

*Universidade Federal de Santa Catarina*  
*Centro Tecnológico*  
*Curso de Engenharia de Controle e Automação Industrial*

***"DESENVOLVIMENTO DE UM AMBIENTE PARA SUPERVISÃO DE  
ESTAÇÕES REMOTAS "***

*Monografia submetida à Universidade Federal de Santa Catarina*  
*como requisito para a aprovação na disciplina*  
***EEL 5901: Projeto de Fim de Curso***

***GUILHERME ERNANI VIEIRA***

*Florianópolis, Dezembro de 1994*

*"Desenvolvimento de um Ambiente para Supervisão de Estações Remotas "*

*Guilherme Ernani Vieira*

Esta monografia foi julgada no contexto da disciplina

**EEL 5901: Projeto de Fim de Curso**

e aprovada na sua forma final pelo

**Curso de Engenharia de Controle e Automação Industrial**

***Banca Examinadora:***

**Eng<sup>o</sup>. Gilson Luiz Pasternak**

Orientador na Empresa

**Prof<sup>o</sup>. Vitório Bruno Mazzola**

Orientador do Curso

**Prof<sup>o</sup>. Augusto Humberto Bruciapaglia**

Responsável pela disciplina e Coordenador do Curso

**Prof<sup>o</sup>. Marcelo Stemmer**

Avaliador pelo Curso

**Carlos Fernando Martins**

Acadêmico - Avaliador

**Claudiane Isabel Grando**

Acadêmico - Avaliador

Agradeço a Deus, a minha noiva e a minha família, aos professores e a  
Telecomunicações de Santa Catarina S/A,  
por mais esta etapa vencida.

## **RESUMO**

Este projeto tem como objetivo maior iniciar o desenvolvimento de um novo sistema de supervisão de estações remotas na Telecomunicações de Santa Catarina S/A (TELESC), que se diferencie do atual sistema, aplicando técnicas mais recentes de comunicação, gerenciamento e programação.

Partindo deste objetivo, desenvolveu-se então, no período de Agosto a Dezembro deste ano, um *software* de supervisão com características como: protocolo de comunicação por pacotes, gerenciamento modelado por redes de Petri e programação orientada a objetos.

Este ambiente é composto por três módulos básicos: (a) O *Supervisor*  $\Rightarrow$  programa a ficar residente na estação remota, responsável pela aquisição dos dados (leitura dos valores das grandezas sendo supervisionadas) e pela comunicação com o Concentrador; (b) O *Concentrador*  $\Rightarrow$  programa que concentra alguns supervisores. Este fica residente numa estação remota de grande porte e é responsável pela conexão do Gerente com o Supervisor; e (c) O *Gerente*  $\Rightarrow$  integra todo o sistema, comunicando-se com todas as estações via os Concentradores. Atende ao usuário do sistema, sem, contudo, deixar de supervisionar as estações, através de um pseudo-paralelismo.

O ambiente permite que o usuário verifique como estão operando todas as estações remotas do estado, observando se há ou não necessidade de manutenção (preventiva e/ou corretiva) em alguma delas.

## **ABSTRACT**

The first objective of this work is to start a development of a new remote center supervision system at Telecomunicações de Santa Catarina S/A (TELESC), which must be completely different from the current one by means of new technologies application, mainly in communication, management and programming concepts.

With these points in mind, from August to December, it was implemented a supervision software with characteristics like: package communication protocol, management modeled by Petri nets and oriented-object programming.

The above mentioned environment was implemented in three basic modules: (a) the *Supervisor*  $\Rightarrow$  permanently running in a remote center. It is responsible for the data acquisition and for the communication to the Centralizer; (b) the *Concentrador*  $\Rightarrow$  responsible for linking few supervisors and for the communication to the Manager; and (c) the *Gerente*  $\Rightarrow$  makes the connection among all supervisors via Centralizers. The Manager serves the system user without breaking the supervision function, through a *pseudo-parallelism*.

This supervision environment allows the system user to know how a remote center is at any time, in any place in the state of Santa Catarina.

## **SUMÁRIO**

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>INTRODUÇÃO .....</b>                          | <b>08</b> |
| <b>2</b> | <b>DIVISÃO DAS TAREFAS .....</b>                 | <b>09</b> |
| <b>3</b> | <b>ESTAÇÃO REMOTA .....</b>                      | <b>10</b> |
| 3.1      | Grandezas .....                                  | 11        |
| 3.2      | As Regiões do Estado .....                       | 12        |
| <b>4</b> | <b>O SISTEMA ATUAL .....</b>                     | <b>13</b> |
| <b>5</b> | <b>OS MEIOS DE COMUNICAÇÃO .....</b>             | <b>16</b> |
| 5.1      | A TransPac .....                                 | 17        |
| 5.2      | A Linha Discada .....                            | 19        |
| 5.3      | O Supervisor, o Concentrador e o Gerente .....   | 20        |
| <b>6</b> | <b>AS INTERFACES (A NÍVEL DE HARDWARE) .....</b> | <b>22</b> |
| 6.1      | Entre a Estação e o Supervisor .....             | 22        |
| 6.2      | Entre o Supervisor e o Concentrador .....        | 23        |
| 6.3      | Entre o Concentrador e o Gerente .....           | 23        |

|           |   |           |
|-----------|---|-----------|
| <b>7</b>  | <b>Os MÓDULOS .....</b>                     | <b>25</b> |
| 7.1       | O Supervisor .....                          | 25        |
| 7.1.1     | A Estrutura Lógica do Supervisor .....      | 27        |
| 7.2       | O Concentrador .....                        | 28        |
| 7.2.1     | A Estrutura Lógica do Concentrador .....    | 29        |
| 7.2.2     | A Varredura .....                           | 31        |
| 7.3       | O Gerente .....                             | 33        |
| 7.3.1     | A Estrutura Lógica do Gerente .....         | 34        |
| 7.3.2     | Um Pseudo-paralelismo .....                 | 36        |
| 7.3.3     | Interface Entre o Gerente e o Usuário ..... | 37        |
| <br>      |   |           |
| <b>8</b>  | <b>COMUNICAÇÃO .....</b>                    | <b>39</b> |
| 8.1       | Os Parâmetros de Comunicação .....          | 39        |
| 8.1.1     | A Bioscom .....                             | 39        |
| 8.2       | O Protocolo de Comunicação .....            | 40        |
| 8.2.1     | O Controle de Erro .....                    | 41        |
| <br>      |   |           |
| <b>9</b>  | <b>CONCLUSÃO .....</b>                      | <b>43</b> |
| <br>      |   |           |
| <b>10</b> | <b>BIBLIOGRAFIA .....</b>                   | <b>45</b> |

## 1 INTRODUÇÃO

A TELESC, empresa subsidiária do grupo TELEBRÁS, responsável pela telecomunicações no estado de Santa Catarina, entra na era da informática e se lança definitivamente na automação de seus serviços. Isto porque, nos últimos cinco anos tem investido seriamente em pesquisa e vem mantendo um contínuo intercâmbio com empresas similares nos Estados Unidos e Canadá, na busca de novos conhecimentos. É neste contexto que se enquadra a realização deste projeto: A busca de novos caminhos.

Durante os meses de Agosto à Dezembro deste ano, tentou-se desenvolver um novo sistema para supervisão das estações remotas da TELESC que apresentasse na sua estrutura idéias inovadoras, não apenas a nível de *hardware*, mas principalmete de *software*.

Desta forma, pode-se dizer que a meta deste trabalho é tentar descobrir novas técnicas que num futuro próximo possam ser efetivamente utilizadas no desenvolvimento de um ambiente para supervisão de estações remotas. Ainda dentro do objetivo, deseja-se sair do campo exclusivamente teórico desenvolvendo um protótipo onde essas novas idéias possam ser realmente aplicadas.

No decorrer desta monografia, tentar-se-á descrever o desenvolvimento de um ambiente para supervisão de estações remotas que busca satisfazer aos requisitos supra citados. Como também poderá ser observado, nenhuma teoria, copiada de livros, foi descrita, buscando-se desta forma a clareza e simplicidade nas explicações.



## ***DIVISÃO DAS TAREFAS***

O projeto de desenvolvimento de um sistema para supervisão de estações remotas teve que ser dividido em duas partes: a aquisição dos dados e o ambiente computacional. Isto porque num período de apenas quatro meses seria praticamente impossível atingir os objetivos desejados.

Desta forma, na primeira parte, estabeleceu-se que seria desenvolvido uma *Placa de Aquisição de Dados*, junto com seu respectivo programa de utilização (Driver). Esta placa deverá ser usada para a aquisição dos valores das grandezas a serem supervisionadas na Estação Remota.

Na outra parte, o Ambiente para Supervisão de Estações Remotas, um conjunto de programas (módulos) computacionais, deveria ser desenvolvido, com a função de trazer a estação localizada remotamente, até o usuário do sistema (técnico da empresa), utilizando-se para tal, da placa de aquisição, de um protocolo de comunicação próprio, modems, linhas telefônicas, microcomputadores, acesso a rede de pacotes, etc. O desenvolvimento desta parte foi atribuído ao estudante do Curso de Engenharia de Controle e Automação Industrial, Guilherme Ernani Vieira.

Apenas a título de complemento, o estudante José Paulo de Lucca Ramos, também do Curso de Engenharia de Controle e Automação Industrial, foi quem ficou responsável pela criação da placa de aquisição de dados, junto com seu respectivo programa de utilização (na monografia escrita por ele sobre seu projeto de fim de curso, pode-se encontrar todos os detalhes do desenvolvimento desta parte do sistema).

### 3 **ESTAÇÃO REMOTA**

Como o projeto diz respeito ao desenvolvimento de um ambiente para supervisão de estações remotas, é importante então deixar bem claro o que vem a ser uma *estação remota*.

Uma maneira simples (talvez até simples demais) de se definir o que é uma estação pode ser: *uma estação é um lugar onde estão agrupados os equipamentos responsáveis pela realização da telecomunicação numa determinada região*.

A TELESC possui uma sede central (pode-se dizer: estação central ou matriz), localizada no bairro de Itacorubi (Florianópolis) e qualquer estação além desta, é chamada de estação remota, pelo fato de se encontrar distante desta.

Os diversos equipamentos responsáveis pela telecomunicação em uma região são enquadrados em quatro grandes áreas dentro da empresa:

- (a) **Transmissão** (Ex: rádio UHF, multiplex, equalizador de cabo coaxial, etc);
- (b) **Comutação** (Ex: juntores, máquina de sinais, modulador, demodulador, etc);
- (c) **Rede** (Ex: pressurizador, receptor de proteção, demodulador, etc);
- (d) **Infraestrutura** (Ex: grupo gerador, retificadores, no-breaks, etc).

É importante dizer que cada estação deve ser climaticamente controlada, pois alguns dos seus dispositivos só operam de maneira eficiente e segura em determinadas faixas de temperatura ambiente e umidade do ar.

### 3.1 *Grandezas*

Neste projeto concentrar-se-á somente na área de infraestrutura, que trata das grandezas (ou variáveis) relacionadas à energia (ou alimentação) da estação.

Assim, as grandezas a serem supervisionadas em cada estação serão: as tensões nos retificadores e no inversor, tensão de saída do grupo gerador (GG), tensão nas baterias de arranque e nível de combustível no GG, tensão e corrente nos *no-break's*, densidade do líquido no interior das baterias, temperatura ambiente e umidade do ar.

Observação: em geral estações de pequeno porte não possuem GG's, já outras possuem tanto GG como *no-breaks*, mas todas devem possuir *no-breaks*.

A seguir, tem-se um esquema ilustrativo de uma estação remota dentro do contexto deste projeto.

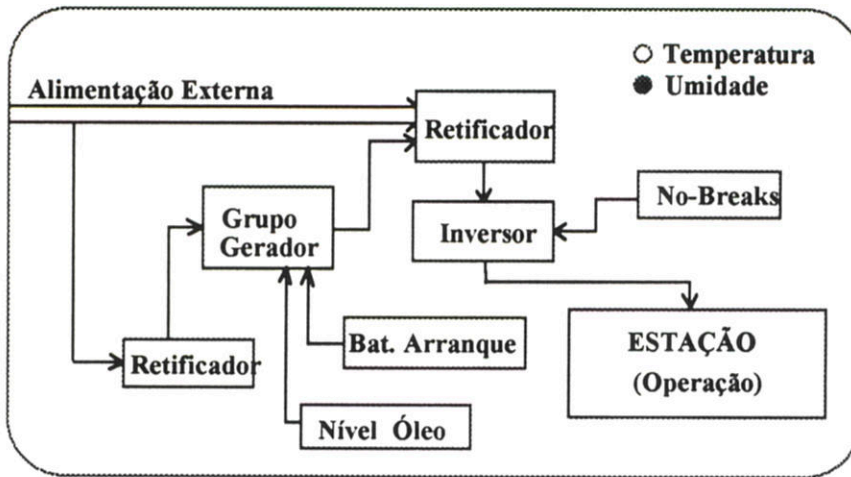


Figura 1: Esquema com as grandezas a serem supervisionadas em uma Estação Remota

### 3.2 As Regiões do Estado

Antes de se descrever o atual sistema de supervisão é importante deixar entendido o seguinte: para a TELESC, o estado catarinense é visto subdividido em seis grandes regiões: a região de Blumenau, de Chapecó, de Criciúma, de Florianópolis, de Joinville e de Lages. Na figura a seguir tem-se uma ilustração desta divisão:

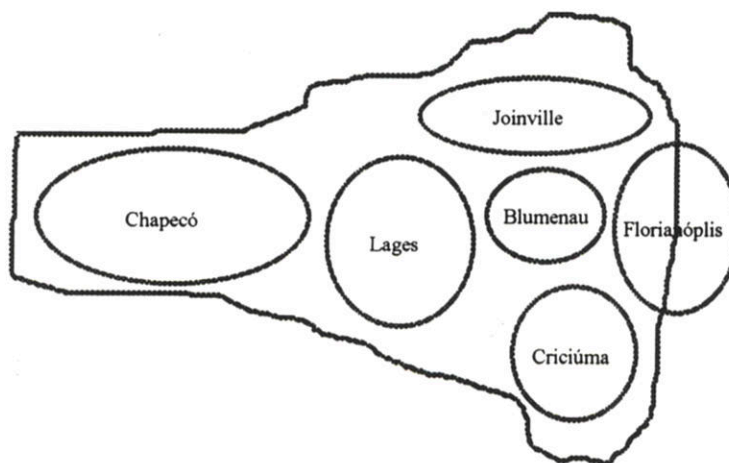


Figura 2: As grandes regiões do Estado Catarinense onde estão espalhadas as estações remotas da TELESC

#### 4 O SISTEMA ATUAL

Ao contrário do que se pode estar pensando, a TELESC já possui um ambiente para supervisão de estações remotas (ER), chamado de *Telesupervisão via PC (TSP)*.

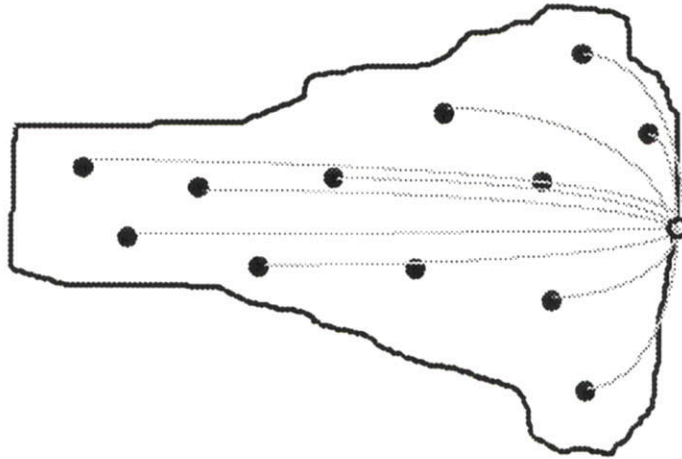
Este opera da seguinte maneira: sempre que ocorre algum problema ou uma pane em uma dada estação, um equipamento chamado TSP-20 capta essa ocorrência e gera um sinal de alarme (com prioridades diferentes). Para interligar todos os TSP-20's foi desenvolvido uma linha de comunicação. Um programa de computador faz então uma varredura contínua, através dessa rede, a todos esses TSP-20's, coletando e armazenando todos os alarmes vindos de todas as estações e mostrando-os na tela do computador no momento da sua chegada. Cada alarme aparece na tela do computador com uma cor correspondente à sua prioridade. Em determinados intervalos de tempo, o funcionário responsável vai até o computador e tira uma listagem dos alarmes mais urgentes e manda técnicos aos locais de onde eles vieram para consertar o defeito ocorrido.

Como pode ser observado, existem algumas desvantagens no atual sistema:

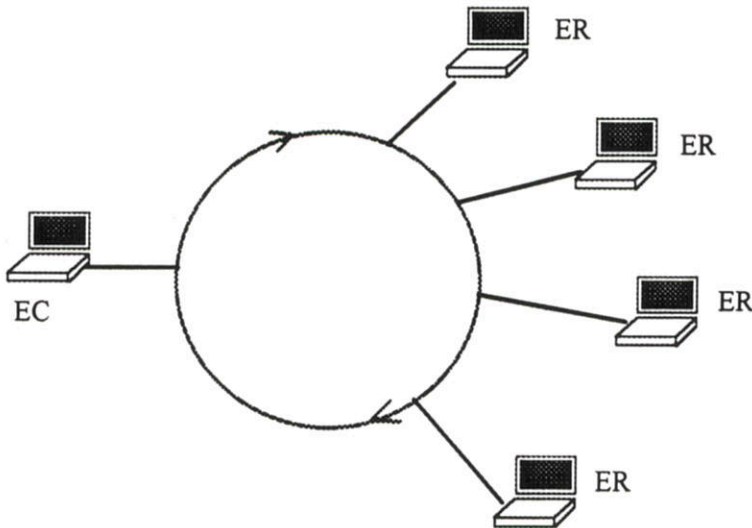
(a) não existe a possibilidade de se fazer manutenção preventiva nas estações. Como se viu, a geração de alarmes, por definição, só se dá a partir da ocorrência de um dado problema e o máximo que se pode fazer nesses casos é a manutenção corretiva;

(b) existe uma linha de comunicação dedicada exclusivamente a este sistema (figura 4), que centraliza todas as estações em um único ponto: na estação central (figura 3).

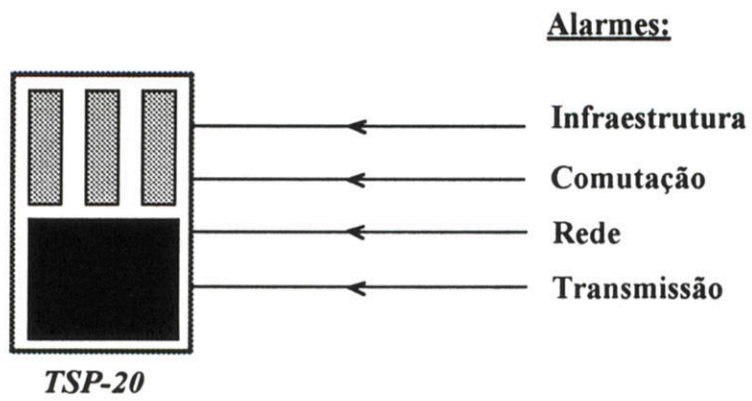
(c) a utilização de TSP-20's, equipamentos ultrapassados, caros e de pouca utilidade se comparados com os atuais micros PC AT, por exemplo (figura 5).



**Figura 3:** Todas as ER estão conectadas unicamente à estação central:



**Figura 4:** Uma rede exclusiva para interligar as ER's



**Figura 5:** Unidade remota concentradora de Alarmes:

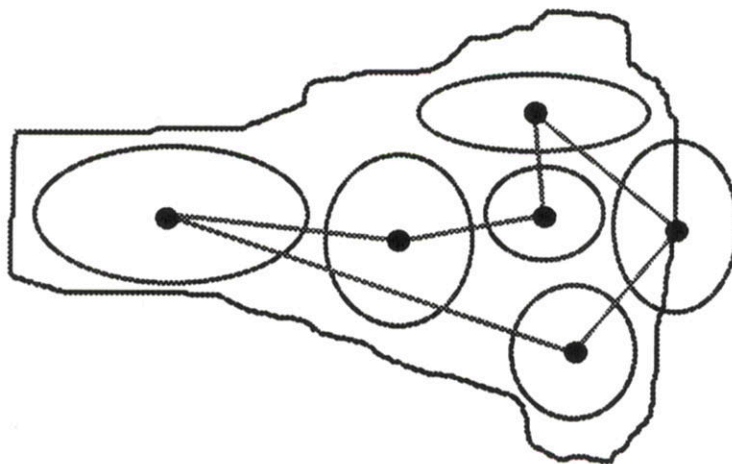
## 5 OS MEIOS DE COMUNICAÇÃO

Deseja-se que o próximo sistema de supervisão de estações remotas utilize a rede de pacotes (TransPac) desenvolvida pela própria empresa. Desta maneira, duas seriam as vantagens obtidas:

1º) Não haveria mais necessidade de se manter os gastos com uma rede exclusiva, como acontece atualmente;

2º) A TransPac é uma rede própria para a troca de dados com essas características, também é mais segura e mais confiável.

Entretanto, até o momento, a rede de pacotes não alcança todas as estações remotas do estado, apenas as grandes cidades (centros) das seis regiões (Blumenau, Chapecó, Criciúma, Florianópolis, Joinville e Lages), conforme ilustra a figura 6.

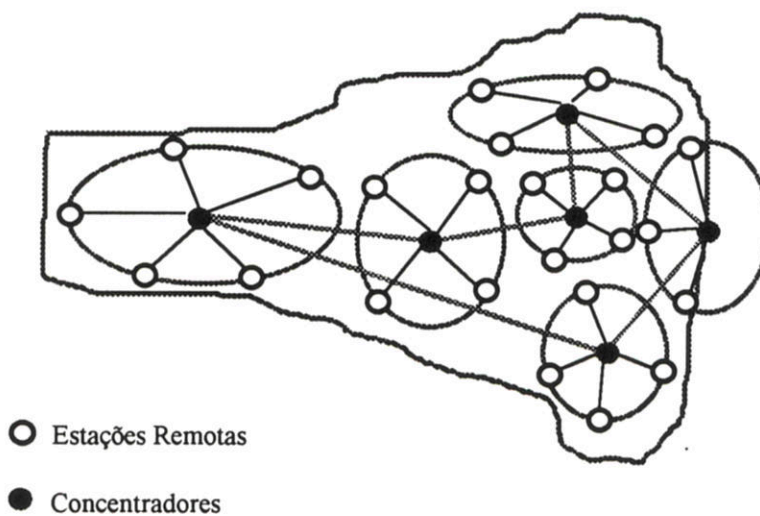


**Figura 6:** A interligação dos grandes centros via TransPac



O fato da TransPac não atingir a todas as ER's e o desejo de se desenvolver um novo sistema de supervisão descentralizado, deixou claro que o próximo sistema deverá possuir "centros" concentradores.

Uma das funções desses **concentradores** seria interligar as estações remotas de uma grandes região aos seus respectivos centros, utilizando-se para isto da Linha Discada. Veja figura 7:



**Figura 7:** Com a utilização conjunta da TransPac e da Linha Discada, o novo sistema poderá alcançar todas as estações remotas do estado.

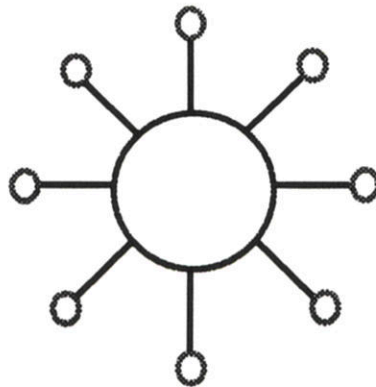
Neste capítulo tenta-se dar uma visão sobre a rede de pacotes e a linha discada.

## 5.1 *A TransPac*

A rede de pacotes - Transpac - é uma rede de comunicação feita exclusivamente para a troca de dados. Foi desenvolvida para atender a demanda de

mercado das empresas que necessitam de um meio de comunicação mais eficiente e mais seguro. Pode-se citar por exemplo, os bancos. Eles necessitam de um meio altamente seguro, para realizar suas transações; numa linha de telefonia comum, a confiabilidade é mais baixa e se está sujeito a perturbações e interferências constantemente, devido a sua própria *natureza* e ao grande número de usuários que a utilizam.

A TransPac foi implementada através da topologia de rede de difusão do tipo anel (figura 8). Nela, uma mensagem, subdividida em pacotes, circula o anel a procura do endereço a que ela foi enviada.



**Figura 8:** Topologia de rede de difusão por anel

Para que um usuário se conecte à rede de pacotes, basta que ele tenha uma linha telefônica conectada a central da TransPac. O usuário também pode escolher entre modo de comunicação síncrono e assíncrono. Se for usar o modo síncrono, deverá utilizar o protocolo de comunicação X.25, caso contrário, deverá desenvolver seu próprio protocolo. É importante lembrar que o protocolo X.25 é tido como sendo um protocolo de comunicação padrão, e a tendência é que mais tarde, as trocas de informações sejam todas implementadas por meio deste protocolo.

Apenas para esclarecer, na comunicação assíncrona não existe sincronismo entre o emissor e o receptor. Este deve estar apto a receber um determinado sinal a qualquer momento do emissor.

Neste projeto definiu-se que o modo de comunicação adotado seria o assíncrono, pois as transmissões a serem efetuadas são demasiadamente simples, sendo desta forma mais fácil definir um protocolo próprio do que utilizar o X.25.

## **5.2            *A Linha Discada***

A linha discada nada mais é do que a utilização da linha telefônica comum como meio de comunicação entre dois pontos (neste caso entre dois computadores).

Para utilizar este meio, basta que as duas pontas que querem se comunicar tenham acesso à rede de telefonia. Liga-se os micros à linha, através de modems, faz-se as devidas inicializações e pronto, o que for enviado de um lado chegará no outro lado, e vice-versa, de forma assíncrona.

É importante lembrar que em qualquer comunicação micro-a-micro, por linha telefônica, deve se ter modems conectando esses micros a esta linha. Isto acontece porque o sinal que percorre esse tipo de linha (tanto para Linha Discada, quanto para Rede de Pacotes) é analógico, enquanto que para os computadores os sinais devem ser unicamente digitais. É o modem é o dispositivo que faz a conversão Analógico-Digital-Analógico.

### 5.3 *O Supervisor, o Concentrador e o Gerente*

Com a estrutura acima sugerida, através dos meios de comunicação estabelecidos, decidiu-se que o novo sistema de supervisão deverá ser formado por três módulos básicos: O *Supervisor*, o *Concentrador* e o *Gerente*.

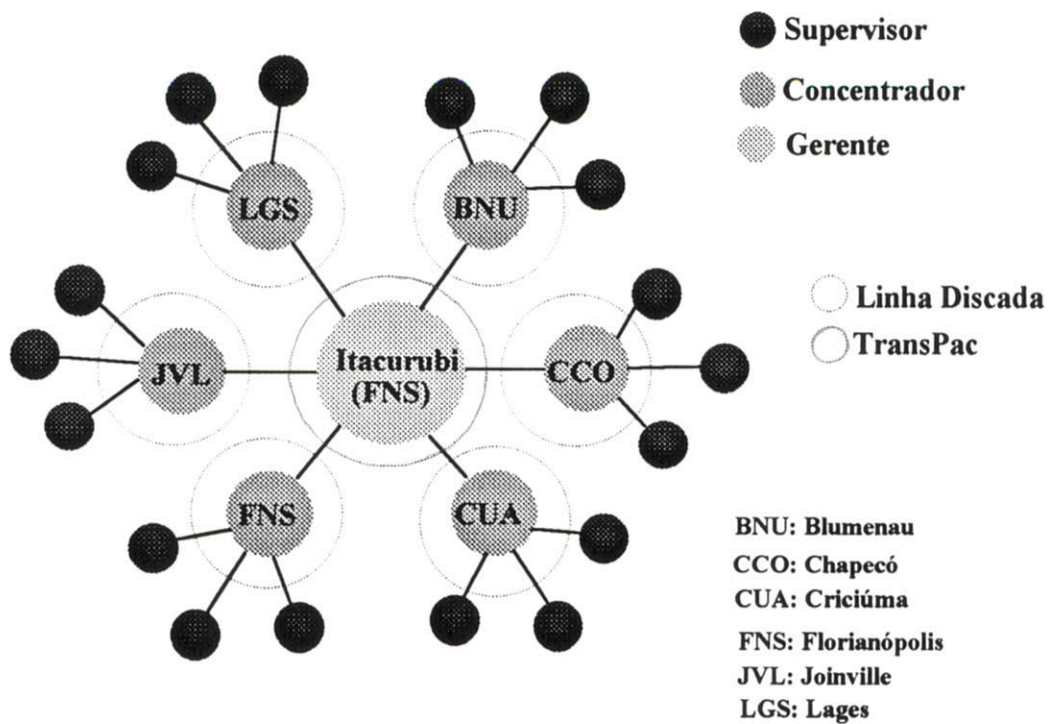
Esses três módulos são programas de computador que ficarão sendo executados constantemente em uma estação:

↳ *Supervisor*: a ficar sendo executado na estação remota (não a estação considerada como centro);

↳ *Concentrador*: a ser executado na estação remota concentradora (centro). Deverá integrar os supervisores da sua região;

↳ *Gerente*: a ficar sendo executado na estação central (Itacurubi - Florianópolis). Deverá integrar todos os concentradores do estado.

A figura a seguir tenta dar uma visão sobre a distribuição dos módulos do Ambiente para Supervisão de Estações Remotas (ASER) proposto:



**Figura 9:** A distribuição dos módulos do ASER

## *AS INTERFACES (A NÍVEL DE HARDWARE)*

O objetivo do presente capítulo é ilustrar através de alguns desenhos, como é feita a interface entre a estação remota e o Supervisor (item 6.1), entre o Supervisor e o Concentrador (item 6.2) e entre o Concentrador e o Gerente (item 6.3).

### *6.1 Entre a Estação e o Supervisor*

A comunicação do Supervisor com a estação se dá através de um arquivo texto criado pelo programa que faz a leitura das grandezas através da placa de aquisição de dados (Cap. 2). Veja figura 10:



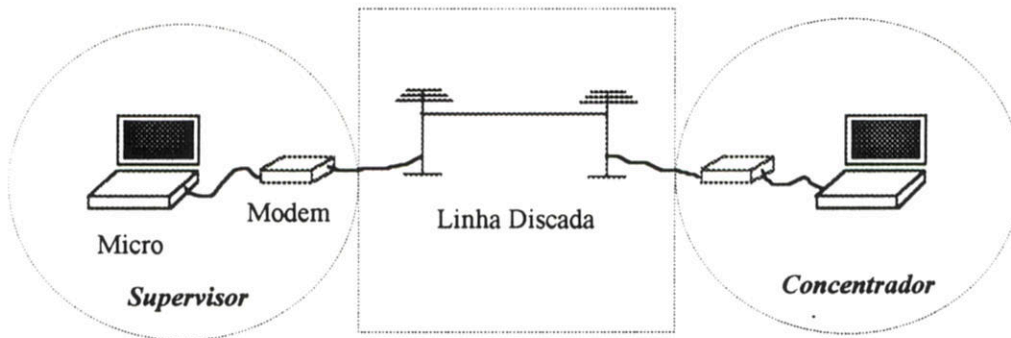
**Figura 10:** Interface entre a Estação e o Supervisor

Neste projeto, a placa de aquisição de dados pode fazer a leitura de apenas oito variáveis de uma estação (com esta simplificação, pôde-se dar seqüência ao desenvolvimento do projeto um pouco mais facilmente).

## 6.2 *Entre o Supervisor e o Concentrador*

Como se sabe, a interface entre o Supervisor e o Concentrador é feita pela comunicação via linha discada (item 5.2).

Como se viu, a comunicação via linha discada exige que os micros estejam conectados a modems. O desenho a seguir tenta ilustrar esta interface:



**Figura 11:** Meio para a comunicação entre o Supervisor e o Concentrador.

## 6.3 *Entre o Concentrador e o Gerente*

Como mencionado no capítulo anterior, a conexão entre o Gerente e os Concentradores é feita via rede de pacotes, conforme tenta ilustrar a figura seguinte:

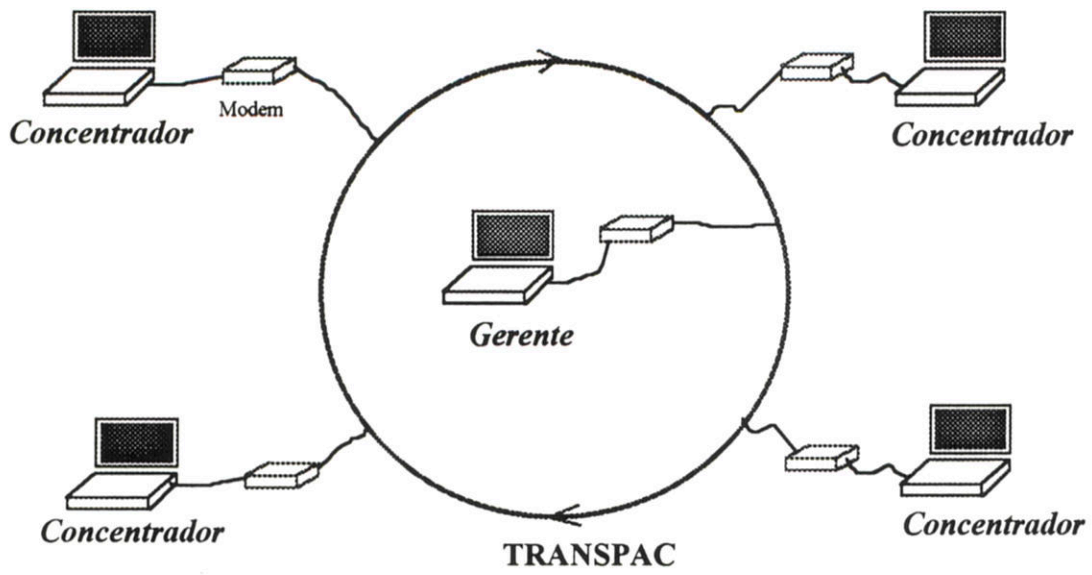


Figura 12: Interface dos Concentradores com o Gerente



## *OS MÓDULOS*

Neste capítulo, serão descritos com mais detalhes as funções e as estruturas de cada um dos módulos anteriormente mencionados.

Como se deseja realizar a supervisão de estações remotas, fica claro a necessidade de se ter em cada uma delas alguma ferramenta que realize, constantemente, a aquisição dos valores das grandezas supervisionadas. Por esta razão, teve que ser implementado um módulo chamado *Supervisor*.

A um nível acima do Supervisor, tem-se o *Concentrador*, que é responsável pela concentração de alguns Supervisores e pela interligação deles com o Gerente.

Para integrar todo o sistema, foi desenvolvido um terceiro módulo chamado *Gerente*. Ele é quem permite que o usuário na central possa verificar como estão operando as estações remotas do estado.

É necessário dizer que todos esses programas foram implementados em linguagem C++, e que infelizmente, por falta de tempo, o módulo Gerente não pode ser concluído.

### *7.1 O Supervisor*

O *Supervisor* é um programa que deverá ficar sendo executado permanentemente em um micro na estação remota. A princípio a idéia de se ter um

microcomputador em cada estação poderia sugerir que os gastos seriam demasiados e por isso talvez não valesse a pena se proceguir neste caminho. Entretanto, um simples PC XT (já quase sendo jogado fora) serve perfeitamente para estes propósitos. Além disso, os TSP-20 deverão brevemente ser substituídos por microcomputadores. Não havia por tanto razões para que esta idéia não fosse aceita.

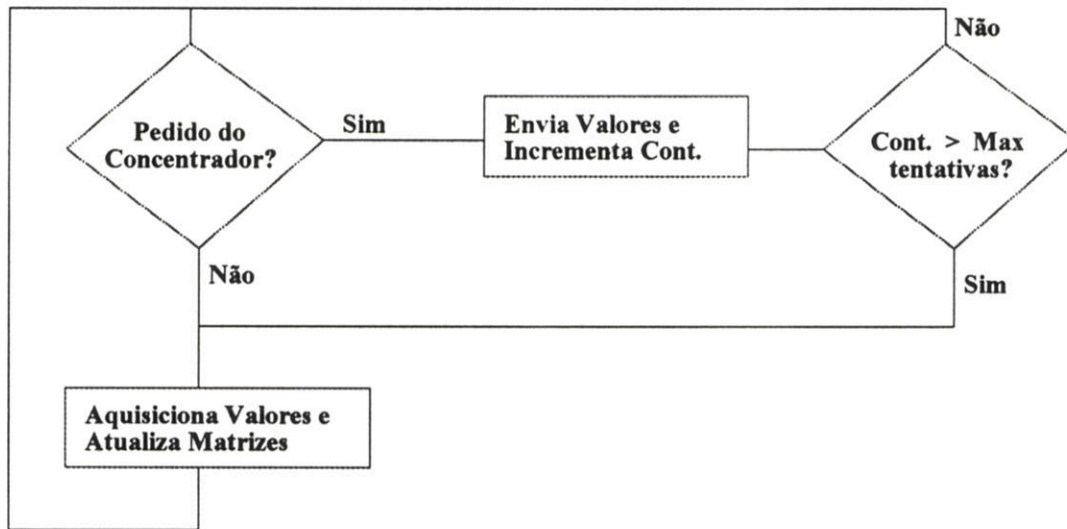
O Supervisor tem duas funções básicas: fazer a leitura dos valores das grandezas que estão sendo supervisionadas e transmiti-los aos concentradores.

O procedimento para a realização dessas funções está simplificadaamente descrito a seguir.

1. o Supervisor chama um programa que faz a comunicação com a placa de aquisição de dados;
2. o programa de aquisição, quando chamado, utiliza-se da placa e faz a leitura de todos os valores das grandezas conectadas à placa. Após a leitura, ele gera uma tabela (na forma de caracteres ASCII) com todos os valores lidos, na forma de um arquivo texto;
3. após a criação do arquivo, o Supervisor faz a leitura deste e atualiza sua própria tabela dos valores das grandezas.
4. logo após a atualização da tabela, o Supervisor verifica se chegou algum sinal do Concentrador; caso nenhum sinal tenha chego, ele reinicia este procedimento, no contrário, chama uma rotina para comunicação com o Concentrador para lhe enviar os valores das grandezas supervisionadas (o protocolo de comunicação utilizado entre o Supervisor e Concentrador será discutido mais a seguir).

### 7.1.1 A Estrutura Lógica do Supervisor

A estrutura lógica usada na implementação do supervisor pode ser facilmente compreendida a partir do diagrama abaixo:



**Figura 13:** A estrutura lógica do Supervisor

Onde:

**Pedido do Concentrador?** ⇔ Verifica se o Concentrador está pedindo que lhe mande os valores das grandezas sendo supervisionadas;

**Envia valores e Incrementa Cont.** ⇔ O Supervisor manda para o Concentrador os valores das grandezas e incrementa um contador;

**Cont. > Max tentativas?** ⇔ Se o contador atingiu um valor máximo, é considerado que houve algum problema na comunicação e por algum motivo desconhecido ela não está sendo estabelecida;

***Aquisiciona Valores e Atualiza Matrizes*** ⇨ O Supervisor chama a rotina para a aquisição dos valores das grandezas (através da placa) e após ter lido esses valores, atualiza sua matriz. Desta forma, nenhuma história é guardada, apenas os últimos valores das grandezas.

## 7.2 ***O Concentrador***

O Concentrador é o módulo que interliga vários supervisores em uma das seis grandes regiões do estado catarinense. É ele que envia, via TransPac, os valores das grandezas de todas as estações da região. Ele também pode ser visto como um intermediador entre o Supervisor e o Gerente.

A presença do Concentrador no sistema de supervisão é uma das maiores diferenças entre o "Telesupervisão via PC" e este que está sendo proposto. Como se viu, no TSP não existe a figura do Concentrador, a central se comunica diretamente com todas as estações, de forma centralizada e quando aconteça alguma falha no micro onde o sistema está rodando (estação central), toda a supervisão fica interrompida. Não existe nenhum ponto de emergência ou de escape.

Neste projeto, os concentradores também têm esta função. Caso alguma pane ocorra na central, eles continuam operando normalmente, supervisionando as estações da região. Isto porque o Concentrador possibilita ao usuário funções semelhantes as do Gerente, como por exemplo uma listagem com os atuais alarmes das estações.

### 7.2.1 A Estrutura Lógica do Concentrador

Da mesma forma que para o Supervisor, a estrutura lógica usada na implementação do Concentrador pode ser facilmente compreendida a partir do diagrama abaixo:

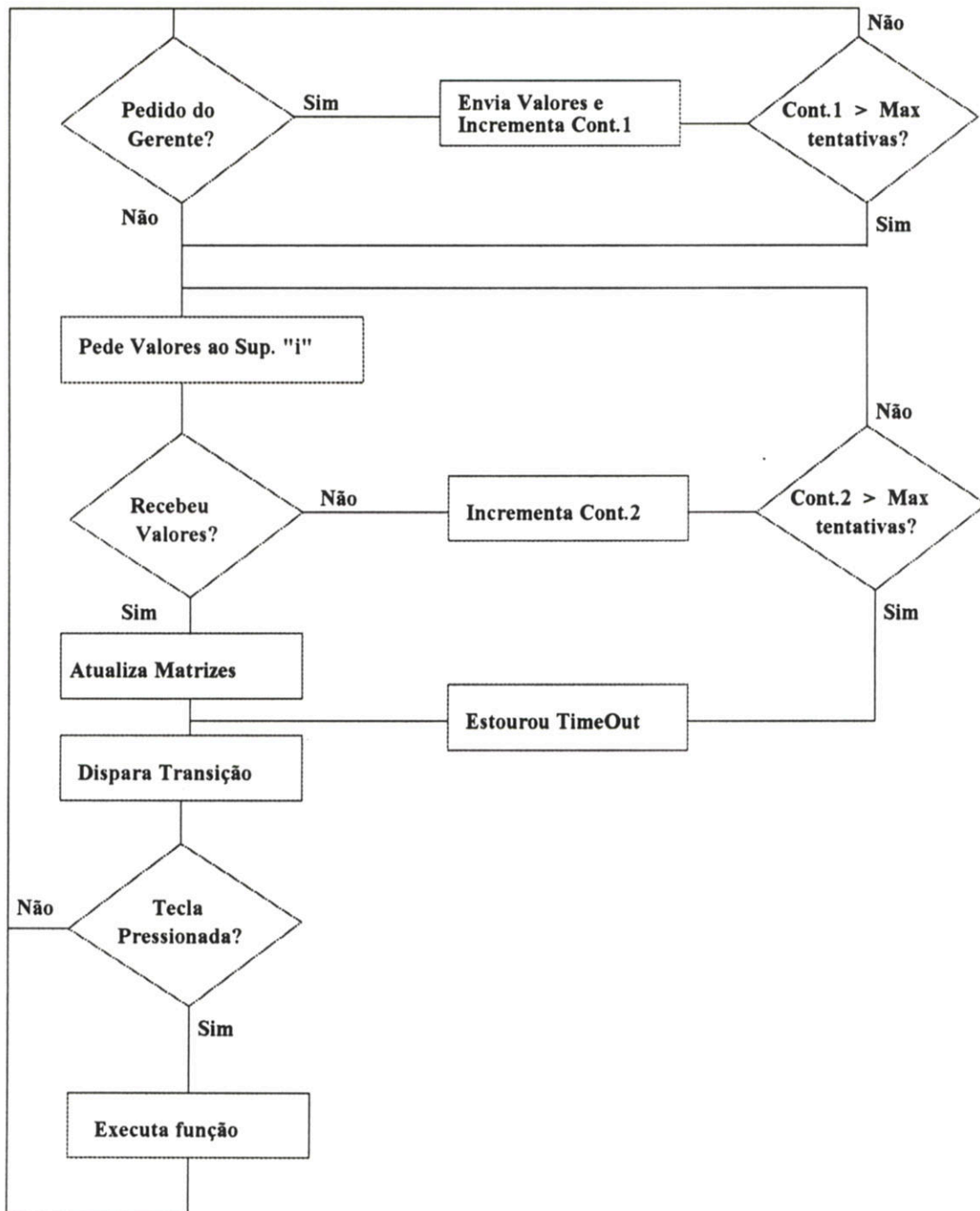


Figura 14: A estrutura lógica do Concentrador

Onde:

***Pedido do Gerente?***  $\Leftrightarrow$  Verifica se o Gerente está pedindo que lhe mande os valores das grandezas das ER's;

***Envia valores e Incrementa Cont.1***  $\Leftrightarrow$  O Concentrador manda para o Gerente os valores das grandezas e incrementa um contador 1;

***Cont.1 > Max tentativas?***  $\Leftrightarrow$  Se o contador 1 atingiu um valor máximo, é considerado que houve algum problema na comunicação e por algum motivo desconhecido ela não está sendo estabelecida;

***Pede Valores ao Sup. "i"***  $\Leftrightarrow$  O Concentrador envia um sinal ao Supervisor "i" para que este lhe mande os valores das grandezas da estação. Como será visto no item seguinte, o gerenciamento da comunicação é modelado por uma rede de Petri (item 7.2.2), onde o deslocamento da ficha através dos lugares indica qual é o supervisor a ser interrogado;

***Recebeu Valores?***  $\Leftrightarrow$  Verifica se os valores das grandezas da estação foram recebidos;

***Cont.2 > Max tentativas?***  $\Leftrightarrow$  Se o contador 2 atingiu um valor máximo, é considerado que houve algum problema na comunicação com o Supervisor "i", desta forma estoura um *time-out* e um alarme anunciando que a estação em questão não responde deve ser aberto;

***Dispara Transição***  $\Leftrightarrow$  Uma transição da rede é disparada, selecionando o próximo Supervisor a ser interrogado no ciclo seguinte;

**Tecla Pressionada?** ⇔ Para que o Concentrador também possa atender um possível usuário, o programa fica constantemente lendo o teclado, e executando uma determinada função caso o usuário tenha apertado alguma tecla (selecione uma opção no menu). Desta forma, quando o ciclo é executado rapidamente, dá a idéia de um pseudo-paralelismo (item 7.3.2).

### 7.2.2 *A Varredura*

Talvez o nome varredura não comunique o verdadeiro sentido do que se deseja expressar nessa função. O que se quer na realidade é implementar uma maneira de se gerenciar a comunicação de um módulo com vários outros módulos. Esta comunicação poderá se dar por exemplo entre o Concentrador e seus supervisores ou entre o Gerente e os Concentradores.

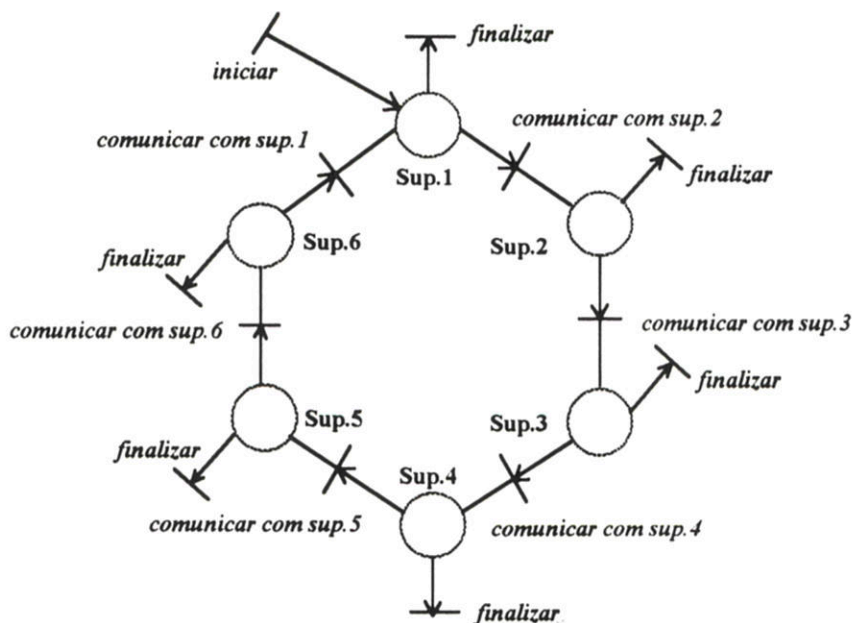
Durante a varredura, o Concentrador interroga um supervisor, depois um outro e assim por diante, até chegar no último; depois, reinicia o ciclo, voltando a interrogar o primeiro.

Facilmente poderia ter sido implementado uma função específica para a esta função, entretanto, no desejo de descobrir técnicas mais modernas, que possam ser facilmente *expandíveis* no futuro, optou-se por modelá-la por *Redes de Petri (RdP)*.

O motivo que levou à utilização de redes de Petri, à utilização de uma outra ferramenta, como por exemplo autômatos, foi o fato de se ter em mãos alguns programas feitos especialmente para se trabalhar com RdP's (\*), que se adequam perfeitamente a função aqui desejada. Esses programas, por terem sido feitos em

Programação Orientada a Objetos, permitiram que a função de gerenciamento da comunicação (varredura) pudesse ser modelada por esta filosofia.

Talvez a melhor maneira de se entender como ficou a estrutura que realiza o gerenciamento da comunicação com os módulos, seja através de um desenho, por isso a figura 15 mostra um desenho (simplificado) de uma RdP que realiza o gerenciamento da comunicação de um Concentrador com seis supervisores:



**Figura 15:** Esquema de uma RdP responsável pelo gerenciamento da comunicação do Concentrador com seis Supervisores.

(\*) Durante seu estágio no Laboratório de Controle e Microinformática - LCMI - o estudante Guilherme Ernani Vieira desenvolveu, sob a orientação do mestrando Francisco de Assis Soares e da Professora Janete Cardoso, um conjunto de programas em linguagem C++, com Orientação por Objetos, que permite a outros usuários utilizar Redes de Petri na modelagem de sistemas a eventos discretos.



Como dito anteriormente, a função de varredura não existe somente nos concentradores, o Gerente também realiza essa mesma função, só que em vez de se comunicar com os supervisores, se comunica com os concentradores. Pelo fato da idéia ser exatamente a mesma, este assunto não será tratado no próximo tópico.

Também é importante salientar o seguinte: quem já teve a oportunidade de trabalhar com redes de Petri pôde perceber que ela é uma ferramenta realmente muito potente na modelagem de sistemas a eventos discretos; entretanto, no contexto deste projeto, dentro do ambiente para supervisão de estações remotas, ela está sendo muito subutilizada. Uma função de poucas linhas faria este mesmo gerenciamento, num tempo ainda menor. Mas o objetivo é exatamente este, encontrar outras técnicas que mais tarde possam ser potencialmente aplicadas no desenvolvimento de um sistema de supervisão na Telecomunicações de Santa Catarina.

### **7.3 O Gerente**

Interligar e supervisionar as estações remotas, mostrar em tempo real os alarmes que acabaram de chegar, gerar relatórios, mostrar os valores das grandezas em uma dada estação, essas são algumas das funções do Gerente.

Durante a implementação deste módulo, uma grande dificuldade encontrada foi o fato de que o Gerente precisa atender ao usuário do sistema, durante o tempo que este desejar, sem contudo, parar de se comunicar com os concentradores. Desta forma, houve a necessidade de se desenvolver um programa que trabalhasse com um pseudo-paralelismo, atendendo a várias tarefas *quase que simultaneamente*.

Não foi mencionado, mas o Concentrador também teve que ser implementado com esta idéia. Isto porque ele deve ficar constantemente se comunicando com as estações, mas também tem que se comunicar com o Gerente quando este desejar; e mais, quando um usuário quiser entrar no módulo de concentração, para verificar a listagem dos alarmes por exemplo, ele também deverá atendê-lo.

### **7.3.1        *A Estrutura Lógica do Gerente***

Da mesma forma que para o Concentrador, a estrutura lógica usada na implementação do Gerente pode ser melhor compreendida a partir do seu diagrama, conforme o desenho abaixo:

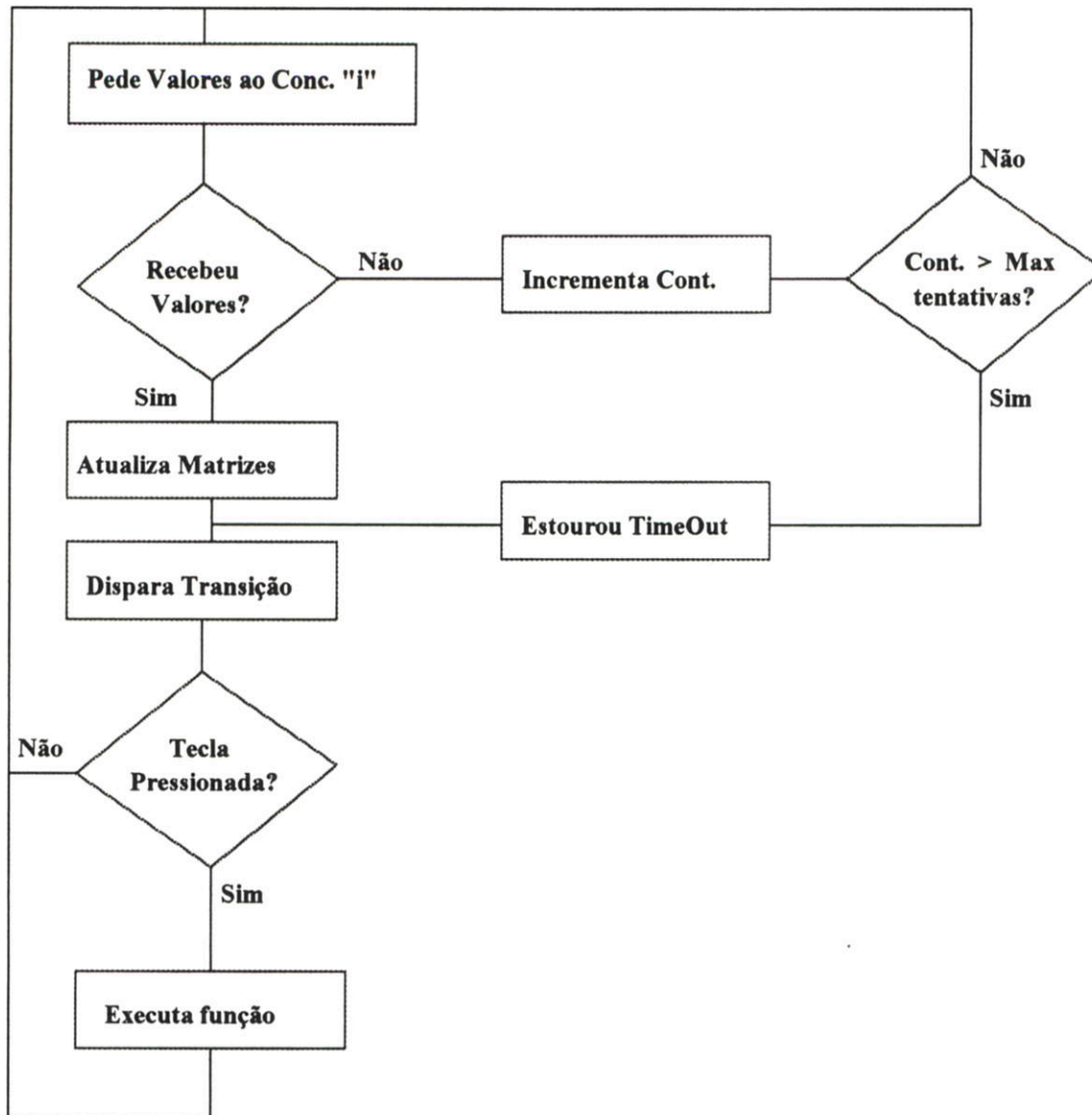


Figura 16: A estrutura lógica do Gerente

Onde:

*Pede Valores ao Conc. "i"*  $\Rightarrow$  O Gerente envia um sinal ao Concentrador "i" para que este lhe mande os valores das grandezas das estações por ele concentradas. O gerenciamento da comunicação é modelado por uma rede de Petri exatamente da mesma forma que no Concentrador;

**Recebeu Valores?** ⇨ Verifica se os valores das grandezas das estações foram recebidos;

**Cont. > Max tentativas?** ⇨ Se o contador atingiu um valor máximo, é considerado que houve algum problema na comunicação com o Concentrador "i", desta forma estoura um *time-out* e um alarme anunciando que o Concentrador em questão não responde deve ser aberto;

**Dispara Transição** ⇨ Como no caso do Concentrador, uma transição da rede é disparada, selecionando o próximo Concentrador a ser interrogado;

**Tecla Pressionada?** ⇨ Para que o Gerente também possa atender um possível usuário.

### 7.3.2 *Um Pseudo-paralelismo*

Não se pode ter um "verdadeiro" paralelismo quando os micros PC's utilizados só possuem um processador. No intuito de se alcançar este paralelismo, o que se faz então é executar um pouco de cada tarefa, sob uma certa velocidade, dando a impressão de que as tarefas estão acontecendo ao mesmo tempo. Este "quase" paralelismo é aqui tratado como sendo um pseudo-paralelismo.

No caso do Concentrador, o pseudo-paralelismo deve acontecer porque deseja-se que ele "converse" com os supervisores e com o Gerente e que também possa atender um possível usuário. Já o Gerente deve atender o usuário do sistema e também deve ficar interrogando os concentradores. O algoritmo usado para implementar o pseudo-paralelismo foi o seguinte:

O programa fica constantemente executando sua função principal (que no caso do Concentrador é se comunicar com os supervisores e no caso do Gerente é se comunicar com os concentradores). Quando ele percebe que alguma tecla foi pressionada, pára a função principal, executa uma breve função correspondente a tecla apertada e retorna a sua função principal. Este *loop*, quando feito de maneira razoavelmente rápida, dá idéia de que o programa atende a várias tarefas simultaneamente.

### 7.3.3 *Interface entre o Gerente e o Usuário*

Num futuro bem próximo, a figura do usuário do sistema de supervisão deverá ser extinta (não a do técnico que vai até a estação para consertá-la, mas sim o usuário que fica em frente ao micro). Hoje em dia ainda não consegue acionar automaticamente, em caso de emergência, um técnico para ir ao local com problemas. Ainda existe a necessidade de haver alguém que faça essa função. Os alarmes chegam, mas uma pessoa é que tem que despachar os técnicos.

Na TELESC, os técnicos são despachados aos locais necessários a partir da abertura de um bilhete de atividade (BA). A idéia é que, num passo posterior, o Gerente faça automaticamente a abertura desses BA's, diminuindo a necessidade do usuário do sistema.

O programa onde o Gerente está rodando oferta ao usuário as seguintes opções:

(a) *ver alarmes* ⇒ Uma nova tela mostra, sob a forma de uma tabela, todos os alarmes armazenados, cada qual com uma cor correspondente a sua prioridade;

(b) *ver estação* ⇒ O programa pergunta ao usuário qual estação ele quer ver. Logo que o usuário selecionou uma estação, o programa mostra um tabela com todos os valores das grandezas supervisionadas daquela estação.

(c) *relatório* ⇒ Quando o usuário escolhe esta opção, é gerado um arquivo texto como uma listagem de todos os alarmes de todas as estações.

Também fica aparecendo na tela principal do Gerente, nas linhas inferiores da tela, o nome da região onde está o Concentrador que naquele momento conversa com o Gerente.

Não foi mencionado, mas esta mesma interface entre o Gerente e o usuário também existe no Concentrador.

## 8 *A COMUNICAÇÃO*

O objetivo do presente capítulo é descrever como foi implementado a comunicação entre os módulos do *Ambiente para Supervisão de Estações Remotas*.

### 8.1 *Os Parâmetros de Comunicação*

A ligação do modem com o computador se dá por comunicação serial assíncrona (RS232-C), através das portas COM1 ou COM2. Neste projeto, essas portas foram inicializadas com as seguintes atribuições:

- ① *velocidade*: 9600 bps (bits por segundo) ; .
- ② *paridade*: par;
- ③ *stop bits*: 1 bit;
- ④ *bits de dados*: 8 bits;

Obs: O modem deve ser inicializado com esses mesmos parâmetros.

#### 8.1.1 *A Bioscom*

A função utilizada para interligar o micro com um periférico, através das portas seriais é chamada, na linguagem C++, de *Bioscom*.

A Bioscom permite que o programador utilize as portas seriais a um nível de programação mais "alto", sem se preocupar com as atribuições de "baixo nível", como por exemplo, as inicializações das portas nos circuitos integrados.

Como não foi possível utilizar a Linha Discada e nem a TransPac na implementação do ambiente, os micros foram ligados diretamente, através das suas respectivas portas seriais, por fios de telefonia comuns.

## 8.2 *O Protocolo de Comunicação*

O protocolo desenvolvido para fazer a comunicação entre os módulos (supervisor-concentrador e concentrador-gerente) procurou utilizar a filosofia de comunicação por pacotes.

Cada pacote é formado pelo conjunto de 18 bytes, conforme pode ser observado na figura 17:

|       |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |      |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| start | bd1 | bc1 | bd2 | bc2 | bd3 | bc3 | bd4 | bc4 | bd5 | bc5 | bd6 | bc6 | bd7 | bc7 | bd8 | bc8 | stop |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|

*start: start byte*  
*bd: byte de dados*  
*bc: byte de controle*  
*stop: stop byte*

**Figura 17:** O pacote para a comunicação

No pacote, cada byte de dado corresponde ao valor de uma grandeza sendo supervisionada. Para isto, existe uma relação (que não é necessariamente a função de normalização) entre o valor real da grandeza e o que vai no pacote. Esta relação é feita de maneira que o valor real (com uma casa de precisão) possa ser transmitido em



um byte (0..255). Por exemplo, se o valor de tensão de uma bateria  $V$  varia de 0,0 a 50,0 volts, pode-se considerar que o valor a ser transmitido  $V'$  seja igual a:  $V' = (V \times 10) / 2$ ; assim  $V'$  estará entre 0 e 250, podendo ser perfeitamente transmitido em um byte.

### 8.2.1 *O Controle de Erro*

A estratégia utilizada para o controle de erro foi a seguinte:

→ o valor do stop byte em cada pacote a ser enviado (stop) corresponde ao resultado de um *Ou-Exclusivo* (^) entre todos os bytes de dados:

$$\text{stop} = \text{bd1} \wedge \text{bd2} \wedge \text{bd3} \wedge \text{bd4} \wedge \text{bd5} \wedge \text{bd6} \wedge \text{bd7} \wedge \text{bd8};$$

→ o programa que recebeu o pacote também faz um *Ou-Exclusivo* entre todos os bytes de dados recebidos (bdr):

$$\text{stop}' = \text{bdr1} \wedge \text{bdr2} \wedge \text{bdr3} \wedge \text{bdr4} \wedge \text{bdr5} \wedge \text{bdr6} \wedge \text{bdr7} \wedge \text{bdr8};$$

→ se os valores dos stop's forem iguais ( $\text{stop}' = \text{stop}$ ), considera-se que não houve problema algum na comunicação, caso contrário é solicitado que este pacote seja novamente enviado.

Esta rotina fica sendo executada até que todos os dados sejam transmitidos ou seja atingido um número máximo de tentativas. Neste caso, é considerado que o Supervisor ou Concentrador em questão *não responde* (estourou o *time-out*) e um alarme para esta categoria é aberto.

É importante deixar claro o seguinte: quando o Supervisor recebe um sinal do Concentrador pedindo para que lhe mande os valores das grandezas daquela estação, com apenas um pacote ele consegue mandar todos os dados. Entretanto, quando a comunicação é entre o Gerente e o Concentrador, este tem que mandar ao Gerente tantos pacotes quanto forem o número de Supervisores por ele concentrados.

## CONCLUSÃO

Pode-se concluir a descrição do Desenvolvimento de um Ambiente para Supervisão de Estações Remotas afirmando que os objetivos desejados foram atingidos .

O ambiente para supervisão de estações remotas desenvolvido possui características como:

- permite a supervisão de todas as estações remotas do estado de Santa Catarina;
- foi subdividido em três módulos: o **Supervisor**, responsável pela aquisição de dados; o **Concentrador**, que tem como função integrar os supervisores em uma dada região, transmitindo os valores das grandezas ao Gerente e podendo atender um usuário em caso de necessidade; e o **Gerente**, que pode ser considerado como o cérebro do ambiente, a partir dele todas as estações remotas podem ser supervisionadas;
- permite a utilização da rede de pacotes - TransPac;
- gerenciamento da comunicação modelado por redes de Petri;
- linguagem C++, com Programação Orientada a Objetos; e
- a existência de um pseudo-paralelismo, que permite que o programa atenda ao usuário sem parar de supervisionar as estações.

Infelizmente o ambiente não pode ser colocado em campo (teste junto a estação) pois não havia acesso disponível ao cabo que faz a conexão do Tecnópolis à rede de pacotes, na estação central.

Por este motivo, as conexões micro-a-micro foram feitas através de pares de fios '*drops*' (fios de telefone comuns), o que na realidade muito pouco ou quase nada alteraram as características obtidas.

A falta de tempo também foi um aspecto que dificultou um pouco o desenvolvimento do projeto. Um sistema deste tipo exige um período maior do que apenas quatro meses, sem esquecer que neste último mês, teve-se que escrever todo projeto, o que consumiu um tempo bastante considerado.

Mas o objetivo central foi atingido. Pôde-se desenvolver um protótipo onde quase todas as idéias acima mencionadas puderam ser realmente aplicadas.

Assim, comunicação por pacotes, gerenciamento por redes de Petri, programação Orientada a Objetos são sem dúvida filosofias que poderão muito em breve fazer parte da implementação de um verdadeiro Ambiente para Supervisão de Estações Remotas, conforme pode ser observado.

SCHILDT, Herbert. **"C Completo e Total"**. Tradução Marcos Ricardo Alcântara Morais. São Paulo: Makron, McGraw-Hill, 1990.

BERRY, John Thomas. **"Programando em C++"**. Tradução Mário Moro Fecchio. São Paulo: Makron, McGraw-Hill, 1991.

ASSUMPCÃO de Milton Mira. **Gerenciamento de Redes: Uma Abordagem de Sistemas Abertos**. Makron Books do Brasil. 1993.

DERFLER, Frank J. **Guia de Conectividade** / Frank J. Derfler Jr.; tradução Insight Serviços de Informática. -Rio de Janeiro: Campus. 1993.

DERFLER, Frank J. **Guia para a Interligação de Redes Locais** / Frank J. Derfler Jr.; tradução Fernando B. Ximenes. Rio de Janeiro: Campus. 1993.

CNT<sub>r</sub>. **Apostila: Básico de Gerência Integrada de Rede**. 2º Edição: Maio/1993.

**APOSTILA: Sistemas Distribuídos e Rede de Computadores**. Marcelo Stemmer e Vitorio Mazzola. UFSC. 1993.