tratar de um dispositivo embarcado.

O modem pode ser programado pela porta serial, geralmente a *asc1*, do modem, que é a porta serial que fica do lado da entrada USB. Para acessar os dados do modem, foi utilizado o *Mobile Exchange Suíte* da Siemens®, um programa que permite acesso direto, como se o modem fosse uma unidade de disco do PC. Para programá-lo, a IDE<sup>14</sup> Eclipse®, com o *plugin*<sup>15</sup> EclipseME, que faz com que o Eclipse® trabalhe com a linguagem J2ME®, foi utilizada. Este *plugin* trabalha em conjunto com outro programa da Siemens®, o *Siemens Mobile Toolkit*®, que coloca a aplicação desenvolvida para ser executada pelo modem.

Algumas aplicações podem usar o TC65 sem precisar programá-lo, apenas enviando comandos pela porta serial. Nem todos os comandos são interpretados pelo modem. Os comandos que são interpretados são os chamados comandos AT. Estes comandos

---

<sup>10</sup> ARM7 – Família de microprocessadores de 32 bits.

<sup>11</sup> Memória Flash – Memória de computador do tipo EEPROM (*Electrically-Erasable Programmable Read-Only Memory*), comumente utilizada em cartões de memória, câmeras digitais, celulares, etc.

<sup>12</sup> Memória RAM (*Random Access memory*) – Memória primária em sistemas eletrônicos digitais.

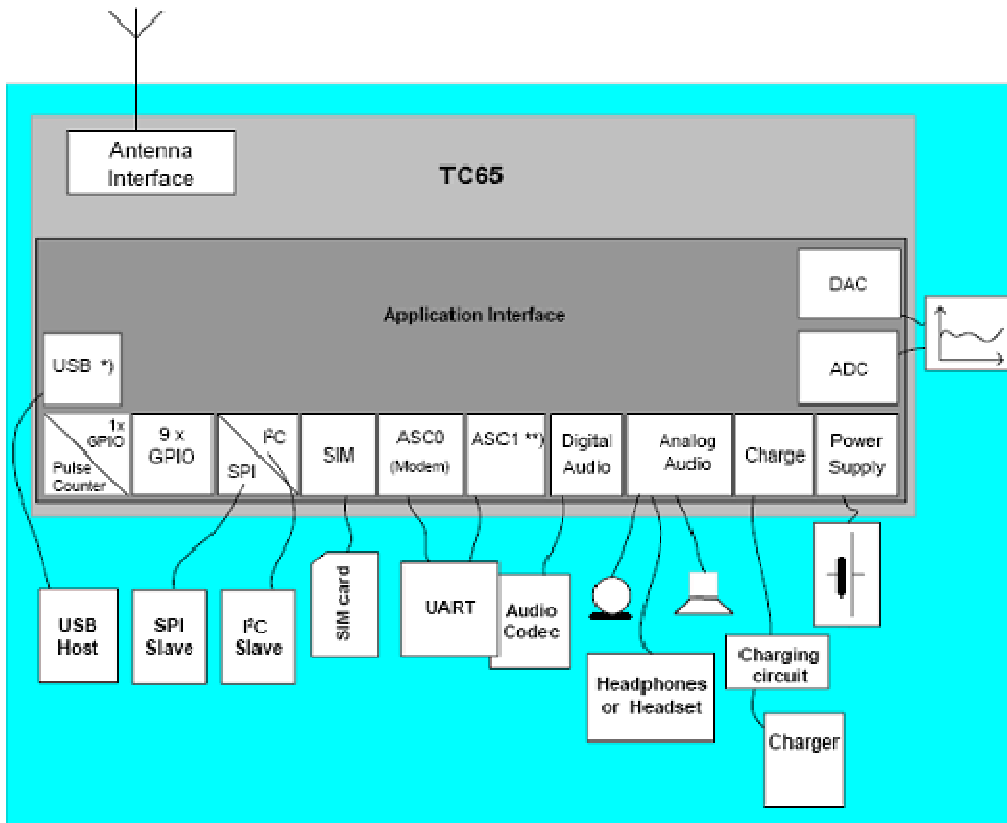
<sup>13</sup> PC (*Personal Computer*) – Computadores pessoais, computadores que as pessoas tem em suas casa, escritórios, etc.

<sup>14</sup> IDE (*Integrated Development Environmet*) – Um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software com o objetivo de acelerar este processo.

<sup>15</sup> Plugin – Programa de computador que adiciona funcionalidade a outro programa.

podem ser encontrados em [18]. Cada comando faz com que o modem execute alguma função, ou define algum atributo do modem. Todos os comandos utilizados no projeto estão explicados no apêndice A.

O módulo também tem algumas características especiais como um relógio de tempo real, cujas funções de Timer podem ser acessadas via comandos AT, e possui também um conversor analógico digital com duas entradas. A Figura 7 representa de forma resumida o modelo de interfaces e componentes do módulo TC65:



**Figura 7. Resumo esquemático do módulo TC65**

Maiores informações sobre o módulo TC65 podem ser encontradas em [19], [20] e [21].

### 3.3. SMS

A tecnologia de envio de mensagens SMS (*Short Message Service*) já é bastante conhecida em todo o mundo. Todos os celulares permitem trocas de mensagens neste formato. SMS é, na verdade, um protocolo de comunicação para troca de mensagens curtas. Esse protocolo foi integrado aos padrões da rede GSM em 1985, sendo também utilizado em redes que antecederam a GSM. A mensagem é chamada de “curta”, pois é

limitada a 160 caracteres (para envio de mensagens maiores, envia-se a mensagem por partes).

Devido às suas características, o serviço SMS é muito usado nos sistemas de telemetria existentes, aliando uma funcionalidade simples ao sistema. Geralmente as mensagens são utilizadas para requisitar informações do sistema, mas, em alguns casos, são também utilizadas para configurar o sistema.

A tecnologia SMS facilitou o desenvolvimento e o crescimento de mensagens de texto, mas vem sendo lentamente substituída pela tecnologia GPRS, devido aos seus custos mais baratos.

Mas por que usar SMS se a tecnologia GPRS é mais barata? A tecnologia GPRS, apesar de seu menor custo, ainda não está tão difundida nos celulares existentes, sendo mais fácil para os usuários enviar mensagens SMS.

No trabalho em questão, a tecnologia SMS foi utilizada da mesma forma que nos sistemas de telemetria, para obter informações de status do equipamento ou ambiente monitorado (no nosso caso para obter os atributos do *nobreak*). Mas, além disso, ela também é utilizada para requisitar ao modem que se comunique de alguma forma com o servidor. O servidor neste caso pode significar duas coisas, um servidor GPRS, que se trata do próprio programa de controle, ou um servidor *web*. Quando o tipo de requisição enviada ao modem é para comunicação GPRS, o modem se conecta ao servidor GPRS (ao programa) e quando a requisição é para atualização da aplicação via OTAP, o modem se conecta ao servidor *web*. Todos estes processos serão melhores descritos no capítulo 4, onde serão apresentadas as funcionalidades do sistema, explicando onde e como as mensagens SMS são utilizadas.

O módulo TC65 usa a tecnologia SMS-CB<sup>16</sup>, que surgiu em 1997 numa demonstração em Paris. Antes, a tecnologia SMS utilizada era a SMS-PP<sup>17</sup> a qual permitia que as mensagens fossem enviadas para um ou poucos usuários. Com o surgimento da tecnologia SMS-CB, as mensagens SMS podem ser enviadas simultaneamente a vários usuários, sendo bastante utilizadas em casos de emergências. O módulo TC65 trabalha com os dois modos SMS, PDU e Texto. O modo texto é o modo mais comum, onde todos conseguem ler a informação transmitida na mensagem. Já no modo PDU, os usuários normais não conseguem ler a mensagem, pois ela vem

---

<sup>16</sup> SMS-CB (*Short Message Service – Cell Broadcast*) – Tecnologia desenvolvida para entrega de mensagens a múltiplos usuários.

<sup>17</sup> SMS-PP (*Short Message Service – Point to Point*) – Tecnologia SMS para poucos usuários.

codificada. Na realidade, quase todas as mensagens SMS enviadas, utilizam o modo PDU, mas a leitura dos celulares, por exemplo, usa o modo Texto, ou seja, eles traduzem a mensagem para que o usuário consiga lê-la. Com o módulo TC65 isso deve ser feito manualmente através de comandos AT (ver Apêndice A), um simples comando define se a leitura será no modo Texto ou PDU.

### 3.4. GPRS

A tecnologia com a maior relevância neste projeto foi, sem dúvida, a tecnologia GPRS. Ela torna possível o controle remoto dos *nobreaks*, permitindo aos usuários a utilização do Software de Controle mesmo sem haver o contato com o módulo TC65. Com já mencionado, GPRS permite a troca de dados a um custo mais barato, o que permite uma maior difusão desta tecnologia em aplicações e aparelhos.

Esta tecnologia utiliza o protocolo WAP<sup>18</sup> para se comunicar com outros dispositivos ou com a Internet. Na teoria, GPRS suporta conexões do tipo IP<sup>19</sup>, PPP<sup>20</sup> e X.25, sendo que esta última removida do padrão, pois ela pode ser suportada através do PPP ou até mesmo pelo IP.

O módulo TC65 trabalha com a tecnologia GPRS. Para isto, tudo que ele precisa é de uma antena conectada e de um SIMCARD válido para conexão. A antena GPRS pode ser vista na Figura 8:



**Figura 8. Antena GPRS**

---

<sup>18</sup> WAP (*Wireless Application Protocol*) – Protocolo de redes sem-fio.

<sup>19</sup> IP (*Internet Protocol*) – Protocolo de comunicação da Internet.

<sup>20</sup> PPP (*Point-to-Point Protocol*) – Protocolo de comunicação ponto-a-ponto.

A tecnologia GPRS é dividida em três classes, A, B e C. Estas classes classificam os aparelhos que utilizam a tecnologia. A Tabela 1. Classes GPRS relaciona as classes e suas possibilidades:

**Tabela 1. Classes GPRS**

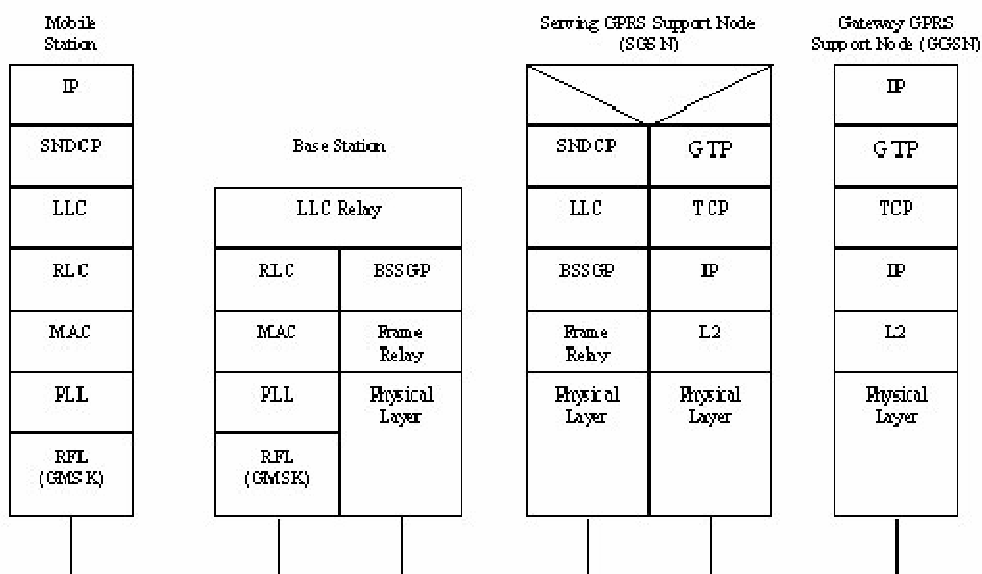
Classe	Possibilidades
Classe A	Podem utilizar os serviços GPRS e os serviços GSM (Voz, SMS) simultaneamente.
Classe B	Podem utilizar ambos os serviços, mas não ao mesmo tempo, quando um dos serviços está em uso o outro é suspenso, havendo a troca automática quando o serviço que suspendeu é finalizado.
Classe C	Parecido com a Classe B, só que a troca dos serviços deve ser feita manualmente e não há a suspensão automática.

O módulo TC65 trabalha com a classe B, suspendendo internamente seus serviços, sem necessidade da intervenção do usuário.

No trabalho apresentado, a comunicação GPRS é feita pela aplicação do modem, através da biblioteca de *sockets* de Java. O modem se comunica com o servidor GPRS, representado pelo Software de Controle. A comunicação GPRS no sistema será apresentada mais a frente, no capítulo 4.

Tanto a tecnologia SMS como a GPRS utilizam os protocolos X.25 e IP para transferência de dados. A Figura 9 mostra a pilha de protocolos utilizadas nas redes GPRS e suas respectivas interfaces de comunicação.

**GPRS Protocol Stack**



**Figura 9. Pilha de protocolos GPRS**

### 3.5. OTAP

OTAP é uma tecnologia que permite a instalação, atualização e remoção de aplicações através do ar, ou seja, permite “configurar” certo dispositivo à distância. Esta tecnologia é utilizada para que o usuário não precise de contato físico com o dispositivo para que o mesmo tenha sua aplicação instalada, atualizada ou removida.

A idéia é enviar uma mensagem SMS para o dispositivo, que irá identificar que esta mensagem é uma requisição para OTAP e realizará o que fora pedido, seja instalação, atualização ou remoção. No caso de instalação ou atualização, o dispositivo irá se comunicar, via GPRS, com um servidor http onde os arquivos necessários foram armazenados. A Figura 10 mostra um esquemático do uso da tecnologia com o módulo TC65 (figura retirada de [19]):

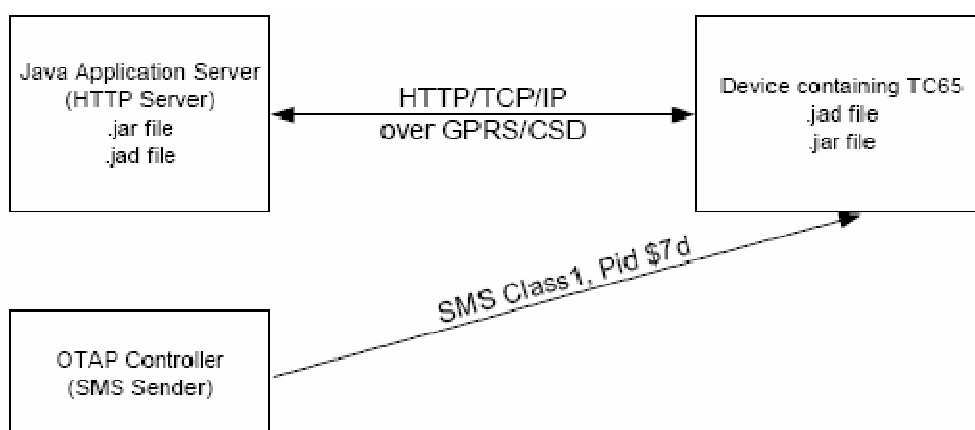


Figura 10. OTAP no TC65

Como a figura mostra, um SMS é enviado ao TC65 que se comunica com um servidor http onde os arquivos *jar* e *jad*, arquivos da aplicação, se encontram.

As mensagens para instalar ou atualizar uma aplicação são idênticas, sendo necessário o envio de duas mensagens, já a mensagem para remover é diferente, sendo menor, apenas um SMS.

Os formatos das mensagens SMS a serem enviadas ao TC65 para que o mesmo inicie OTAP podem ser vistas em [19], só que, neste trabalho o uso desta tecnologia é feita de maneira diferente. As mensagens utilizadas são muito extensas e complicadas, para facilitar a vida do usuário, este trabalho requer que o mesmo configure apenas o endereço http, o nome do arquivo *jad* e que um SMS simples seja enviado ao modem.

Este processo será apresentado na seção 4.1.4. Ambas as formas podem ser utilizadas para se iniciar o OTAP neste sistema.

## 4. Desenvolvimento do *Framework* para controle remoto

O sistema que foi desenvolvido neste trabalho é representado pela Figura 11.

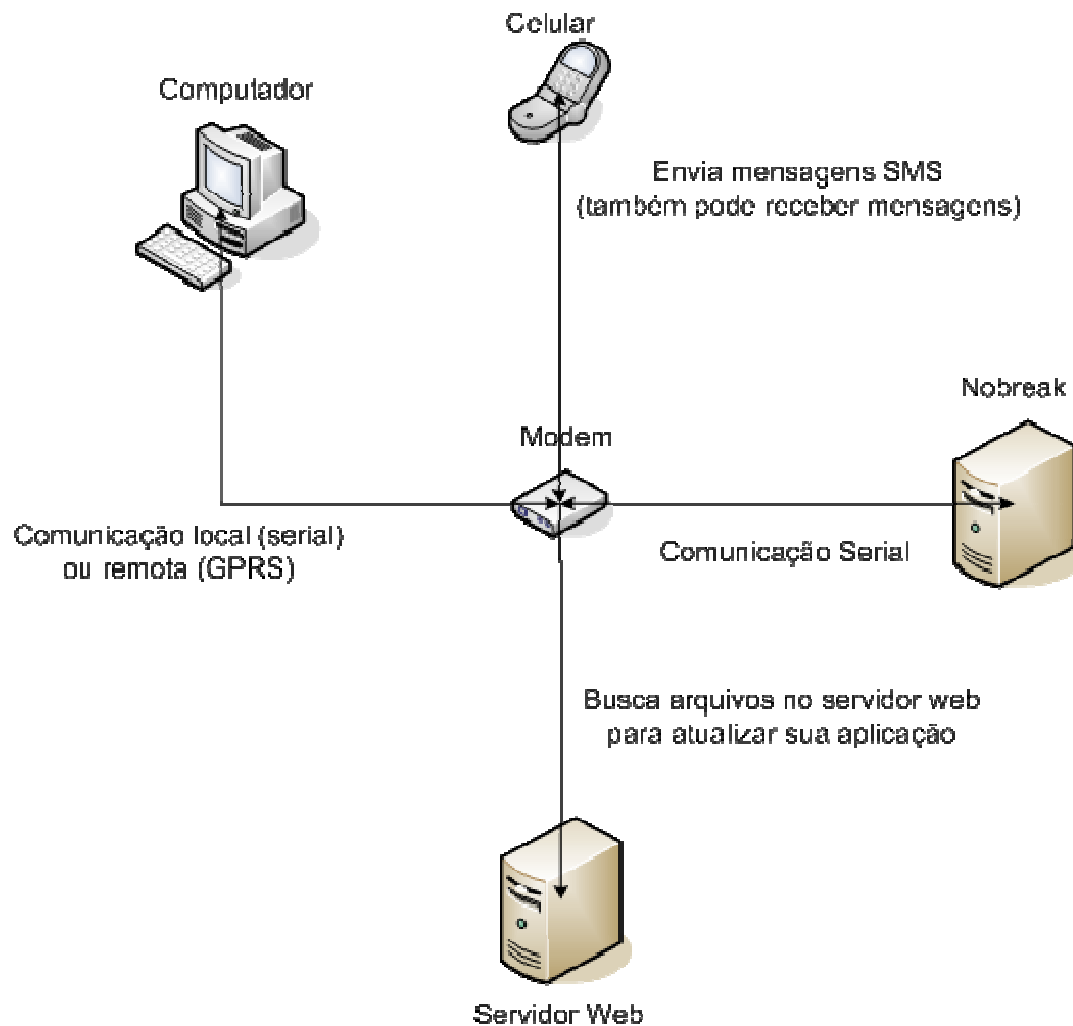


Figura 11. Arquitetura do sistema

Nesta arquitetura são apresentadas as cinco entidades que fazem parte do sistema.

Basicamente o sistema funciona da seguinte forma:

- O módulo de comunicação (representado pelo modem) começa a enviar comandos ao *nobreak*, requisitando seu status atual;
- O *nobreak* envia o status ao modem, que checa se algum dos alarmes do *nobreak* fora disparado;

- Caso haja nenhum alarme, o modem verifica as mensagens recebidas. Caso haja, ele envia um SMS aos usuários responsáveis por tal alarme e só depois faz a verificação das mensagens recebidas.

Neste capítulo o sistema, ou o *framework*, desenvolvido será explicado mais abrangentemente.

O *framework* desenvolvido neste trabalho é dividido em dois projetos diferentes, que se comunicam entre si a fim de realizar um monitoramento e controle, neste caso, de *nobreaks*. O primeiro é o Projeto Modem, uma plataforma embarcada que faz o uso do módulo de comunicação que, como já fora dito (seção 3.1), executa aplicações escritas em JME®. O módulo é utilizado para se comunicar diretamente (via porta serial) com o *nobreak*, obtendo seus dados e enviando comandos. O projeto tem um “ciclo de vida” que se inicia desde o momento em que o modem é alimentado e só para se o mesmo for desligado. O projeto também permite comunicações GPRS e SMS com o módulo, basta o usuário enviar simples mensagens ao modem para iniciar as comunicações (o ciclo de vida e os modos de comunicação com o modem serão apresentados mais abaixo, na subseção 4.1 deste capítulo).

O segundo projeto é o Projeto Interface do Modem, que se trata de um programa de interface com o usuário para controle e monitoramento (também chamado neste documento de *Software de Controle*). Através dele, o usuário pode definir as configurações do modem e se comunicar com ele. O sistema é regido por um arquivo de configurações (as propriedades do sistema), o modem também faz uso deste arquivo, a partir dele que algumas funcionalidades são iniciadas. O arquivo é configurado através do programa, que também o envia ao modem. Este projeto também é responsável pela comunicação GPRS com o modem, um servidor GPRS é iniciado juntamente com o programa. O modem, então, se conecta a este servidor para enviar e receber dados.

Praticamente todo o contato do usuário com o projeto será feito através do *software* de controle. A única necessidade de interação com o modem será para ligá-lo ou reiniciá-lo em alguns casos.

O arquivo de configuração usado pelo projeto é mais bem explicado no Apêndice B, ao final deste documento, e também todas as funcionalidades aqui citadas serão mais bem explicadas na subseção 4.2 deste documento.

## 4.1 Plataforma Embarcada

A aplicação embarcada foi desenvolvida para que o usuário não precise de qualquer contato com o modem, a única tarefa com relação ao modem será a de alimentá-lo. O modem utilizará sua função *autostart* para iniciar automaticamente a aplicação assim que for ligado. Esta função pode ser desabilitada como pode ser visto em 1.1.1.4.1.4.2.7.

O modem é quem faz toda a comunicação com o *nobreak*. A comunicação com o modem pode ser feita através de comandos AT, como já dito, que são enviados ao modem pela interface serial.

A plataforma também utiliza o arquivo de configuração para carregar as propriedades e atributos que serão utilizados no controle e monitoramento dos equipamentos, este arquivo deve estar atualizado para que o modem funcione de acordo com o que foi configurado (ver 1.1.1.4.1.4.2.10).

As principais funcionalidades da aplicação embarcada são apresentadas nas subseções a seguir.

### 4.1.1. Timer

O modem faz uso de um timer para comunicação com o *nobreak*. A cada intervalo de tempo ('x' segundos definidos pelo usuário através do programa de controle, ver 1.1.1.4.1.4.2.1) o modem requisita o status do *nobreak* e checa se não há algum alarme ativo. Em caso afirmativo, significa que o *nobreak* está com algum mau funcionamento, um ou mais dos seus alarmes fora disparado, sendo enviada uma mensagem SMS é enviada aos usuários responsáveis por cada alarme (os responsáveis pelos alarmes também são escolhidos através do programa de controle, ver 1.1.1.4.1.4.2.5).

Além da obtenção do status, o modem também verificará as mensagens recebidas pelo chip. Algumas mensagens podem ser enviadas ao modem requisitando que o mesmo realize alguma operação. Estas mensagens são apresentadas em 4.1.6 e é através delas que algumas funcionalidades disponibilizadas pelo modem são acessadas.

#### 4.1.2. Timer do log

Um outro timer que pode ser utilizado pelo modem é o timer para salvamento de *log*. Este timer só será iniciado caso o usuário tenha marcado a opção para habilitá-lo no programa (ver 1.1.1.4.1.4.2.3).

A função deste timer é (a cada 'x' minutos, definidos pelo usuário) iniciar uma conexão GPRS com o Software de Controle (que funciona como um servidor GPRS), enviando o último status do modem que fora obtido. O programa, então, salva esse status em um arquivo de *log*. Este *log* pode ser exibido na tela do próprio software ou através de um leitor de texto qualquer.

Para utilizar esta funcionalidade, o programa deve estar rodando, pois ele será o servidor GPRS ao qual o modem se conectará.

#### 4.1.3. GPRS

O modem pode se comunicar com o programa de controle através da tecnologia GPRS, enviando e recebendo informações. Existem duas opções de comunicação GPRS. Na primeira, o usuário poderá requisitar o status atual do *nobreak*, além de fazer *download* ou *upload* do arquivo de configuração. Na segunda, o modem funcionará como um terminal de comunicação entre o PC e o *nobreak*, do mesmo jeito que ele faz via serial. Essa opção é mais usada quando se quer um controle remoto do *nobreak*, quando não se está conectado via serial com ele.

O endereço do servidor GPRS e a porta de conexão também são configurados pelo programa de controle. Para iniciar as conexões GPRS, mensagens devem ser enviadas ao modem. Estas mensagens serão mostradas adiante.

Uma melhor explicação sobre as conexões GPRS entre o modem e o software pode ser visto em 1.1.1.4.1.4.2.9.

#### 4.1.4. OTAP

OTAP é uma tecnologia através da qual é possível atualizar a aplicação existente no modem à distância, sem necessitar ao menos parar a aplicação que o módulo de comunicação está executando. Para isso, é necessário um servidor http onde são armazenados os arquivos *jad* e *jar* (gerados pelo JME®) que representam a aplicação. O modem se conectará nesse endereço buscando os arquivos. O endereço será configurado pelo programa, mas também poderá ser passado na mensagem (ver 4.1.6).

O arquivo *jad* deve ser editado, ele possui um campo chamado “MIDlet-Jar-URL”. Além dos arquivos estarem no servidor, neste campo deve ser passado o endereço do arquivo *jar*, por exemplo:

“MIDlet-Jar-URL: http://host:porta/update/Modem.jar”

Maiores informações sobre a tecnologia OTAP são encontradas em 3.3.5.

#### 4.1.5. Serial

A conexão serial pode ser usada quando o modem está diretamente ligado ao PC, barateando os custos, já que nada será gasto com transferência de dados pelo modem. Este tipo de comunicação serve como uma alternativa ao GPRS, permitindo um controle local.

Com esta comunicação o usuário pode enviar as alterações ao modem, usar o modem como um terminal de comunicação entre o PC e o *nobreak*, e habilitar ou desabilitar a função de “*autostart*” do modem.

As comunicações utilizando a interface serial são apresentadas no próximo capítulo (1.1.1.4.1.4.2.9).

#### 4.1.6. Mensagens SMS

O usuário também pode se comunicar com o modem através de mensagens SMS. São quatro tipos de mensagens válidas que o usuário pode enviar ao modem, três delas são mensagens de comando, a última é uma mensagem de requisição. Abaixo estão listadas as mensagens aceitas pelo sistema e os seus formatos. As palavras em itálicos significam que devem ser substituídas de acordo com os atributos do sistema:

- Requisição de status: este tipo de mensagem requisita ao modem o último status obtido do *nobreak*. O formato da mensagem passado é “*senhaDoModem login status*”, caso a senha esteja correta, o modem enviará o status ao número do usuário com o login passado.
- Comunicação GPRS: mensagem no formato “*senhaDoModem gprs [endereçoDeConexão]*”. O modem irá tentar se conectar via GPRS com o endereço passado caso a senha esteja correta. O endereço está entre colchetes

pois ele é opcional, se este campo não for passado na mensagem o modem tentará se conectar ao endereço contido no arquivo de configuração.

- Comunicação GPRS-Serial: mensagem no formato “*senhaDoModem serial [endereçoDeConexão]*”. O endereço entre colchetes representa o mesmo do tipo de mensagem anterior. Ao receber esta mensagem, o modem parará os timers e funcionará como um terminal de comunicação.
- Iniciar OTAP: mensagem no formato “*senhaDoModem otap [endereçoDeConexão]*”. O modem irá se conectar ao endereço passado, baixará os arquivos ‘*jad*’ e ‘*jar*’ e irá atualizar a sua aplicação com eles. Caso o *autostart* esteja habilitado, a nova aplicação será automaticamente iniciada. O endereço entre colchetes funciona da mesma maneira das outras mensagens, sendo opcional e em sua ausência o modem irá consultar o arquivo de configuração.

A Tabela 2. Mensagens SMS aceitas pelo modem. resume as mensagens aceitas e os seus formatos:

**Tabela 2. Mensagens SMS aceitas pelo modem.**

<b>Mensagem</b>	<b>Formato</b>	<b>Funcionalidade</b>
Requisição de status	<i>senha login status</i>	Obter status do <i>nobreak</i> .
Comunicação GPRS	<i>senha gprs [endereço]</i>	Comunicação GPRS com o software.
GPRS-Serial	<i>senha serial [endereço]</i>	Comunicação GPRS em ponte.
OTAP	<i>senha otap [endereço]</i>	Atualização da aplicação via OTAP.

## **4.2 Software de Controle**

O Software de Controle permite ao usuário configurar o módulo de comunicação, da maneira que melhor lhe convier, para que este realize o monitoramento de *nobreaks* (lembrando que o *software* foi desenvolvido para o estudo de caso do projeto, algumas alterações precisam ser feitas caso o equipamento a ser monitorado seja alterado). O

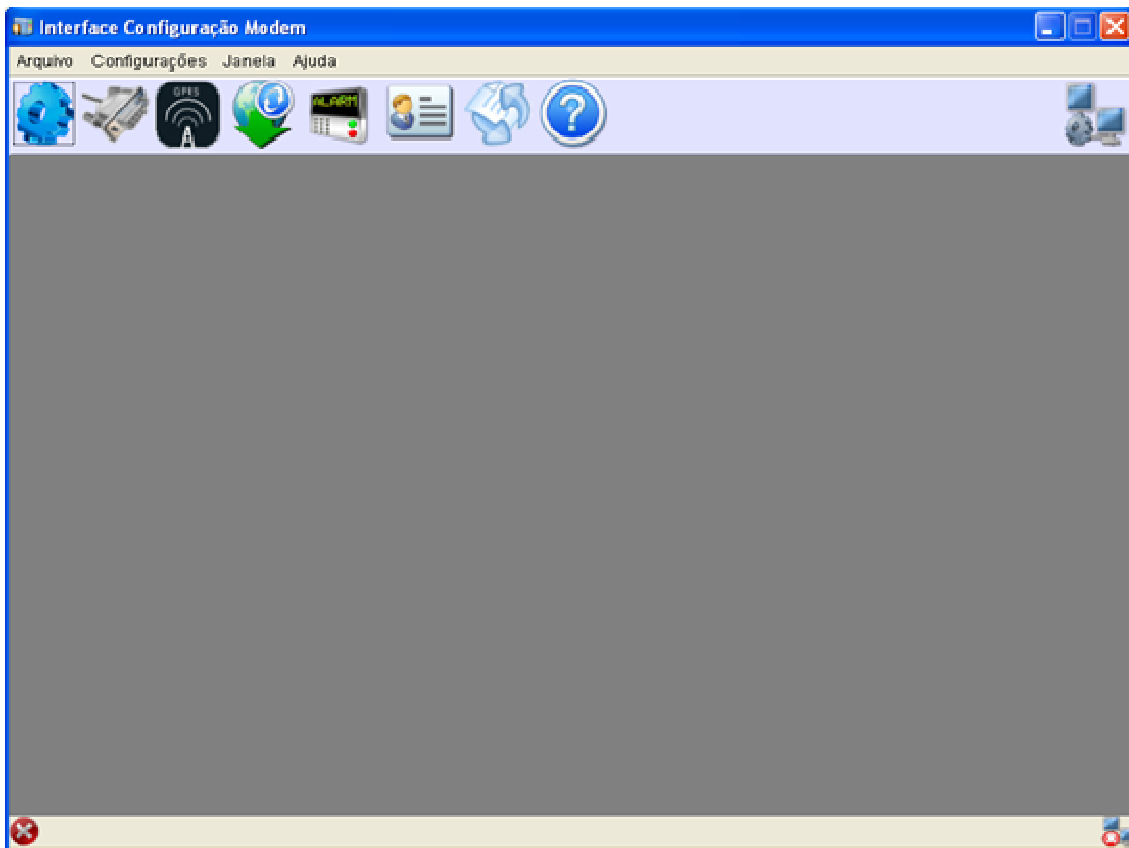
programa permite um monitoramento e controle remoto ou local, se comunicando com o modem via serial (local) ou GPRS (remoto).

O seu principal objetivo é definir propriedades que serão utilizadas pelo modem e permitir aos usuários se comunicarem diretamente com o *nobreak* através do modem, que, neste caso, atua como uma ponte de comunicação entre o PC e o *nobreak*. Isto pode ser feito através da própria porta serial ou através de uma conexão GPRS.

O funcionamento do software é baseado em um arquivo de configuração, como já mencionado (seção 4), onde tudo que foi definido é armazenado. O programa também utiliza este arquivo para carregar as configurações em suas telas. Sempre que alterações são realizadas através do programa, o arquivo é atualizado.

O modem faz uso do mesmo arquivo, através do qual ele carrega os atributos que serão utilizados no monitoramento do *nobreak* e para comunicação externa (GPRS, SMS e OTAP). Quando alterações são realizadas no arquivo utilizado pelo programa, há a necessidade de enviar este novo arquivo ao modem. Até isto acontecer, o modem estará trabalhando com um arquivo antigo (o último que foi recebido por ele). Este envio é realizado através do próprio programa, seja por uma conexão local ou remota.

As funcionalidades aqui introduzidas e mais algumas outras ainda não mencionadas serão melhor explicadas ao longo deste capítulo. A Figura 12 mostra a tela inicial do programa:

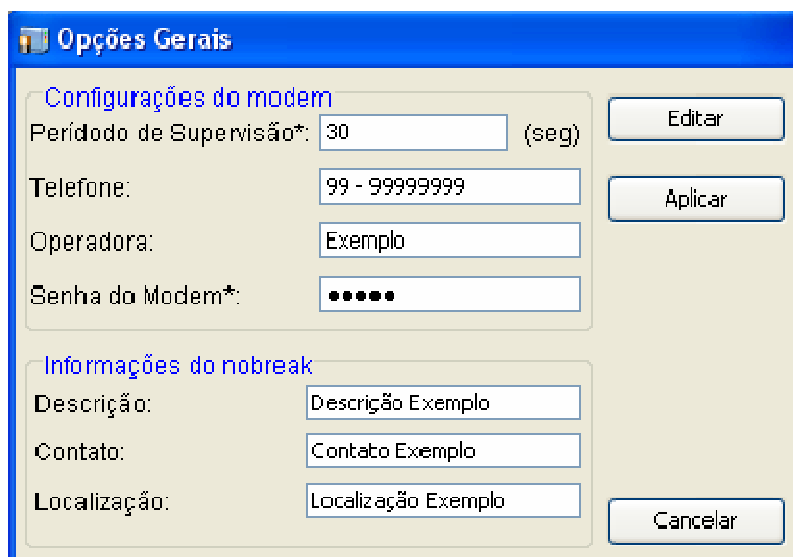


**Figura 12. Tela inicial do Software de Controle**

As subseções a seguir abordarão as telas e funcionalidades dos botões e menus que o usuário pode acessar.

#### **4.2.1. Tela Configurações Gerais**

Nesta tela, as configurações gerais do sistema serão definidas. A maioria delas serve apenas para informação a quem utiliza o programa, como “Telefone”, “Operadora”, “Descrição”, “Contato” e “Localização”. Os campos que são realmente importantes para o sistema são “Senha” e “Período de supervisão”. A Figura 13 mostra a tela de configurações gerais:



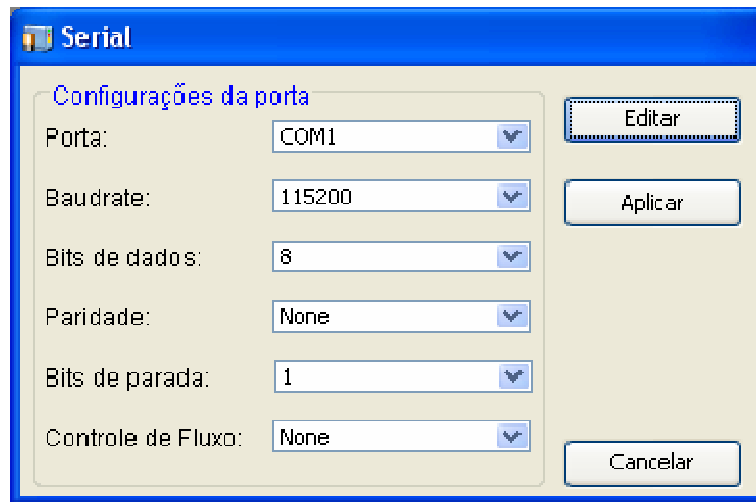
**Figura 13. Tela Configurações Gerais**

O campo “Período de supervisão” indica o período em que o modem irá obter o status do *nobreak* sob monitoramento. A cada intervalo de tempo definido, o modem irá requisitar o status, checará se há algum alarme ativo e verificará também as mensagens recebidas. Ou seja, esse é o período em que ciclo principal do modem é ativado.

O campo senha também é utilizado pelo modem. Sempre que o usuário envia uma requisição SMS, a senha é a primeira palavra que deve aparecer no texto da mensagem (ver 4.1.6).

#### **4.2.2. Tela Configurações da Serial**

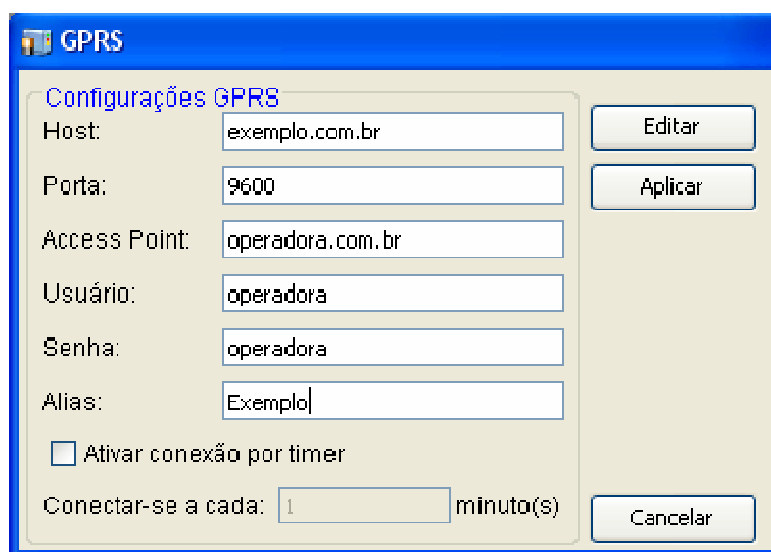
Nesta tela serão configuradas as propriedades da porta serial que o PC utilizará para se comunicar com o modem. As propriedades a serem configuradas são comuns a todas as portas seriais: “*Baudrate*”, “Bits de dados”, “Bits de parada”, “Paridade”, “Controle de fluxo” e a “Porta” utilizada. A Figura 14 mostra a tela de configuração serial:



**Figura 14. Tela Configurações da Serial**

#### **4.2.3. Tela Configurações GPRS**

Através desta tela serão configurados os atributos a serem utilizados pelo modem para iniciar uma comunicação GPRS com o Software de Controle. A Figura 15 mostra a tela de configurações GPRS:



**Figura 15. Tela Configurações GPRS**

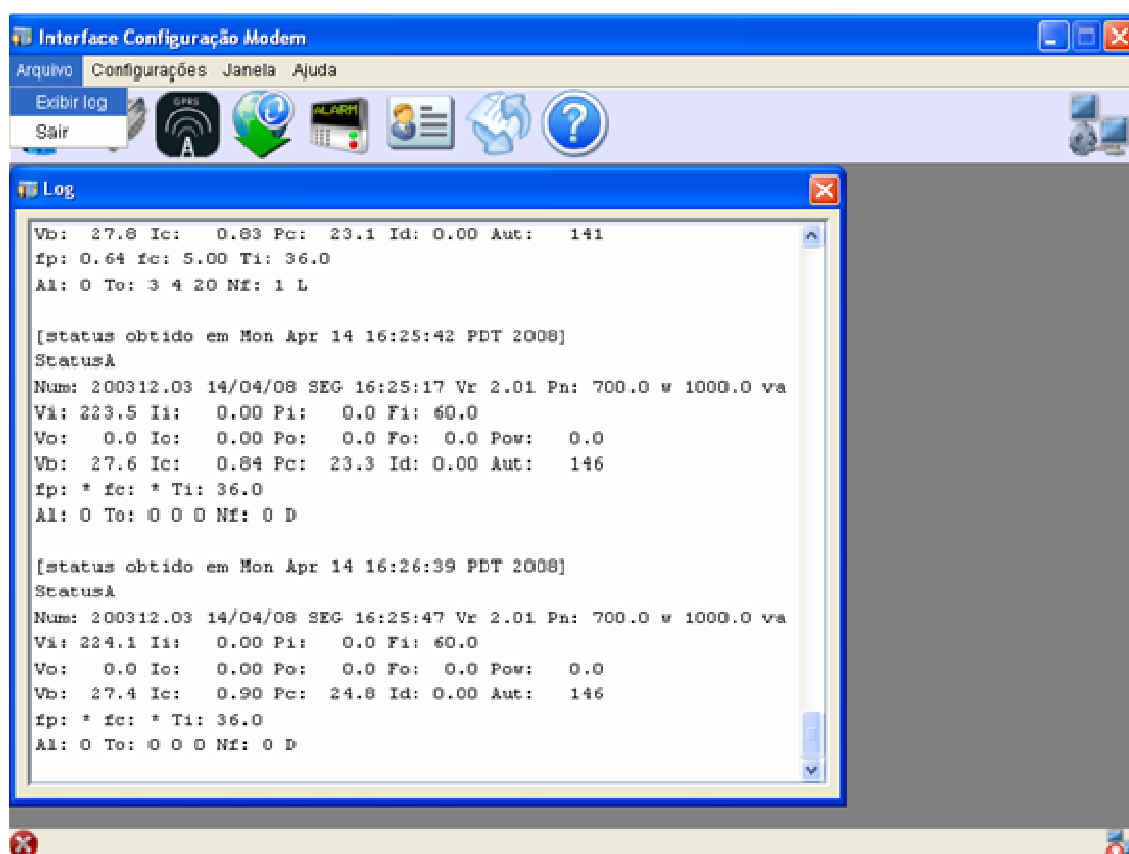
Os campos “Host” e “Porta” são utilizados pelo software, que atua como um servidor GPRS ao qual o módulo de comunicação irá se conectar. Já os campos “Access Point”, “Usuário” e “Senha” serão utilizados pelo modem e dependem da operadora do SIMCARD utilizado no módulo. A Tabela 3 mostra como configurar estes atributos corretamente:

**Tabela 3. Atributos para conexão GPRS de cada operadora.**

<b>Operadora</b>	<b>Access Point</b>	<b>Usuário</b>	<b>Senha</b>
Claro	claro.com.br	claro	claro
Oi	gprs.oi.com.br	oi	oi
Tim	tim.br	tim	tim
Vivo	zap.vivo.com.br	vivo	vivo

O campo “Alias” é utilizado para o envio de SMS requisitando uma conexão GPRS (ver 4.1.6).

A opção, “Ativar conexão por timer” também é utilizada pelo modem, caso esta opção esteja marcada, o modem irá se comunicar, via GPRS, a cada intervalo de tempo definido pelo usuário, enviando o último status obtido. O programa criará um arquivo de *log*, armazenando todos os status recebidos. Este arquivo pode ser visualizado no próprio programa através do menu “Arquivo”. A Figura 16 mostra o caminho e a tela do *log*:



**Figura 16. Abrindo a tela de log**

#### 4.2.4. Tela Configurações OTAP

Como já foi explicado na seção 3.3.5, tudo que é necessário para que o modem se atualize com a tecnologia OTAP é do endereço do servidor *web* onde estão os arquivos necessários, e do nome do arquivo *jad*. Portanto, esta tela contém apenas estes dois campos, como mostra a Figura 17:

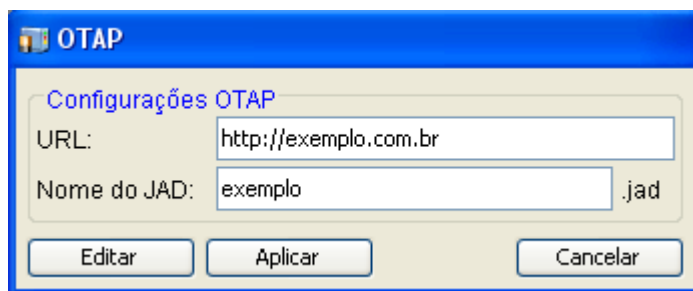


Figura 17. Tela Configurações OTAP

O importante de lembrar é que no campo “URL” (endereço do servidor) deve-se começar com “http://” e que no campo “Nome do JAD” não se deve colocar a extensão.

#### 4.2.5. Tela de Configuração de Alarmes

Cada *nobreak* possui uma diversidade de alarmes que ele pode disparar caso algo esteja errado com o seu funcionamento. Neste sistema, cada um destes alarmes terá um ou mais níveis associados a ele, que determinarão quem serão os responsáveis por este alarme. É nesta tela que os usuários do sistema definirão os níveis de cada alarme.

Este sistema foi desenvolvido para um grupo de *nobreaks* específico, por isso, os alarmes que são apresentados aqui estão relacionados com os equipamentos escolhidos para o estudo de caso. Um trabalho futuro seria permitir aos usuários editar os alarmes que podem ser disparados.

A Figura 18 mostra a tela em questão:

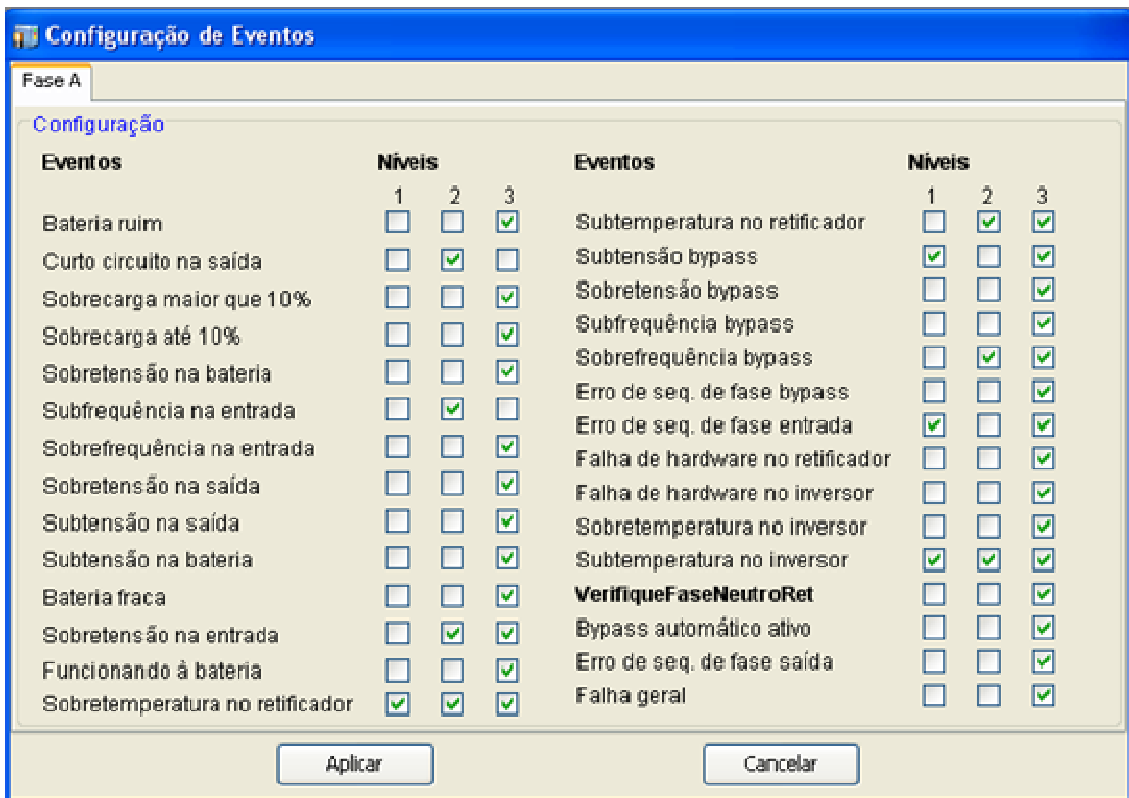


Figura 18. Tela de Configuração de Alarmes

Na Figura 18 está exemplificada os alarmes de um *nobreak* monofásico. Caso o *nobreak* sob monitoramento seja trifásico<sup>21</sup>, o usuário deverá definir isto através do programa, como mostra a Figura 19:

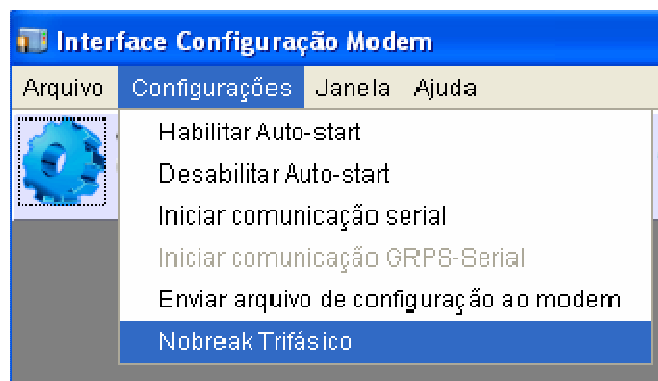


Figura 19. Configurando o programa para *nobreaks* trifásicos

Neste caso, a tela de configuração de alarmes conterà três abas, uma para cada fase, e o modem irá obter o status de todas as fases, como mostra a Figura 20.

<sup>21</sup> *Nobreak* Trifásico – Possui três fases, como se fossem três *nobreaks* diferentes.

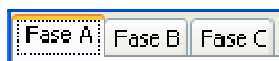


Figura 20. Abas de alarmes para *nobreaks* trifásicos

#### 4.2.6. Tela Configurar Usuários

Os usuários do sistema poderão ser adicionados, removidos ou terem seus dados editados nesta tela. A Figura 21 mostra a tela de configuração de usuários:

Figura 21. Tela Configurações de Usuários

No quadro a esquerda (Lista de usuários) estarão listados todos os usuários cadastrados no sistema, um clique no nome do usuário irá carregar seus dados no quadro ao lado (Informações). Os campos com um '\*' são campos cujo preenchimento é obrigatório para se adicionar ou editar um usuário, sem eles o sistema poderia não funcionar.

O campo “Nível” determina as responsabilidades do usuário. Cada alarme que pode ser disparado pelo *nobreak* tem um ou mais níveis associados (ver Figura 18). Os níveis de cada alarme representam quais usuários serão responsáveis por ele. Por exemplo, se um alarme nível 2 for disparado, todos os usuários cujo nível seja igual a 2 serão alertados deste disparo (através de uma mensagem SMS). Caso um alarme tenha os níveis 2 e 3, todos os usuários com nível 2 e todos os usuários com nível 3 receberão a mensagem informando sobre o ocorrido.

O campo “Login” é utilizado para recebimento de SMS com o status do *nobreak* (ver 4.1.6) e o campo “Celular” é utilizado pelo modem, que enviará o SMS para este número.

#### 4.2.7. Habilitando e desabilitando o Autostart

O *Autostart* é uma funcionalidade do módulo de comunicação que indica se a aplicação contida no modem será iniciada automaticamente assim que o modem for ligado e qual aplicação será iniciada. Habilitar ou desabilitar o *Autostart* é feito através do envio de um comando AT ao módulo (ver Apêndice A – Comandos AT). No sistema, esta função pode ser habilitada ou desabilitada através do Software de Controle no menu “Configurações”, como apresentada na Figura 22:

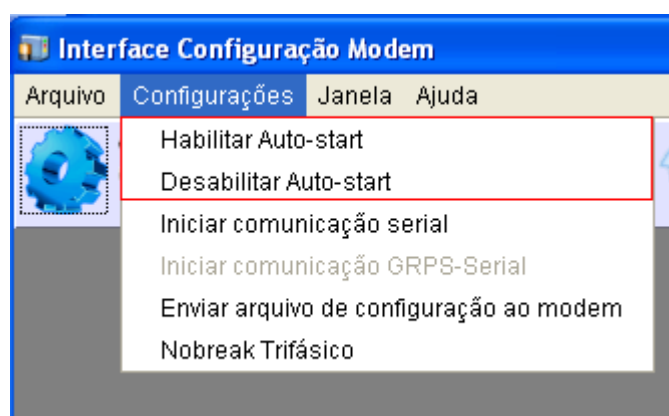


Figura 22. Habilitar ou Desabilitar o *Autostart*

Para desabilitar o *Autostart* um processo deve ser seguido. Primeiramente, o modem deverá estar desligado. Com o modem desligado e o programa rodando, clica-se no menu "Desabilitar Auto-start" e em seguida liga-se o modem.

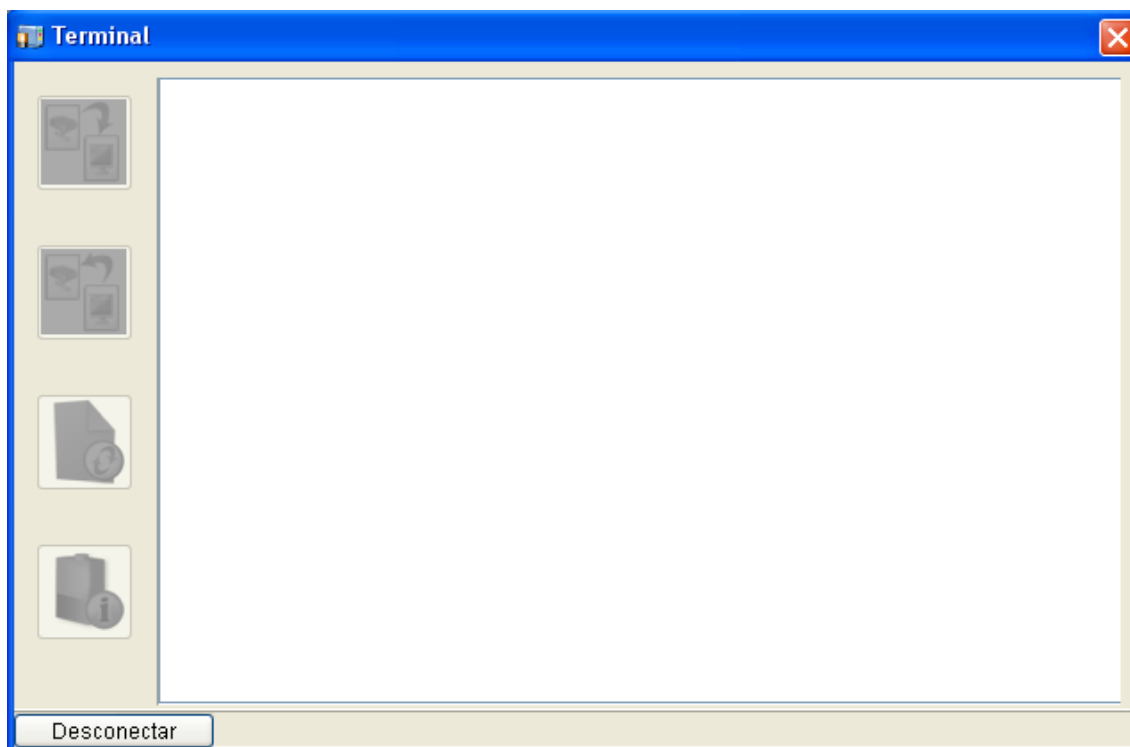
Este procedimento é necessário, pois, assim que alguma aplicação é iniciada no módulo de comunicação, ele não receberá mais comandos AT pela porta serial, e sim apenas os que a própria aplicação enviar, portanto, para se desabilitar o *Autostart*, é necessário enviar o comando antes que a aplicação seja iniciada. Existe um pequeno intervalo entre a alimentação do modem e o início da aplicação, aproximadamente 3 segundos, e é durante este intervalo que o comando AT deve ser enviado.

#### 4.2.8. Terminal

Esta janela é utilizada para controlar o modem remotamente utilizando a tecnologia GPRS. Através dela o usuário pode enviar um novo arquivo de configuração ao modem, obter o arquivo de configuração do modem, obter o status do *nobreak* ou carregar as

configurações do modem no arquivo de configuração do sistema (para isto o arquivo de configuração do modem deve ter sido obtido).

Ela só poderá ser utilizada a partir do momento em que o modem se conectar ao servidor do programa, para tal envia-se um SMS ao modem requisitando a conexão (ver 4.1.6). Todos os botões estarão desabilitados até que a conexão esteja ativa, como mostra a Figura 23:



**Figura 23. Terminal para controle remoto via GPRS**

A partir do momento em que a conexão GPRS for estabelecida, o usuário poderá utilizar esta janela normalmente, clicando nos botões o quanto quiser. A janela poderá inclusive ser fechada e reaberta e a conexão não será desfeita. Para fechar a conexão, o usuário precisará clicar no botão “Desconectar”. Mas, como já fora visto (seção 3.4), não há problemas em relação a custos caso o usuário esqueça de desconectar o GPRS, já que a tarifa é cobrada pela quantidade de dados enviados e não pelo tempo de conexão. O único problema nesse caso seria pelo fato de que nenhuma outra conexão poderia ser feita ao programa.

#### 4.2.9. Conexões GPRS e Serial

O sistema apresentado permite dois tipos de conexão GPRS. O primeiro é para controle e monitoramento remoto do *nobreak*, acessado pela janela Terminal do programa (ver 4.2.8), e o segundo é utilizado quando o usuário deseja se comunicar diretamente com o *nobreak*, fazendo com que o modem funcione como uma ponte de comunicação entre o PC e o *nobreak* (este segundo tipo de comunicação GPRS é chamado de “comunicação em modo ponte”). Todos os comandos que o usuário enviar através do *software* serão repassados pelo modem ao *nobreak*, e a resposta deste será enviada ao programa, para a sua visualização. A Figura 24 mostra a tela utilizada para o segundo modo GPRS.

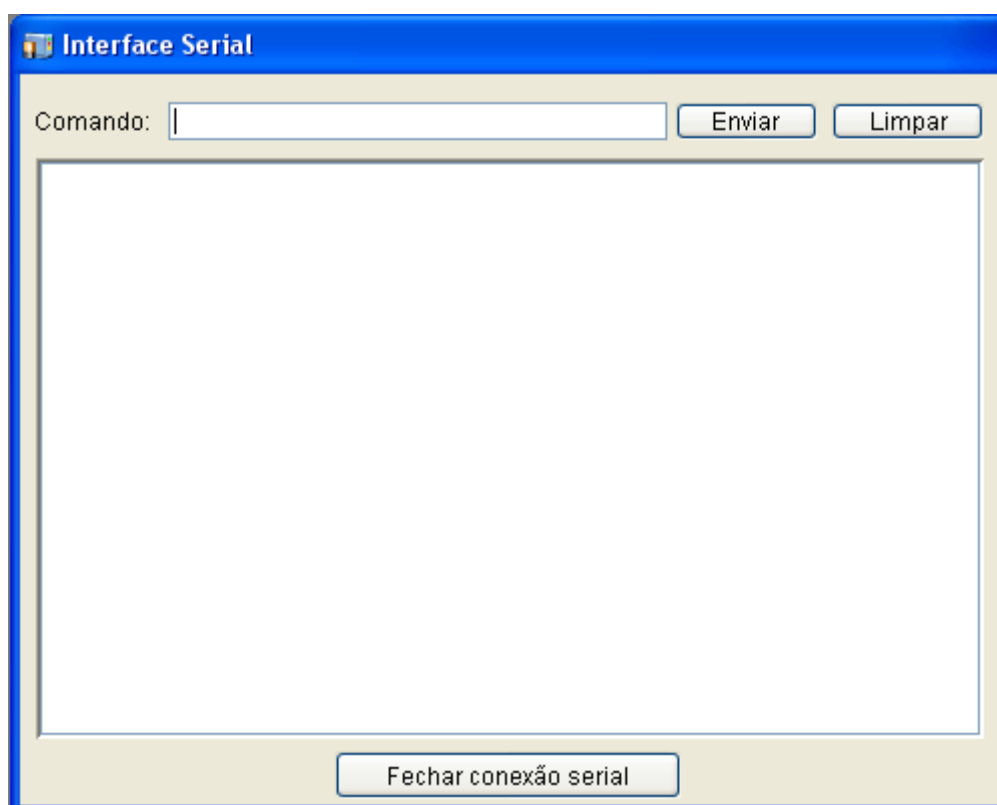


Figura 24. Tela para funcionamento de Ponte do modem

Esta tela também é utilizada para uma comunicação em ponte através da porta serial. Resumindo, o programa possibilita a comunicação direta com o *nobreak* através de comunicação local (via serial) ou remota (via GPRS). Em ambos os casos, a tela e o funcionamento da mesma são idênticos.

Em relação às comunicações por GPRS, há uma forma de detectar que o modem se conectou ao programa. Sem isso, o usuário teria que adivinhar quando isso acontecesse.

Na tela do programa, existe um ícone no canto inferior direito que indica se o modem está conectado ou não ao programa. A Figura 25 mostra as imagens do ícone.



**Figura 25. Imagens do ícone de conexão GPRS**

A imagem da esquerda é a imagem inicial do programa, indicando que nenhuma conexão GPRS foi detectada. Assim que o modem se conectar ao programa o ícone do meio aparecerá piscando no lugar do anterior. A imagem da direita representa que a conexão foi detectada e o usuário já está ciente, ele abriu a janela para usar a conexão.

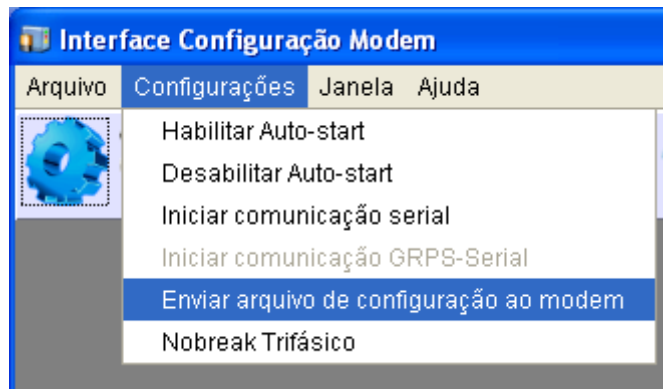
Nos dois modos de conexão GPRS, esse esquema de imagem funciona. Mas como o usuário saberá qual tipo de conexão GPRS está ativa? Os dois modos de conexão são distintos. Cada um leva a aplicação do modem a uma rotina diferente, portanto, caso o usuário queira utilizar o primeiro modo, ele não deverá abrir a janela de comunicação em modo ponte, e vice-versa. O programa não dá suporte a isto, o usuário é que deverá abrir a janela correta de acordo com a mensagem que ele enviou (como visto, as mensagens são diferentes para os dois modos). Utilizar a janela errada acarretará no mau funcionamento do *software*.

#### **4.2.10. Enviar Arquivo de Configuração ao modem**

Sempre que alguma alteração for realizada nos atributos do Arquivo de Configuração pelo Software de Controle, o arquivo que está no modem estará “desatualizado”. As alterações realizadas no software não serão enviadas automaticamente ao modem, mas isso pode ser feito através do próprio programa, de dois modos, GPRS ou Serial.

O modo GPRS já foi apresentado, utilizando a janela Terminal (ver 4.2.8).

Caso o PC e o módulo de comunicação estejam conectados por um cabo serial adequado, o usuário poderá enviar o arquivo de configuração através do menu “Configurações” do programa, como mostra a Figura 26.



**Figura 26. Enviando um novo Arquivo de Configuração ao modem**

Assim que alguma alteração for realizada, um ícone no canto inferior esquerdo da tela do programa ficará piscando, informando que alterações foram feitas e o novo arquivo não foi enviado ao modem. A Figura 27 mostra as imagens do ícone.



**Figura 27. Imagens do ícone de alterações no Arquivo de Configuração**

A imagem da esquerda é a inicial, quando nenhuma alteração foi realizada. A da direita informa das alterações.

Ao receber as alterações o modem irá carregar as novas propriedades e levará um tempo até se tornar operável novamente, portanto, o usuário só deverá enviar o novo arquivo de configuração quando tiver feito todas as mudanças desejadas, evitando ficar enviando novos arquivos com maior frequência.

## 5. Conclusão

Este documento teve como objetivo apresentar o *framework* desenvolvido neste trabalho que representa um sistema de telemetria utilizando a rede GSM para monitoramento e controle de equipamentos, assim como mostra as tecnologias que podem ser utilizadas para tal funcionalidade.

Como caso de estudo para testes, os equipamentos utilizados foram *nobreaks*, que possuem as características adequadas ao sistema.

Através de uma pesquisa inicial, foram apresentadas as tecnologias utilizadas no desenvolvimento, assim como suas vantagens e desvantagens e realizando comparações entre elas próprias, assim como com outras tecnologias existentes.

Dois módulos foram desenvolvidos no trabalho, uma aplicação embarcada, bastante genérica, que pode ser utilizada em qualquer situação onde o sistema possa ser aplicado, e um Software de Controle, específico para o estudo de caso, mas que pode ser facilmente alterado para outras necessidades.

Como contribuições, este trabalho apresenta ao usuário vários detalhes a respeito do módulo TC65 que não são facilmente encontrados, assim como apresenta um método mais prático para a utilização da tecnologia OTAP.

O trabalho trás também um *Framework* de telemetria mais barato, robusto e flexível do que os estudados durante o levantamento bibliográfico, além de permitir mais funcionalidades.

### 5.1. Dificuldades Encontradas

Durante o desenvolvimento do *framework*, algumas dificuldades foram encontradas, o que retardou um pouco o processo de desenvolvimento. O objetivo desta seção é apresentar as dificuldades para que futuramente, elas possam ser evitadas.

As dificuldades foram:

- A principal dificuldade durante o desenvolvimento do sistema foi em relação ao cabo serial que precisa ser utilizado na comunicação com o módulo TC65, pois não havia documentação sobre este e um detalhe deve ser levado em consideração, isto é mais bem explicado no Apêndice C - Interfaces.

- Durante os testes da tecnologia GPRS do módulo TC65, foram encontradas dificuldades pelo fato de não se conhecer bem como era feita este tipo de comunicação com o modem, um grande problema foi descobrir a necessidade da utilização dos atributos de conexão GPRS de cada operadora, apresentados na Tabela 3. Um tempo foi perdido até se descobrir sobre a necessidade da utilização destes atributos de conexão de cada operadora para iniciar a conexão.
- Na utilização da tecnologia OTAP, é necessário o envio de duas mensagens SMS (ver [19]) extensas e complexas para que o modem inicie o processo. Os testes com o envio das mensagens não estavam funcionando então foi utilizada uma outra forma que também realiza a requisição para OTAP, atingida através do envio de um comando AT ao modem. Desta forma, a mensagem que precisa ser enviada foi simplificada e a própria aplicação embarcada envia o comando AT (ver 4.1.4 e 4.1.6).
- Ainda na utilização da tecnologia OTAP, algum tempo foi desperdiçado para que fosse descoberto que era necessário alterar o arquivo *Jad* da aplicação para acrescentar o endereço do arquivo *Jar* (ver 4.1.4).

Durante o desenvolvimento do Software de Controle, na implementação do servidor GPRS, foram encontrados problemas, pois o servidor encerrava a conexão após passar um tempo sem receber requisições, algumas opções devem ser editadas para que este mantenha a conexão sempre ativa. A opção a ser editada é a “*time out*”.

## 5.2. Trabalhos Futuros

Como trabalhos futuros são citados:

- Melhorias na interface do Software de Controle;
- Aprimorar as conexões GPRS, integrando as formas de comunicação. Atualmente, os usuários podem se comunicar via GPRS de dois formatos diferentes e distintos, a idéia é unificar a conexão, permitindo aos usuários realizar apenas uma conexão tanto para trabalhar o modem como ponte, como para controlá-lo remotamente;
- Realizar o envio de status ao usuário (quando requisitado via SMS) utilizando GPRS, barateando os custos, já que via GPRS é mais barato do que via SMS.

- Utilização de um banco de dados onde serão armazenadas as propriedades do sistema, deixando apenas o modem utilizando o arquivo de configuração. O Software seria alterado para montar o arquivo de configuração antes de enviar os dados ao modem.
- Permitir ao usuário configurar a comunicação serial entre modem e *nobreak*. Atualmente o usuário pode configurar a comunicação entre modem e PC, sendo a comunicação entre modem e *nobreak* uma comunicação padrão, tendo o *nobreak* que se ajustar a ela.

## Referências

- [1] GSM World. Disponível em: <http://www.gsmworld.com/>. Acesso em: 07 de março de 2008.
- [2] WirelessBR. Disponível em: <http://www.wirelessbrasil.org/>. Acesso em: 18 de abril de 2008.
- [3] Samanji, A. - General Packet Radio Service (GPRS)  
Disponível em: <http://ieeexplore.ieee.org/iel5/45/21535/00997970.pdf>. Acesso em: 10 de março de 2008.
- [4] Java 2 Platform Micro Edition (J2ME) Technology for Creating Mobile Devices. White Paper. Sun, 2000.  
Disponível em <http://www.java.sun.com/>. Acesso em: 07 de outubro de 2005
- [5] Dias, K. L.; Fontes, W. P. – Desenvolvimento de Aplicações Para Dispositivos Móveis utilizando a plataforma J2ME.
- [6] ALA 2 – Sistema de alarme GSM “SMS”. Disponível no site:  
<http://www.prodelin.com.br/MANUAIS/ALARME/ALA2CAT.PDF>. Acesso em: 10 de Janeiro de 2008.
- [7] RASMANAGER. Disponível no site:  
[http://www.generex.de/generex/download/datasheets/datasheet\\_RASMANAGER\\_en.pdf](http://www.generex.de/generex/download/datasheets/datasheet_RASMANAGER_en.pdf)
- [8] Nobreaks Masterys. Disponível em:  
[http://www.socomec.fr/masterys/p\\_ups\\_masterys\\_communication\\_esc2\\_uk.htm](http://www.socomec.fr/masterys/p_ups_masterys_communication_esc2_uk.htm)
- [9] FOXFIX GPRS. Disponível no site:  
[http://www.foxsystems.com.br/foxfix\\_desc.pdf](http://www.foxsystems.com.br/foxfix_desc.pdf). Acesso em: 10 de Janeiro de 2008.
- [10] Rastreador GSM full. Disponível em:  
<http://www.globalsafe.com.br/gsmfull.htm>  
Acesso em: 10 de Janeiro de 2008.
- [11] LocaBox. Disponível em:  
[http://www.itecdiffusion.com/BR/rastreamento\\_localizacao\\_gps\\_gsm\\_gprs.html](http://www.itecdiffusion.com/BR/rastreamento_localizacao_gps_gsm_gprs.html)  
Acesso em: 10 de Janeiro de 2008.
- [12] HunterPro. Disponível em: <http://www.gpshunter.com/index.htm>. Acesso em: 10 de Janeiro de 2008.

- [13] S. Constantin, F. Moldoveanu, R. Campeanu, I. Baciú, S. M. Grigorescu, B. Carstea, and V. Voinea. GPRS Based System for Atmospheric Pollution Monitoring and Warning.
- [14] SCHAEFER, S. Secure Trade Lane: a sensor network solution for more predictable and more secure container shipments.
- [15] Oliver Holland, A. Hamid Aghvami. Dynamic Switching between One-to-Many Download Methods in "All-IP" Cellular Networks.
- [16] Tore Fjellheim. Over-the-air Deployment of Applications in Multi-Platform Environments.
- [17] H. Dubois-Ferrière, L. Fabre, R. Meier e P. Metrailler. TinyNode: a comprehensive platform for wireless sensor network applications.
- [18] Siemens AT Command Set. Disponível em:  
[http://www.automation.siemens.com/download/internet/cache/3/1431392/pub/de/TC65\\_atc\\_v02000.pdf](http://www.automation.siemens.com/download/internet/cache/3/1431392/pub/de/TC65_atc_v02000.pdf)
- [19] TC65 AC75 JAVA User's Guide, Siemens, Versão 08.
- [20] TC65 Release Notes, Siemens, Versão 02.
- [21] TC65 Hardware Interface Description, Siemens, Versão 1.041.

## Apêndice A – Comandos AT

Neste apêndice serão apresentados os comandos AT que foram utilizados no projeto. Para uma lista mais geral sobre estes comandos ver [18]. Comando AT são comandos que podem ser enviados ao módulo TC65 que fazem com que o mesmo realize alguma operação. Essa operação pode ser a definição de alguma propriedade, o envio de alguma mensagem ou até mesmo uma chamada para um número qualquer. A maior parte dos comandos existentes não foi utilizada neste projeto, os que foram estão descritos a seguir:

- AT^SJOTAP

O comando acima é usado para controlar e iniciar operações OTAP para instalação e atualização de aplicações Java. Este comando requer vários parâmetros para sua inicialização. Entre estes parâmetros encontram-se, nome de usuário e senha, tipo de conexão, operadora do SIMCARD do modem, endereço do servidor *web* onde o modem irá buscar a aplicação a ser instalada, o diretório onde será instalada a aplicação, entre outros que não são usados no caso deste projeto. A referência para este comando encontra-se na página 523 de [18];

- AT+CMGF

Este é o comando que define se o módulo TC65 irá trabalhar no modo Texto ou no modo PDU para leitura de mensagens SMS. Este comando recebe apenas um parâmetro que indica o modo, sendo '0' para o modo PDU e '1' para o modo Texto.

Ex: "AT+CMGF=1" define que o modem trabalhará no modo Texto.

Referência na página 365 de [18];

- AT+CMGS

Comando responsável por enviar uma mensagem SMS a um número qualquer. Este comando depende do comando anterior, pois a mensagem a ser enviada deve estar no formato definido. No caso do modo Texto (usado neste projeto), o parâmetro passado para o comando é o número do telefone ao qual a mensagem será enviada, neste número deve conter também o DDD do telefone. Por exemplo, para enviar uma mensagem a um número de Recife, é enviado ao modem o comando

“AT+CMGS=”81XXXXXXXXXX””, onde os ‘X’ representam os números do telefone. É importante notar que o DDD+número encontra-se entre aspas, é assim que o comando deve ser enviado ao módulo. Após o envio do comando, o texto da mensagem deverá ser enviado. Para finalizar o texto, se envia o código ASCII de “ctrl+z”, que é, em hexadecimal, “001A”. Referência na página 370 de [18].

- AT+CMGR

Este comando é utilizado para fazer a leitura das mensagens SMS recebidas pelo modem. Mais uma vez, este comando depende do formato de mensagem definido. O parâmetro passado para este comando representa a posição de memória da mensagem no modem. Se o parâmetro seja maior do que o limite da memória, uma mensagem de erro é retornada pelo modem. Caso contrário, o modem retornará a mensagem que se encontra na posição indicada no formato definido. Referência na página 368 de [18].

- AT+CMGD

Comando que apaga as mensagens da memória do modem. Recebe como parâmetro a posição da mensagem, assim como no comando acima. Posições inexistentes acarretarão em mensagens de erro. Referência na página 364 de [18].

- AT+ICF

Este comando é responsável por definir as características que a porta serial do módulo terá. Essas características são: bits de dados, de parada e de paridade. Recebe apenas um parâmetro, um inteiro, que define as três características de acordo com a tabela abaixo:

<b>Inteiro</b>	<b>Configuração</b>
1	8 bits de dados, 0 de paridade, 2 de parada
2	8 bits de dados, 1 de paridade, 1 de parada
3	8 bits de dados, 0 de paridade, 1 de parada
5	7 bits de dados, 1 de paridade, 1 de parada

**Tabela 4. Descrição do parâmetro para o comando AT+ICF**

A Referência deste comando pode ser encontrada na página 108 de [18].

- AT\Q

Este comando também se refere à comunicação serial a exemplo do anterior. Ele é responsável por definir o controle de fluxo usado para a comunicação serial. Geralmente, os parâmetros passados para os comandos vêm depois de um '=', como em "AT+GMGF=1". No caso deste comando em particular, o parâmetro é passado depois de uma '\'. Por exemplo: "AT\Q0". O número '0' neste caso é o parâmetro. A tabela abaixo mostra os possíveis parâmetros e seus resultados:

<b>Inteiro</b>	<b>Controle de Fluxo</b>
0	Nenhum, desativa o controle de fluxo.
1	Controle por <i>software</i> Xon/Xoff.
2	Controle com CTS <sup>22</sup> por DTE <sup>23</sup> .
3	Controle por <i>hardware</i> RTS <sup>24</sup> /CTS.

**Tabela 5. Descrição do parâmetro para o comando AT\Q**

A referência para este comando encontra-se na página 103 de [18].

- AT^SCFG

Comando responsável por configurações gerais do modem. No projeto em questão, o comando é utilizado apenas para habilitar ou desabilitar o *auto-start* do modem. Recebe dois parâmetros, o primeiro que é padrão, "Userware/Autostart" e o segundo que indica se está habilitando ou desabilitando o *auto-start* (0 desabilita, 1 habilita). Por exemplo, para habilitar o *auto-start*, o seguinte comando é enviado ao modem: "AT^SCFG="Userware/Autostart",1". Mais uma vez, as aspas são obrigatórias. A referência para o comando pode ser encontrada na página 56 de [18].

- AT^SJRA

Este último comando não é usado na aplicação, mas foi bastante usado durante o desenvolvimento e os testes do projeto. A sua função é de iniciar alguma aplicação que se encontre na memória do modem. O parâmetro, único, passado, é o caminho do arquivo *jad* que está no modem, também entre aspas. Por exemplo: "AT^SJRA="a:/modem.jad"". A referência para o comando pode ser encontrada na página 519 de [18].

<sup>22</sup> CTS (*Clear To Send*) – Bit da serial que indica que o equipamento está pronto para receber dados.

<sup>23</sup> DTE (*Data Terminal Equipment*) – Equipamento que converte informações do usuário em dados a serem transmitidos.

<sup>24</sup> RTS (*Ready To Send*) – Bit da serial que indica que o dispositivo está pronto para enviar dados.

## Apêndice B – Arquivo de Configuração

Como já foi dito, o *software* de controle da aplicação embarcada funciona baseado em um arquivo de configuração, de onde ele tira as informações para configurar a si próprias e ao modem. Este arquivo também é usado pelo modem para definir suas propriedades.

Segue abaixo um exemplo do arquivo de configuração que o modem e a interface usam. Tudo que estiver após ‘--’ significam comentários adicionados neste documento :

```
#####
#      ConfiguracoesModem      #
#####
PeriodoSupervisao: 30 --período de cada ciclo
Telefone: xxxxxxxx --os 'x' representam números
Operadora: UFPE
SenhaModem: teste
Descricao: Modem
Contato: rbsm@cin.ufpe.br
Localizacao: CIn

#####
#      ConfiguracaoSerial      #
#####
Porta: COM1
Baudrate: 115200
BitsDados: 8
ControleFluxo: None
Paridade: None
BitsParada: 1

#####
#      Usuarios      #
#####
Lista:
Renato

Nome: Renato
Telefone: xxxxxxxx --os 'x' representam números
Celular: xxxxxxxxxxxx --os 'x' representam números, colocar o DDD
Email: rbsm@cin.ufpe.br
Nivel: 1
Setor: teste
Descricao: exemplo
Login: rbsm

#####
#      ConfiguracoesGPRS      #
#####
Host: gprs.ufpe.br
PortaGPRS: 9600
AccessPoint: gprs.oi.com.br
User: oi
Password: oi
```

Alias: gprs  
Timer: nao  
Tempo: 1

```
#####  
# ConfiguracoesOTAP #  
#####  
URL: http://gprs.ufpe.br:8080/update  
JAD: modem
```

```
#####  
# Alarmes #  
#####
```

StatusA

AL1: 1  
AL2: 1  
AL3: 1  
AL4: 1  
AL5: 1  
AL6: 1  
AL7: 1  
AL8: 1  
AL9: 1  
AL10: 1  
AL11: 1  
AL12: 1  
AL13: 1  
AL14: 1  
AL15: 1  
AL16: 1  
AL17: 1  
AL18: 1  
AL19: 1  
AL20: 1  
AL21: 1  
AL22: 1  
AL23: 1  
AL24: 1  
AL25: 1  
AL26: 1  
AL27: 1  
AL28: 1  
AL29: 1

StatusB

AL1: 2  
AL2: 2  
AL3: 2  
AL4: 2  
AL5: 2  
AL6: 2  
AL7: 2  
AL8: 2  
AL9: 2  
AL10: 2  
AL11: 2  
AL12: 2  
AL13: 2  
AL14: 2  
AL15: 2  
AL16: 2

```
AL17: 2
AL18: 2
AL19: 2
AL20: 2
AL21: 2
AL22: 2
AL23: 2
AL24: 2
AL25: 2
AL26: 2
AL27: 2
AL28: 2
AL29: 2
```

```
StatusC
```

```
AL1: 3
AL2: 3
AL3: 3
AL4: 3
AL5: 3
AL6: 3
AL7: 3
AL8: 3
AL9: 3
AL10: 3
AL11: 3
AL12: 3
AL13: 3
AL14: 3
AL15: 3
AL16: 3
AL17: 3
AL18: 3
AL19: 3
AL20: 3
AL21: 3
AL22: 3
AL23: 3
AL24: 3
AL25: 3
AL26: 3
AL27: 3
AL28: 3
AL29: 3
```

```
--statusB e statusC podem não existir, depende do número de
fases --do nobreak
```

```
EOF --end of file, todo arquivo deve ter isto no final
```

```
--final do arquivo.
```

No projeto são considerados *nobreaks* trifásicos, que possuem 3 fases (StatusA, StatusB e StatusC representam cada uma das fases). Caso o *nobreak* fosse monofásico, o arquivo de configuração não teria “StatusB” nem “StatusC”, pois representam as outras fases do *nobreak*.

Caso o arquivo tenha sido alterado para um formato que o programa não aceita, um erro será emitido e o programa será fechado.

## Apêndice C - Interfaces

Um detalhe importante do *framework* são as interfaces de comunicação entre os periféricos que o compõe. Caso alguma conexão esteja incorreta, o sistema poderá não funcionar corretamente ou até não funcionar.

Esta seção apresenta as interfaces de comunicação modem-*nobreak* e modem-PC, abordando sobre os pinos do cabo serial, dos componentes do modem e das configurações necessárias para a comunicação.

- **Interface Modem-*Nobreak***

A interface entre o modem e o *nobreak* é realizada através da porta serial *asc0* do modem (a porta de trás do modem). A comunicação do modem com o *nobreak* usa um *baudrate*<sup>25</sup> que deve estar na mesma velocidade que o do *nobreak*, 8 bits de dados, um bit de parada e nenhum bit de paridade, e não usa controle de fluxo. Toda esta configuração é realizada com envio de comandos AT ao modem. Essas configurações podem mudar caso o equipamento sob monitoração seja alterado. Para os *nobreaks* usados nos testes realizados no trabalho, esta é a configuração necessária, mas isso não quer dizer que todos os *nobreaks* precisem desta configuração.

Como já foi mencionada, a interface serial do modem é diferente de outras interfaces encontradas em outros dispositivos. Geralmente se utilizam cabos seriais que invertem os pinos '2' e '3' entre suas extremidades. No caso do TC65, os pinos '2' e '3', que correspondem ao RX (receptor) e o TX (transmissor), são invertidos no módulo, ou seja, o cabo serial necessário não deve inverter. O pino '5' do cabo (GND, o terra da porta) também tem que estar ligado (pino '5' no pino '5'). Outra peculiaridade é que, para usar um cabo serial de apenas 3 pinos (RX, TX e GND) é necessário que a ponta do cabo que estará ligada ao modem esteja com os pinos '4' e '6', e '7' e '8' conectados, para que o modem faça um controle de fluxo interno. Outras tentativas de comunicação do modem usando cabos de 3 pinos não funcionarão se o pino '4' não estiver ligado ao pino '6', ou o pino '7' não estiver ligado ao pino '8'. A Figura 2 mostra um esquemático de como deverá ser um cabo serial para utilizar no *framework*.

---

<sup>25</sup> *Baudrate* – Velocidade de comunicação da porta serial.

Também como já foi dito, na seção 3.1, pode-se usar um cabo de 9 pinos paralelos, não sendo necessário nenhum ajuste. A Figura 28 representa o cabo com os 9 pinos paralelos:

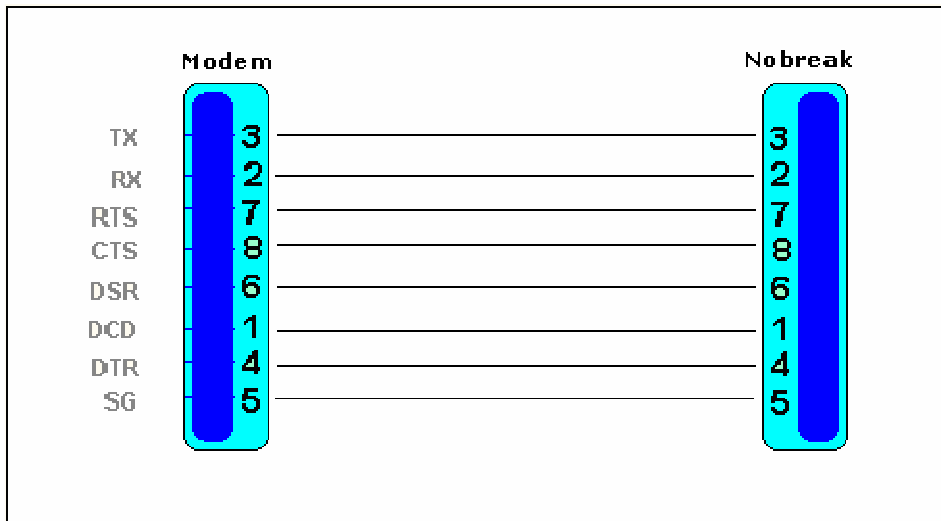


Figura 28. Esquemático de um cabo com 9 pinos paralelos

- **Interface Modem-PC**

A comunicação entre o modem e o PC é realizada através da porta *asc1* do módulo TC65 (a porta que fica na parte frontal do modem). As configurações da porta são semelhantes às da porta *asc0*, a única diferença é o *baudrate*, que nesse caso deve estar de acordo com o *baudrate* do computador, sendo possível utilizar velocidades maiores do que na interface com o *nobreak*. O cabo necessário para a comunicação é um cabo serial de 9 pinos paralelos. Outros cabos podem ser usados, mas sempre levando em consideração a mudança que o módulo já faz em relação aos pinos do receptor e do transmissor.