

IMPLEMENTAÇÃO DE UM PROTOCOLO  
PARA SINCRONIZAÇÃO DE RELÓGIO

Miguel Antonio Sovierzoski,  
José Luiz Andriolli Pereira

fax. (041) 224-5170  
e.mail "CEFETPR @BRFAPPEFPBITNET"

CEFET/PR - CPGII  
Rua Sete de Setembro, 3165, Centro  
80230 Curitiba - PR

Sumário

Este artigo trata do problema de sincronização dos relógios das entidades de um sistema distribuído em ambiente industrial. É apresentada uma solução, baseada na alteração de um serviço da rede, evitando-se o overhead causado por serviços específicos de sincronismo de relógio.

Abstract

This paper covers the problem of synchronization of clocks of entities belonging to a distributed system in an industrial environment. A solution is presented, based on an existing network service, therefore, avoiding the introduction of overhead that would be caused by self contained services, dedicated to clock synchronism.

## Implementação de um Protocolo para Sincronização de Relógio

---

### 1 Introdução

---

Neste artigo, abordaremos os problemas da sincronização de relógios em um sistema distribuído. Este SD é composto de equipamentos auxiliares de testes da linha de produção de centrais públicas, na fábrica da EQUITEL em Curitiba. O sistema apresentado constitui-se de diversos equipamentos, com um grande fluxo de dados, os quais devem ser enviados para um servidor remoto. É extremamente importante uma sincronização precisa dos relógios.

Analizamos um caso prático onde necessitamos de um mecanismo de sincronização. A partir daí, desenvolvemos algumas idéias sobre a resolução deste problema. Escolhemos a solução mais adequada ao contexto e descrevemos a implementação do mesmo.

### 2 Descrição do Sistema Distribuído de teste

---

Na linha de produção de centrais públicas, após testes IN-CIRCUIT nas placas e testes preliminares a nível de sistema, as centrais (denominadas de Equipamentos sob Teste - EST) entram em um processo de BURN-IN dinâmico. Permanecem alguns dias sob variação controlada de temperatura e sendo monitoradas através de programas de auto-teste. O processo de BURN-IN visa um envelhecimento precoce, eliminando os problemas de mortalidade infantil que aparecem no início da vida dos equipamentos [4].

Foi desenvolvido um sistema de equipamentos auxiliares de teste (EAT), os quais concentram as mensagens geradas pelos programas de auto-teste dos EST. Devido ao volume de equipamentos e ao tráfego de informações, tornou-se necessário elaborar um sistema distribuído de equipamentos auxiliares de teste, com um eficiente método de sincronização de relógio para indicar o horário em que as mensagens ocorrem.

A seguir, mostramos toda a estrutura do sistema distribuído.

O EST envia, através de uma interface paralela padrão Centronics, as mensagens dos programas de auto-teste, para um dispositivo impressor. Entre estas informações destacam-se : o início de um ciclo de teste, a posição da falha, o fim de um ciclo de teste, o número de ciclos de teste executados, etc.

Se houver alguma falha, os técnicos podem determinar através dos relatórios, qual foi a placa do EST que apresentou defeitos. Esta placa defeituosa é então consertada, retornando ao teste de BURN-IN.

Um EST pode ter até quatro saídas paralelas padrão Centronics, inviabilizando a utilização de dispositivos impressores devido a quantidade necessária.

A estratégia adotada consistiu em desenvolver uma estrutura de concentração dos dados, a qual coletasse os relatórios de testes emitidos pelos ESTs, e gerasse novos relatórios contendo informações adicionais, necessários para o gerenciamento do sistema. O EAT tem por finalidade justamente concentrar os relatórios de testes de diversos ESTs.

### 3 Características Desejadas do Sistema

---

Na fase de especificação do sistema, levantamos uma série de serviços que deveriam ser implementados para facilitar a sua utilização, bem como, ir de encontro com as políticas de automação de fábrica.

Este sistema permitiria filtrar os dados por canais do EST, podendo gerar relatórios de falhas de cada equipamento com os respectivos momentos de ocorrência. Característica bem interessante para os técnicos que trabalham nesta área.

Outra vantagem seria a possibilidade de efetuar a consulta dos relatórios, diretamente no computador, procurando eliminar os relatórios de testes em papel.

Uma terceira característica que está sendo estudada, é a possibilidade de um tratamento estatístico dos dados, analisando qual o tempo mínimo que o equipamento necessita permanecer em BURN-IN, para sair da área de mortalidade

infantil dos componentes. Com base nestes dados, pode-se alterar a duração do teste em BURN-IN, acarretando em um custo de produção menor para o equipamento.

Porém, para atender a algumas características, seria necessário manter os EATs com o mesmo horário, sendo necessário um forte serviço de sincronização de relógios.

Na seqüência desenvolvemos algumas idéias para a implementação do sistema de sincronização de relógio.

#### 4 Análise para Implementação do Sistema

Com as premissas do sistema, procuramos desenvolver uma estrutura de equipamentos e uma topologia de sistema de forma a atender as necessidades levantadas.

Seguem os estudos efetuados e as suas decorrências.

A idéia inicial para o novo sistema foi desenvolver Equipamentos Auxiliares de Teste (EAT), que possuem canais de entrada para as saídas paralelas padrão Centronics dos ESTs, e uma interface para uma rede de comunicação fabril.

Desta forma, quando um EAT estiver com todos os canais de entrada ocupados, simplesmente instalamos mais um EAT e conectamos este a rede. Com isto, podemos ampliar o número de canais a medida que se faça necessário.

Os EATs devem possuir buffers de recepção das mensagens que chegam através dos canais, e agregar a esta, qual o canal que recebeu a mensagem, bem como o horário de ocorrência. Isto se faz necessário para o sistema de gerenciamento determinar a origem da mensagem, ou seja, qual EST enviou a mensagem. O horário de ocorrência serve como informação auxiliar aos operadores do sistema.

Na medida do possível os EATs enviam, via rede, as mensagens armazenadas, para o computador gerenciador do sistema.

Na figura 1 vemos a estrutura do sistema distribuído de teste.

Como é necessário identificar o momento em que os relatórios de testes são gerados, propusemos a colocação de dispositivos de relógio de tempo real nos EATs.

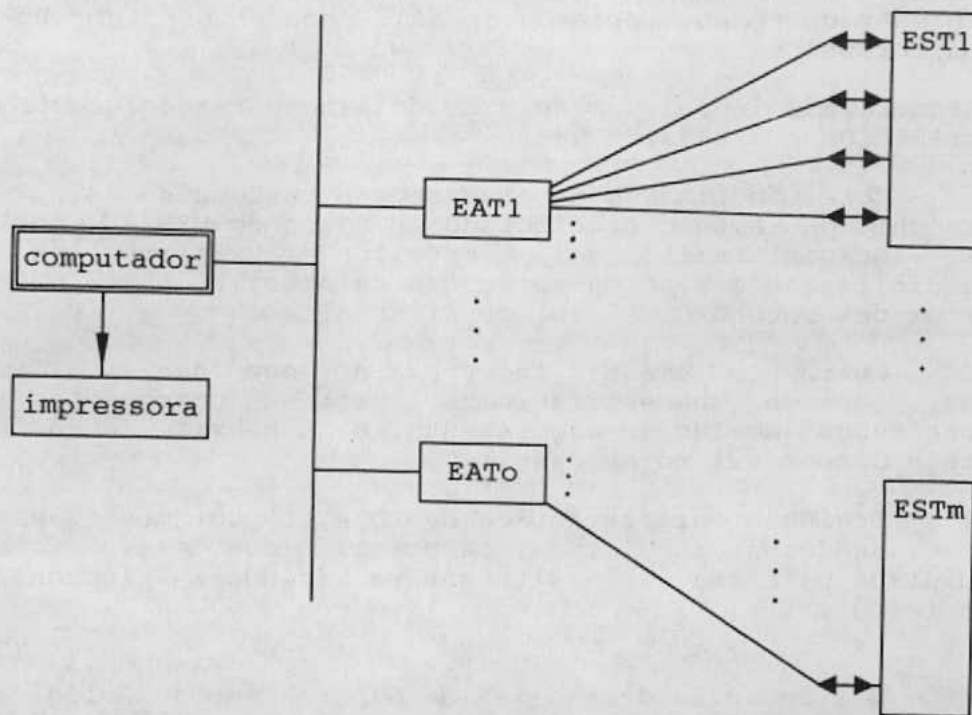


Figura 1 - Estrutura do sistema distribuído de teste.

Caso esta idéia viesse a ser adotada, traria alguns problemas, entre eles:

- 1) O circuito integrado de relógio utilizado (MK48T02 - Mostek), possui uma bateria solidária, a qual tem duração finita, não podendo ser recarregada nem substituída.

Nos EATs, com o passar do tempo, seria necessário trocar estes componentes, provocando uma paralisação temporária de parte do sistema.

- 2) Alguma disfunção de hardware ou software no EAT, poderia travar o funcionamento do relógio.

Estes problemas seriam resolvidos com um serviço de rede que executasse a inicialização do relógio. Porém, este serviço deveria ser executado de forma transparente e sem intervenção de operador, pois o sistema funciona ininterruptamente, e os operadores permanecem somente em horário comercial.

- 3) O relógio em questão, apresenta um erro menor que 35 ppm, isto corresponde a um erro de +- 1,5 minuto/mês. Com o passar do tempo, teríamos os EATs cada um com um horário diferente.

Poderíamos incluir um serviço de sincronização/acerto dos relógios dos EATs.

Concluindo: além de elevarmos o custo dos EATs, pois este chip de relógio está cotado em torno de US\$ 30, teríamos que incluir mais dois serviços auxiliares na rede (inicialização e sincronização dos relógios), para podermos ter um desempenho confiável do sistema.

Outro problema é a incerteza de que daqui a alguns anos, quando necessitassemos destes componentes para substituição devido ao esgotamento da bateria, este ainda esteja disponível no mercado.

Devido a característica da rede, de ser mestre-escravo e procurando dar ao sistema, características de simplicidade, evoluímos para uma outra alternativa visando a sincronização dos relógios.

#### 4.1 Evolução das Idéias de Sincronismo de Relógios

---

O maior problema é conseguir um sistema de relógio nos EATs, confiável e simples de desenvolver. Os trabalhos de Drummond [3] e Lamport [6], tratam justamente do problemas de SD que envolvem relógios, porém as soluções apresentadas são muito poderosas para o SD em questão.

Montando uma estrutura de relógios virtuais nos EATs, temos o problema de diferença de horários com a passagem do tempo, devido ao erro ser acumulativo, sendo necessário um serviço de sincronização de horário para todos os EATs. Este problema também é detalhadamente analisado nos trabalhos citados acima.

A coleta dos dados dos EATs, é feita ciclicamente (polling) através de um computador mestre dedicado para esta sistema. Isto é facilitado por uma rede proprietária, desenvolvida para atender a interligação de equipamentos na fábrica. A sincronização entre os diversos relógios virtuais, seria feita através da alteração de algum serviço desta rede.

A alteração neste serviço consiste em tornar todas as

mensagens de requisição de dados portadores do horário atual do computador. O EAT recebe esta requisição e o horário, e assume este horário como sendo o seu horário atual. Desta forma, cada questionamento a um EAT, executa automaticamente o serviço de sincronização do seu relógio.

E o erro acumulativo do relógio virtual, não é significativo, devido a velocidade com que é executado a varredura cíclica dos EATs.

O sistema deve funcionar ininterruptamente. Quando ocorre um retorno de energia, todo o sistema deve voltar a operação, sem a intervenção de um operador.

No retorno da energia, os EATs estão com o valor do relógio virtual desatualizados exatamente do intervalo de tempo de falta de energia. Porém, o relógio de tempo real existente no computador, está com o valor atual. A medida que o mestre vai executando a varredura cíclica nos EATs, estes vão atualizando os seus relógios virtuais.

Ao término do primeira ciclo de varredura, após o retorno da energia, todos os relógios virtuais estão atualizados com o valor atual.

Desta forma, conseguimos eliminar os problemas do relógio de tempo real dos EATs, sem aumentar o número de serviços da rede.

Na implementação, citamos maiores detalhes sobre os mecanismos de sincronização de relógio no sistema distribuído de teste.

## 5 Implementação

-----

Para implementarmos a comunicação entre os EATs e o computador, utilizamos uma rede de comunicação industrial, que vem sendo desenvolvida para as aplicações de automatização de fábrica.

### 5.1 A Organização dos Dados

-----

As mensagens de testes, recebidas pelos EATs, estão em código ASCII, e seu tamanho não ultrapassa 1 kbytes.

Quando as mensagens são transferidas para os EATs, via interface paralela padrão Centronics, este agrega a mensagem, a data e hora correntes, o número do canal que recebeu a mensagem e o número do EAT. Estas informações adicionais são necessárias para o operador identificar o horário de ocorrência, e para o software aplicativo no computador conseguir endereçar corretamente o destino desta mensagem.

1 octeto	1 octeto	6 octetos	variável
número.EAT	número.canal	data e hora	mensagem

Figura 2 - Estrutura dos dados

Podemos ver na figura 2, a estrutura dos dados que são armazenadas nos EATs e que posteriormente trafegam pela rede.

## 5.2 A Comunicação

-----

A rede de comunicação industrial baseia-se na proposta de sugerida pelo mini-MAP.

Ela ainda está em desenvolvimento, porém um bom número de primitivas já estão disponíveis, viabilizando a sua utilização imediata no sistema. Na seqüência apresentamos algumas das suas principais características.

### 5.2.1 A Camada Física

-----

Utilizamos o canal serial do microcontrolador 8051 da Intel, como canal de comunicação. A interface elétrica utilizada no barramento, foi o padrão de interface EIA RS-485, que permite até 32 nós ligados na linha. Um cabo par trançado sem blindagem é o meio de transmissão, no caso "half-duplex". Para o ambiente fabril onde foi instalado o sistema, não houve restrições quando a esta configuração.

### 5.2.2 A Camada de Enlace

-----

Até o momento, a rede adota o modelo de mestre único.



Um mecanismo de controle de acesso ao meio não é necessário, pois este é controlado pelo mestre da rede, que é único.

Porém como existe a previsão de evolução para o padrão IEEE802.4 a sub-camada de acesso ao meio (MAC) foi parcialmente implementada.

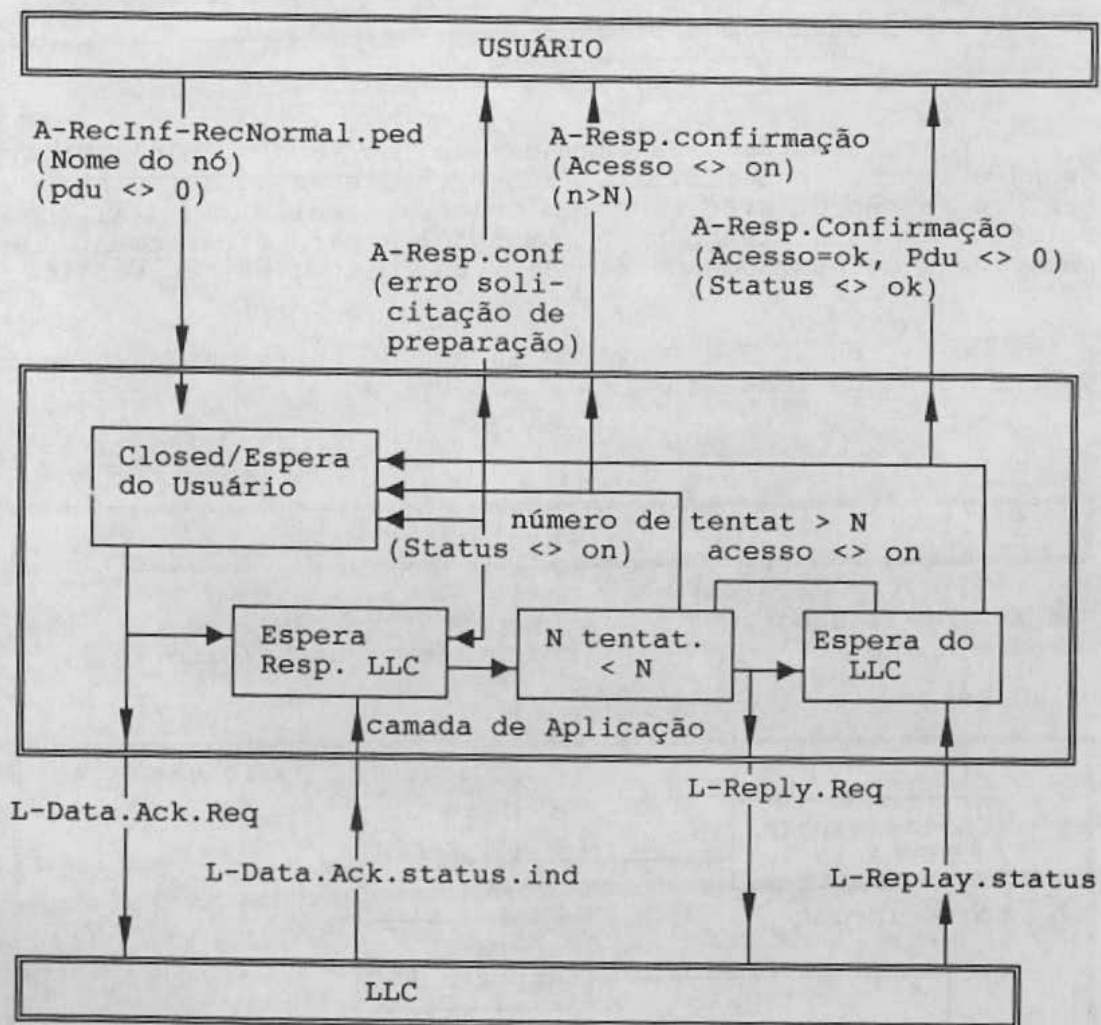


Figura 3 - Serviço de Leitura simples de um Nó.

O LLC tipo 3 foi escolhido para ser implementado dentro da camada de enlace, por vir de encontro com as necessidades de se ter uma máquina de controle do enlace lógico, que suportasse o esquema de varredura imposta por uma rede de mestre único e também por ter todos os serviços

confirmados imediatamente a nível do LLC. O resultado esperado é um sistema com capacidade de correção de alguns erros, e a detecção imediata de falha de um equipamento [1].

### 5.2.3 Camada de Aplicação

-----

A implementação da camada de aplicação não requer nenhuma norma específica. Isto é resultado de dois fatores básicos: a não existência de um consenso a nível mundial com relação a suas funções e protocolos para o barramento de campo, e a necessidade de atender a aplicações específicas [1].

Nas figuras 3 e 4, vemos as MEF dos serviços da camada de aplicação da rede.

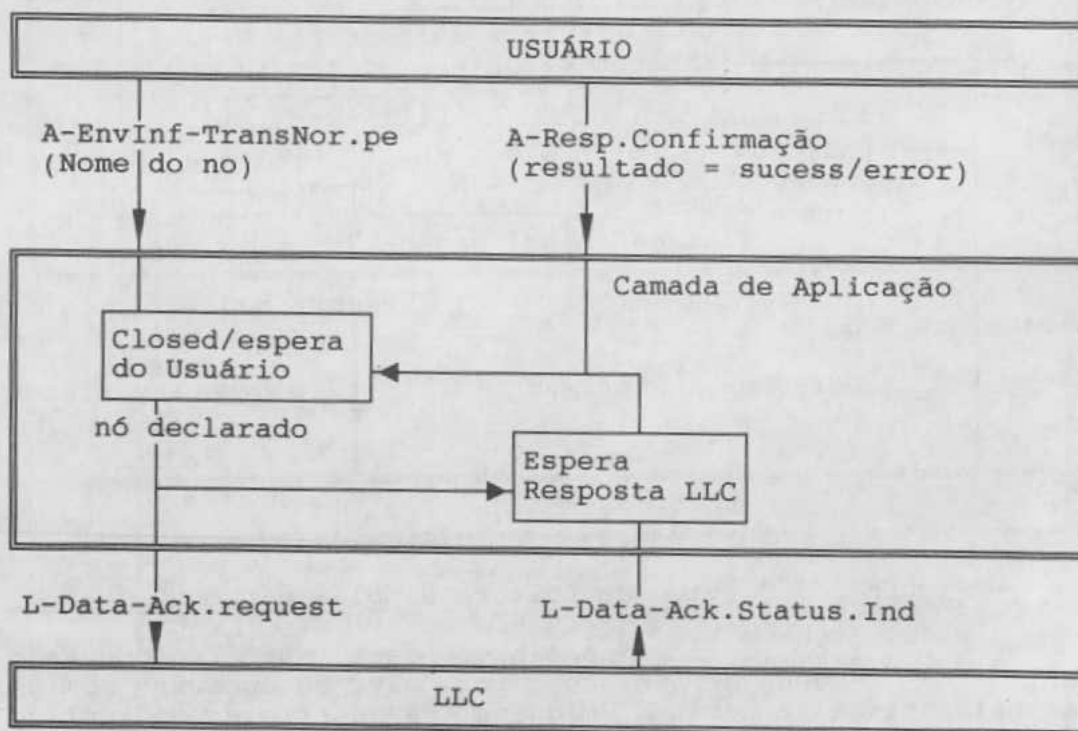


Figura 4 - Serviço de Envio de Informação para um Nó.

### 5.3 A Sincronização dos Relógios

---

É necessário garantir que os EATs possuam o mesmo horário. Todos os EATs serão sincronizados com o horário do computador mestre, que utilizará a sua base de tempo como sendo a hora correta. Esta base de tempo é obtida pela utilização de um hardware específico para a geração de hora e data, e que esta conectado num dos slots do computador mestre.

A coleta de dados é feita em dois passos pela camada de aplicação: No primeiro passo, é enviado um pedido de dados pelo mestre para um dos EATs, via primitiva A-RecInf-RecNormal.request. Este prepara os dados que possui disponíveis para transmissão e os envia para o mestre. O mestre deve, então, armazenar os dados em um lugar seguro (unidade de disco ou winchester). Em seguida é enviada uma ordem de eliminação dos dados transmitidos do buffer de armazenamento temporário do EAT. Esta eliminação é feita através da primitiva A-EnvInf-TransNormal.request.

Para efetuarmos o sincronismo entre os relógios, tínhamos duas alternativas:

A primeira alternativa seria, periodicamente ativar um serviço específico de acerto de relógios. Este sistema tem um problema intrínseco de aumentar a ocupação da rede, o que diminui a taxa real de transferência de dados dos EATs para o mestre, e, conseqüentemente, a quantidade de EATs que podem ser atendidos por um único concentrador.

A segunda alternativa, que nos pareceu ser a mais viável, é utilizar a própria primitiva A-RecInf-RecNormal.request, que efetua a requisição de dados, para transportar a hora correta. Este procedimento acrescenta apenas alguns bytes adicionais à mensagem, e o "overhead" adicional comparado com a solução anterior é mínimo. Semelhante a solução sugerida por Drummond [3], onde as mensagens de difusão recebem um timestamp para sincronização de relógios.

O horário recebido via A-RecInf-RecNormal.request é então comparado com a hora e a data do relógio virtual do EAT. Caso a diferença ultrapasse o valor permitido, o relógio virtual é inicializado com este horário recebido.

Nos EATs foi implementada uma máquina de relógio virtual para garantir que as mensagens de falha possuam o

horário de ocorrência correto, mesmo que a rede sofra alguma perturbação longa. Neste caso, contudo, não se pode garantir a hora correta dos EATs com a mesma precisão.

#### 5.4 Programas Aplicativos

-----

Devido a especificidade da implementação, foram desenvolvidos programas aplicativos para as entidades ligadas a rede, no caso um programa para o computador (mestre), e outro para os Equipamentos Auxiliar de Teste.

Nos EATs temos alguns processos sendo executados, dentre eles, destacamos: a varredura dos canais; verificando se algum EST deseja enviar mensagens; o gerenciamento de buffers de mensagens; o relógio virtual; além dos serviços de rede.

O computador executa a função de gerente da rede, efetuando a varredura cíclica dos EATs, o gerenciamento das mensagens recebidas, organizando-as em arquivos em memórias de massa, além de ser a interface com os operadores do sistema.

#### 6 Conclusões

-----

Verificamos que uma pequena adaptação em um serviço da rede, permitiu um eficiente processo de sincronização de relógios virtuais nas entidades de um sistema distribuído em um ambiente industrial.

Isto foi possível, dado que a camada de aplicação está fora de normas internacionais de redes.

Caso contrário, não teríamos a mesma flexibilidade para alterar os seus serviços, sendo necessário a implementação de processos melhor elaborados para sincronização de relógio. Estes processos acarretariam em um alto "overhead", com significativa queda de performance e aumento de custos de implantação e tempo de desenvolvimento.

O sistema encontra-se em fase de implantação, e estamos especificando o controle estatístico das falhas a ser incluído no computador mestre.

## Referências

- [1] Dergint, D. E. A.,  
Um Barramento de Campo para Automação de Garantia de  
Qualidade de Produção,  
Dissertação de Mestrado, CEFET/PR 1990.
- [2] DIN V 19245 - Profibus, January 88.
- [3] Drummond, R.,  
Impact of Communication Networks on Fault-Tolerant  
Distributed Computing,  
A Thesis for the Degree of Doctor of Philosophy,  
Cornell University, June 86.
- [4] Jensen, F.; Petersen, N. E.,  
BURN-IN: an engineering approach to the design and  
analysis of Burn-In procedures,  
Chichester, John Wiley & Sons, 1982, 166p.
- [5] IEEE Standard 802.4 - 1985, "Token passing bus access  
method and physical layer specification".
- [6] Lamport, L. and Melliar-Smith, P. M.,  
Synchronizing clocks in the presence of faults,  
Journal of the ACM 32, 1 (January 85), pp 52-78.
- [7] Pleinevaux, P. and Decotignie, J. D.,  
Time Critical Communication Networks : Field Buses,  
IEEE Network Vol.2, No.3 pp 55-63 May 88.