José Augusto de Almeida Pinheiro de Carvalho

Avaliação do Desempenho de Redes PROFIBUS-DP Suportada em Técnicas de Injecção de Faltas

Dissertação submetida para obtenção do grau de
Doutor em Engenharia Electrotécnica e de Computadores
pela Faculdade de Engenharia da Universidade do Porto

Trabalho realizado sob a orientação do Professor Doutor

Adriano da Silva Carvalho

Professor Associado com Agregação
da Faculdade de Engenharia da Universidade do Porto

e co-orientado pelo Professor Doutor

Paulo José Lopes Machado Portugal

Professor Auxiliar
da Faculdade de Engenharia da Universidade do Porto

Universidade do Porto
Faculdade de Engenharia
Departamento de Engenharia Electrotécnica e de Computadores
2006

Ao meu filho Pedro pelas brincadeiras adiadas.

Agradecimentos

Em primeiro lugar, desejo exprimir a minha amizade e o meu reconhecimento ao Professor Doutor Adriano da Silva Carvalho, meu professor e orientador nas várias etapas de formação superior, pela forma atenta e esclarecida como sempre orientou a minha actividade científica e pelo empenho que colocou na obtenção das condições que possibilitaram a preparação desta dissertação.

Ao Professor Doutor Paulo José Lopes Machado Portugal meu co-orientador pela amizade, o empenho e contributo para o aprofundamento e discussão de aspectos relevantes para a dissertação.

Aos corpos directivos da Escola Superior de Tecnologia e de Gestão do Instituto Politécnico de Bragança, pela condições criadas que possibilitaram a realização desta dissertação. Aos meus colegas que me apoiaram e incentivaram ao longo da dissertação.

Mais que todos à minha esposa, aos meus pais e meus sogros pelo apoio e paciência demonstrada.

Resumo

Limitações de natureza tecnológica condicionaram o desenvolvimento dos sistemas de controlo a uma abordagem que conduziu à centralização da sua estrutura. Com a evolução da tecnologia esta abordagem alterou-se e actualmente muitas das estruturas de controlo assentam sobre o paradigma da distribuição de tarefas. Neste contexto, as redes de comunicação desempenham um papel fundamental na estrutura e na operação do sistema de controlo. Nos níveis inferiores de controlo a comunicação é estabelecida através de redes de campo que fazem a interligação entre pequenos controladores, sensores e actuadores, num processo tipicamente caracterizado pela troca de pequenas mensagens muitas das vezes de natureza cíclica, e usualmente com restrições temporais críticas.

Dado o papel das redes de campo nos sistemas controlo, e fruto do aumento de complexidade, custos de desenvolvimento dos sistemas e sua exploração, e também do tipo de tarefas que desempenham, tornaram importante a consideração de aspectos relacionados com o efeito da operação da rede na disponibilidade do sistema, na segurança de pessoas, ou mesmo de impactos no ambiente em que se inserem. Assim, de uma forma genérica importa avaliar modos de operação das redes que potencialmente contribuam para uma diminuição da confiança no funcionamento dos sistemas que suportam. Neste contexto, a avaliação do funcionamento das redes de campo na presença de faltas, assume relevância na identificação das principais fontes de degradação de desempenho, que no limite possam levar o sistema a falhar a função para que foi especificado.

No domínio da avaliação e da validação de sistemas os métodos experimentais são ferramentas poderosas, que podem ser utilizadas para alterar as normais condições de operação de um sistema e avaliar como desempenha nestas condições A técnica de injecção de faltas inclui-se nesta classe de métodos e permite introduzir de forma intencional e controlada erros no sistema. Assim, constitui uma solução eficaz na avaliação de sistemas complexos, para os quais a aplicação de outras técnicas é difícil, ou por vezes falha.

Nesta dissertação pretende-se contribuir para a caracterização do funcionamento de uma das redes de campo mais difundidas em aplicações de automação industrial – o PROFIBUS-DP. A caracterização é centrada no seu funcionamento em modo degradado, designadamente quando a operação é perturbada por interferências de natureza electromagnética. Este padrão de perturbação é susceptível de conduzir a rede para modos de operação que têm potenciais impactos na disponibilidade da rede e na sua capacidade de garantir o cumprimento de restrições temporais em aplicação de tempo-real. A área da avaliação do funcionamento na presença de faltas é uma área onde existe pouco trabalho publicado relativo ao comportamento do PROFIBUS-DP, e o que existe é manifestamente insuficiente para fornecer uma caracterização sólida do seu desempenho nestas condições de operação.

Para suportar a avaliação da rede foi desenvolvido um ambiente de injecção constituído por uma infra-estrutura de injecção de faltas na qual se inclui uma rede PROFIBUS-DP. A infra-estrutura de injecção implementa a técnica de injecção física de faltas, capaz de injectar faltas num ambiente distribuído de acordo com um processo estocástico. O *hardware* do módulo de injecção foi desenvolvido de forma a apresentar uma elevada controlabilidade do processo de injecção, assim como, de uma elevada resolução das faltas injectadas, permitindo injectar com precisão, faltas de um só bit e mais que um bit no barramento de comunicação. Os nós de comunicação foram desenvolvidos de acordo com a norma do PROFIBUS-DP. A estrutura do software e do hardware dos nós de comunicação permitem o acesso à camada de ligação de dados, e assim obter informação relevante relativa aos estados de operação do protocolo.

A avaliação foi efectuada de forma a identificar eventos que afectem a organização das estações no anel lógico da rede PROFIBUS-DP, tendo sido identificados seis eventos com impactos importantes ao nível do desempenho da rede. Três destes relacionados com perdas de *token*, e os restantes associados a saídas não intencionais de estações do anel. A sua probabilidade foi inferida para diferentes cenários de faltas, assim como, foi avaliada a forma como os mecanismos de recuperação dos eventos identificados desempenham em condições de faltas. As experiências incidiram igualmente sobre a questão da resposta da rede nestas condições de operação. Neste contexto, quando a rede opera na configuração multi-mestre foram observados impactos significativos quer resposta temporal da rede, quer no tempo de recuperação dos mecanismos de tolerância a faltas. Este comportamento deixa antever uma má resposta da configuração multi-master em cenários de faltas.

Esta página foi intencionalmente deixada em branco

Abstract

The available technology has conditioned for a long time the control systems development to an approach that leads to the centralization of its structure. Nowadays with the technological advances this approach is changed, and the control structure is evolving according to the distribution task paradigm. In this context, communication networks play a prominent role both in the control structure and in the system operation. At lowest control levels the communications are established by fieldbus networks which interconnect small controllers, sensors and actuators, in a process typically characterized by the cyclical exchange of small messages, usually with time critical properties.

The intensive investment in the development of control systems, its complexity and the type of tasks that play had become important to take into account all aspects of system components operation that can interfere in the system availability, security of people, or some environmental impacts. Thus as one of most important system component, the fieldbus network operation should be assessed in order to identify operation modes, that potentially contribute for a reduction of system dependability. In this context, the assessments of network performability are important to identify the main sources of performance degradation, which in the limit driving the system to fail.

In the assessment and validation domains, experimental methods are powerful tool that can be used to cause abnormal system operation conditions and to verify its behaviour. The fault injection techniques are included in these classes of methods. They allow at intentionally introducing system errors, and thus constitute an efficient solution in the evaluation of complex systems where the application of other techniques is difficult or some times fail.

This thesis is intended to contribute to the characterization of one widely used fieldbus networks in automation application domains – PROFIBUS-DP working in faults conditions. The characterization addresses PROFIBUS-DP performability aspects, when its operation is disturbed by event such as ones caused by electromagnetic interference (EMI). The EMI can induce operation modes that have potential impacts in the network availability and its capacity to guarantee the fulfilment of deadlines of real-time communications. In this area of PROFIBUS-DP performability there is a lack of published work and the existing ones is manifestly insufficient to provide a solid characterization of its performance in presence of faults. In this context this thesis intends to contribute to knowledge of PROFIBUS-DP behaviour in these scenarios.

To assess the network, it is developed a fault injection framework based on a fault injection infrastructure, which holds a real PROFIBUS-DP network. The fault injection infrastructure implements the physical fault injection technique, which is able to inject faults into a distributed system, according a stochastic process. The hardware of the injector module is developed to present both high level of controllability and high fault resolution, which at allows accurately injecting either single bit or multi bits errors in the communication bus. The communication nodes are developed in accordance with the PROFIBUS-DP standard (IEC-61158). The software and hardware structure of the communication nodes allows at acceding the data link layer and at obtaining most of relevant data related to protocol operation states.

One of purposes of network performabily assessment is to identify events that disturb the organization of the stations in the PROFIBUS-DP logical ring. In this context, they have been identified six events which have important potential impacts on the network performance. Three of these are related with token losses, and the remaining ones are associated to unintentional logical ring station removals. The events probability is inferred for different fault scenarios, as well as, is verified how the associated recovery mechanisms performs in such conditions. The fault injection experiments are also applied to assess the network real-time behaviour in presence of faults. In this context, for the multi-master network configuration mode they are observed significant impacts both on the network response time and on the fault tolerance mechanisms recovery time. From the performability viewpoint this allows to foresee a bad behaviour of this network configuration in faults scenarios.

Esta página foi intencionalmente deixada em branco

Résumé

Des limitations de nature technologique ont conditionné le développement des systèmes de contrôle à un abordage qui a conduit à la centralisation de sa structure. Avec l'évolution technologique cette abordage c'est modifiée et actuellement beaucoup de ces structures de contrôle sont basées sur le paradigme de la distribution de fonctions. Dans ce contexte, les réseaux de communication jouent un papier fondamental dans la structure et dans l'opération du système. Dans les niveaux de contrôle inférieurs les communications sont établies par des réseaux de terrain qui connectent les petits contrôleurs, capteur et actuactuers dans un processus typiquement caractérisé par l'échange de petits messages de nature cyclique, et usuellement avec des restrictions temporelle critiques.

Etant donné la nature des réseaux de terrain dans les systèmes de contrôle, l'accroissement de sa complexité, couts de développement du système et de son exploitation et aussi les fonction exécutées, les aspects relatifs aux effets de l'opération du réseau dans la disponibilité du système, la sécurité de personnes ou même ses impacts négatifs dans l'environnement sont devenue des aspect très important du fonctionnement do système. Ainsi le fonctionnement des réseaux de terrain, une des plus importantes composantes du système, doivent être évaluée de façon à identifier las causes responsable de la réduction du niveau de confiance en son fonctionnement. Dans ce contexte l'évaluation du fonctionnement du réseau de terrain en présence de fautes, est importante pour l'identification des principales sources de dégradation de sa performance et qui, dans un cas limite, puissent même rendre le système inopérant.

Dans le domaine de l'évaluation et de la validation de systèmes, les méthodes expérimentales sont des outils puissants, qui peuvent être utilisés pour modifier et évaluer les conditions normales de fonctionnement du système. La technique d'injection fautes, qui fait partie dune classe de méthodes qui permette l'introduction intentionnelle d'erreurs dans le système, constitue une solution efficace dans l'évaluation de systèmes complexes, dans lesquels l'application d'autres techniques est difficile ou échoue.

Le contribue de cette thèse est la caractérisation d'un réseau de terrain très diffusé dans le domaine d'applications de automatismes industriels - PROFIBUS-DP. La caractérisation se focalise sur les aspects de performance en présence d'erreurs de PROFIBUS-DP, quand le réseau est perturbé par des événements résultant d'interférences électromagnétiques (EMI). L'EMI peut induire des manières d'opération avec potentiels impacts dans la disponibilité du réseau de communication et dans la capacité d'assurer l'accomplissement des compromis pour les communications en temps réel. Dans l'évaluation du fonctionnement en présence de fautes du PROFIBUS-DP il existe peu de publications, et celles que existent sont insuffisantes pour donné une caractérisation solide de la performance du PROBIBUS-DP dans ce cadre opérationnel. Dans ce contexte, cette thèse prétend contribuer à une meilleure connaissance du comportement du PROFIBUS-DP face a un scénario de fautes.

Pour analyser le comportement du réseau, on a développé une infrastructure permettant l'injection d'erreur dans un réseau de terrain PROFIBUS-DP. L'infrastructure d'injection d'erreurs applique la technique physique d'injection de fautes dans un environnement distribué, selon un processus stochastique. Le matériel informatique du module d'injection a été développé de façon à manipulé facilement le processus d'injection de fautes et permettre une grande résolution des erreurs injectées, permettant d'injecter dans le bus de communication des fautes d'un seul ou plusieurs bits. Les nœuds de communication ont été développés conformément à la norme de PROFIBUS-DP (IEC-61158). La structure du logiciel et du matériel informatique de nœuds de communication permettent d'accéder à la couche de liaison de données et obtenir des informations importantes sur les états d'opération du protocole.

Un des objectifs présent dans l'évaluation de la performance de PROFIBUS-DP en présence de fautes a été l'identification d'événements qui dérangent l'organisation des stations dans l'anneau logique. Dans ce contexte, ont été identifiés six événements qui possiblement ont des impacts dans la performance du réseau de communication. Trois de ces événements sont corrélés avec des pertes de jetons, les autres sont associés au déplacement non intentionnel de station de l'anneau logique. La probabilité des occurrences a été estimée pour différents scénarios de fautes et a été vérifiée le fonctionnement des mécanismes de récupération de fautes dans ces conditions. Les expériences d'injection de fautes ont été également appliquées pour évaluer le comportement en temps réel du réseau en présence de ces fautes. Dans ce contexte en mode de configuration de réseau de multi-maître a été observé des impacts significatifs soit dans la réponse temporelle du réseau, soit dans les temps de récupération des mécanismes de tolérance aux fautes. Ce comportement laisse prévoir une mauvaise réponse de la configuration multi-maître dans des scénarios de fautes.

Esta página foi intencionalmente deixada em branco

Índice

Índice de Figuras

Esta página foi intencionalmente deixada em branco

Índice de Tabelas

Esta página foi intencionalmente deixada em branco

Lista de Acrónimos

AS-i	Actuactor and Sensor Interface
AFIT	Advanced Fault Injection Tool
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
ASPC2	Advanced Siemens PROFIBUS Controller 2
BCT	Bus Cycle Time
BEL	Bit Error Length
BER	Bit Error Rate
BIA	Berufsgenossenschaftliches Institut für Arbeitssicherheit
CAD	Computer Aided Design
CAN	Controller Area Network
CENELEC	Comité Europeu para a Normalização Electrotécnica
CIM	Computer Integrated Manufacturing
COTS	Commercial Off-The-Shelf
CPF	Communication Profiles Families
CPU	Central Processing Unit
CNC	Computer Numerical Control
CRC	Cyclic Redundancy Check
DA	Destination Address
DDLM	Direct Data Link Mapper
DEFINE	Distributed Fault Injection and moNitoring Environment
DOCTOR	integrateD sOftware fault injeCTiOn envirRonement
DP	Decentralised Periphery
EBCDIC	Extended Binary Coded Decimal Interchange Code
EMI	Electromagnetic Interference
ESPRIT	European Strategic Program on Research in Information Technology
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IS	Intrinsic Safety
FASST	Fault Tolerant Architecture with Stable Storage Technology
FC	Function Code
FCS	Frame Check Sequence

FDL	Fieldbus Data Link
FERRARI	Fault and Error Automatic Real-time Injection
FIAT	Fault Injection Based Automated Testing Environments
FIMBUL	Fault Injection and Monitoring using BUilt in Logic
FINE	Fault Injection and moNitoring Environment
FIST	Fault Injection system for Study of Transient fault effects
FMS	Fieldbus Message Specification
FPGA	Field Programmable Gate Array
FTAPE	Fault Tolerance And Performance Evaluator
HSA	High Station Address
IP	Internet Protocol
ISA	Instrumentation Society of America
ISO	International Organization for Standardization
LAAS	Laboratory for Automatics and Systems Analysis
LAN	Local Area Network
LAS	List of Active Stations
LASER	Light Amplification by Stimulated Emission of Radiation
LIVE	Low Intrusion and Validation Environment
MAC	Media Access Control
MAFALDA	Microkernel Assessment by Fault injection Analysis
MAFT	Multicomputer Architecture for Fault Tolerance
MAP	Manufacturing Automation Protocol
MARS	Maintainable Real-Time System
MBP	Manchester coded Bus Power
MBU	Multi Bit Upset
MMU	Memory Management Unit
MTBT	Mean Time Between Failures
MTTC	Mean Time To Catastrophic failure
MTTR	Mean Time To Repair
NFATE	Network Fault Tolerance and Performance Evaluator
NRZ	Non Return to Zero
OSI	Open System Interconnection
PA	Process Automation
PDU	Protocol Data Unit
PLC	Programmable Logic Controller
PROFIBUS	PROcess FIeld BUS
RISC	Reduced Instruction Set Computer
SA	Source Address
SCRIBO	Self-Checking RISC Board
SD	Start Delimiter
SDN	Send Data with No acknowledge

SDS	Smart Distributed Systems
SEL	Single Event Latchup
SEU	Single Event Upset
SFI	Software Fault Injector
SIL	Safety Integrity Level
SNCF	Société Nationale des Chemins de fer Français
SOC	System-On-a-Chip
SPARC	Scalable Processor ARChitecture
SRD	Send and Request Data with acknowledge
TAP	Test. Access Port
TCP	Transmission Control Protocol
T_{bit}	bit Time
T_{CT}	Token Cycle Time
T_{ID}	Idle Time
T_{TH}	Token Holding Time
TMR	Triple Modular Redundancy
TOP	Technical an Office Protocol
T_{RR}	Real Rotation Time
TS	This Station
T_{SL}	Slot Time
T_{TF}	Token Frame Time
T_{TO}	Time-Out Time
T_{TR}	Target Rotation Time
TÜV	Technischer Überwachungs-Verein
UART	Universal Asynchronous Receiver Transmitter
UDP	User Datagram Protocol
VHDL	VHSIC Hardware Description Language
VLSI	Very Large Scale Integration
WCRcT	Worst Case Recovery Time
WCRT	Worst Case Response Time

Esta página foi intencionalmente deixada em branco

Capítulo 1

Introdução

1.1 Contexto

Os modernos sistemas de produção foram ao longo dos tempos incorporando, profundas alterações nas suas estruturas, fruto de evoluções induzidas pela sua envolvente. Os mercados, e mais especificamente a sua volatilidade, muitas das vezes associada a fenómenos de moda geradores de uma elevada diversidade de produtos com um curto ciclo de vida, tiveram um contributo de relevo para esta dinâmica [Rembold93].

A evolução dos sistemas de fabrico efectuou-se através da crescente automatização de processos, acompanhada de integração de novas tecnologias. Um dos eventos percursores desta mudança ocorreu com a introdução de sistemas computacionais no apoio, ou mesmo na execução das tarefas de controlo. Este evento estabeleceu os alicerces para o aparecimento de novos paradigmas nas arquitecturas dos sistemas de controlo que se estendeu à própria estrutura dos sistemas de fabrico.

Tipicamente as primeiras arquitecturas de controlo seguiam uma estrutura centralizada, na qual todas as tarefas eram efectuadas num computador. Embora a introdução do computador tivesse constituído inequivocamente uma vantagem, o aumento em dimensão e a complexidade dos sistemas de controlo puseram em evidência limitações das arquitecturas centralizadas. Os desenvolvimentos tecnológicos que se verificaram ao longo das últimas décadas em áreas como a da micro-electrónica, permitiu o desenvolvimento de hardware com elevado desempenho a custos reduzidos. Assim, foram reunidas as condições técnicas e económicas que viabilizaram o aparecimento de paradigmas baseados na especialização e cooperação entre recursos que se encontram distribuídos.

Actualmente, os modernos sistemas de controlo, que populam o ambiente fabril, recorrem a modernas soluções de automação onde o uso intensivo de sistemas computacionais baseados em microprocessadores, e de sistemas de comunicações, permitem a integração e partilha de informação entre todos os intervenientes do processo. Ao nível da interconecção de pequenos controladores como controladores lógicos programáveis (PLC), controlo numérico por computador (CNC), com sensores e actuadores, redes baseadas em microprocessadores de baixo custo têm vindo a substituir as anteriores arquitecturas centralizadas com ligações ponto a ponto. Estas redes denominadas por redes de campo assumiram um papel dominante nos sistemas de automação industrial, abrangendo um vasto campo de aplicação que se estende a toda a área de fabrico e indústria de processos. Foram constituindo-se como suporte para os modernos sistemas de controlo distribuído [Decotignie93].

A incorporação de tecnologias, capazes de suportar um largo espectro de funcionalidades, fez surgir questões relacionadas com a correcta operação destes sistemas, nomeadamente com a necessidade de identificação de modos de operação de sistemas de elevada complexidade. A necessidade desta identificação torna-se mais importante para cenários em que o sistema é afectado por eventos que não derivam da sua normal operação. Os eventos fontes das perturbações podem ter diversas origens e podem ocorrer em qualquer estágio da vida do sistema, ou seja, desde a fase de concepção até à fase de operação passado pela implementação.

Nos sistemas electrónicos as perturbações manifestam-se através de erros, que podem levar os sistemas a apresentar uma degradação da qualidade do serviço ou mesmo um modo de operação não compatível com as especificações para as quais foram projectados. Em consequência pode resultar desses modos de operação, prejuízos que derivem de uma não total rentabilização de equipamentos de capital intensivo, ou em certos casos degradação ou inutilização de produtos processados por esses equipamentos. Em casos mais graves, podem mesmo colocar em perigo pessoas, bens, ou ter repercussões ambientais não toleráveis. Neste contexto, o funcionamento dos sistemas de controlo deve ser avaliado quanto ao efeito de tais perturbações no seu comportamento, nomeadamente no que respeita à confiança no seu funcionamento e à degradação de desempenho.

Esta avaliação é muitas das vezes uma tarefa de difícil execução. As principais dificuldades estão associadas à forma complexa como os erros são originados e como se propagam nos sistemas electrónicos, onde existe uma profunda inter-relação entre *hardware* e *software*. Este comportamento torna difícil a obtenção de um modelo analítico que descreva de forma precisa o funcionamento do sistema, obrigando muitas vezes ao recuso a técnicas experimentais para efectuar este tipo de análise.

1.2 O Sistema de Controlo

Um sistema de controlo consiste num conjunto de componentes que estão interligados de forma a estabelecer uma configuração que permita obter do objecto controlado a resposta desejada. A acção de controlo num dado sistema é efectuada tipicamente com base em duas informações: uma vinda do processo que está a ser controlado, e outra originária do utilizador, com a indicação da resposta desejada. Do ponto de vista tecnológico, actualmente grande parte dos sistemas de controlo são baseados na utilização de sistemas computacionais. Estes fazem parte de um sistema que pode ser decomposto em três subsistemas: processo controlado, controlador e o operador [Kopetz98] (Fig.1.1).

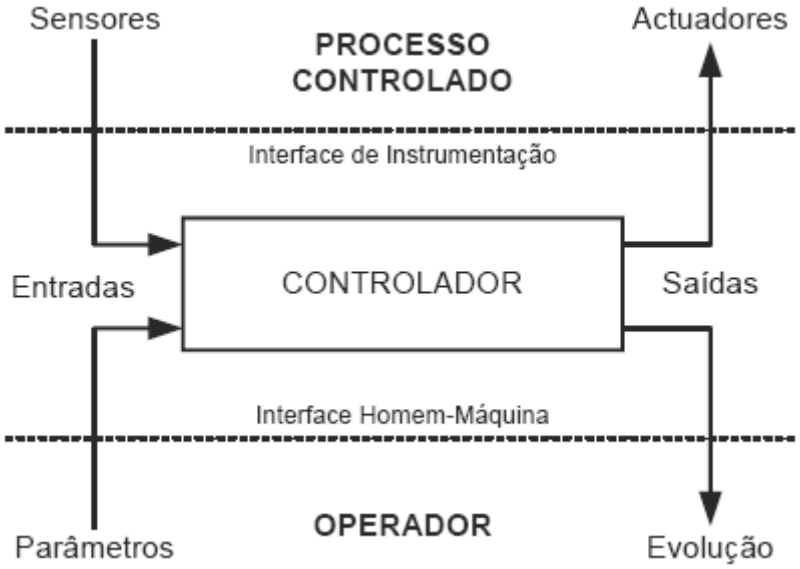

Figura 1.1 - O Sistema de Controlo

O sistema computacional é o elemento central do sistema de controlo, e desempenha a função de controlador. Na sua acção de controlo comunica com os demais subsistemas através de duas interfaces: a interface Homem-Máquina (H-M) e a interface de instrumentação.

A interface H-M permite trocar informação entre o operador e controlador. Para tal, esta interface é provida de dispositivos de entrada e saída que permitem a troca de informação inteligível ao operador, de que são exemplos os teclados os e *display's*. Através dos dispositivos de entrada o operador pode configurar o controlador transferindo parâmetros relevantes para serem usados no controlo do objecto controlado. Através dos dispositivos de saída o operador pode monitorizar e inteirar-se do estado processo.

A interface de instrumentação estabelece a comunicação entre o controlador e o processo controlado, e é constituída por sensores e actuadores. Os sensores permitem ao controlador obter amostras do estado do processo controlado. Com base no estado do processo (obtidos pelos sensores), conjugado com o valor

desejado para a resposta do sistema (fornecidos através da interface H-M) e de acordo com uma determinada lei de controlo, o controlador recorre aos actuadores para regular o sistema.

Características físicas associadas ao objecto controlado, que se reflectem na dinâmica do processo, ou mesmo relacionadas com a sua função, podem requerer que a acção reguladora obedeça a restrições temporais. Neste caso, os resultados obtidos a partir de operações matemáticas das leis de controlo não podem ser considerados correctos unicamente pela sua validade nesse domínio. A validade destes resultados depende também do instante em que estes são produzidos. Esta dimensão temporal confere-lhes a denominação de sistemas de controlo de tempo-real.

Desta forma, um sistema de controlo de tempo-real deve reagir quer a eventos que têm a sua origem no objecto controlado e no operador, ou gerar periodicamente eventos, aos quais está associado o cálculo de um novo valor para a regulação do objecto controlado. O período entre os eventos que estão na origem da produção de novos valores para a regulação é imposto pelas características do processo controlado, e o limite temporal máximo em que os resultados devem ser produzidos é designado por *deadline*[1].

Dependendo das características do sistema o desrespeito de uma ou mais *deadlines* podem ter implicações diversas no comportamento do sistema. Neste contexto, de acordo com o tipo de resposta o sistema pode assumir diferentes designações. No caso do não cumprimento de uma *deadline* resultar uma degradação da resposta do sistema, mas que ainda assim esteja dentro de parâmetros que possam ser considerados aceitáveis, o sistema toma a designação de sistemas de controlo de tempo-real com requisitos temporais moderados (*soft real-time control systems*). Quando do não cumprimento de uma *deadline* resulte uma resposta do sistema inaceitável, o sistema de controlo toma a designação de sistemas de controlo com requisitos temporais críticos (*hard real-time control systems*) [Kopetz98].

1.2.1 Sistemas de Controlo Distribuído de Tempo-Real

O crescimento exponencial da utilização de sistemas de controlo por computador verificado na área da automação e na indústria de processos, durante as décadas de 70 e 80, foi acompanhado igualmente por uma cada vez maior integração de tarefas no sistema de controlo.

Este cenário fez aumentar de forma significativa a dimensão de tais sistemas levando a que as arquitecturas centralizadas baseadas em ligações ponto a ponto apresentassem um conjunto de factores, quer de cariz económico, quer ao nível da complexidade de instalação e manutenção, que eram manifestamente

[1]Ao longo da dissertação são adoptados alguns termos em inglês, a sua utilização deriva de ser comum o seu emprego na abordagem técnica dos assuntos, ou de estes fazerem parte de um processo avançado de assimilação, com a sua frequente utilização durante o estabelecimento de uma conversação.

desfavoráveis à sua utilização [Pleinevaux88] [Cavalieri97] [Zurawski05]. Em consequência, uma nova tendência apontando para a distribuição de tarefas, fez surgir os designados sistemas de controlo distribuídos. Nestes, o sistema de controlo é dividido em blocos funcionais que encapsulam funções lógicas bem definidas no sistema, em unidades designadas por nós, que operam com base na troca de informação entre unidades.

Neste contexto, um sistema de controlo distribuído é constituído por um conjuntos de nós que se encontram ligados por um sistema de comunicações [Kopetz98] [Pimentel90] (Fig. 1.2). Os nós são providos de maior ou menor capacidade de cálculo de forma a lhes permitir executar operações locais relacionadas com a sua função no sistema.

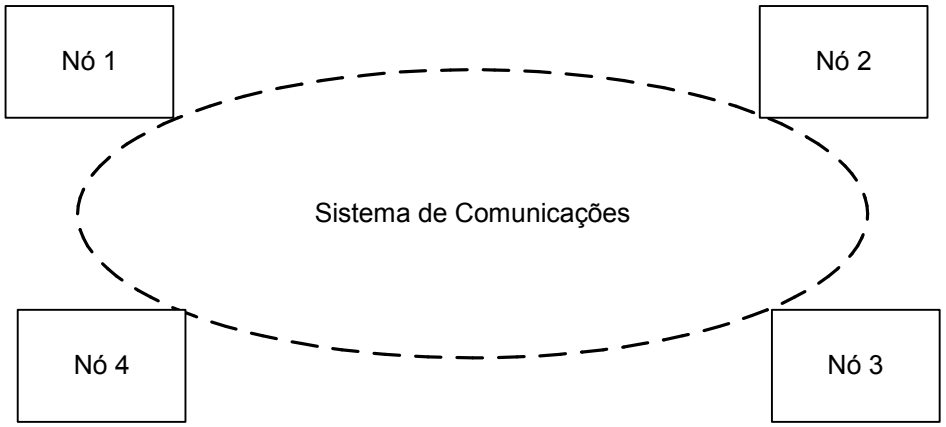

Figura 1.2 - Sistema distribuído de tempo-real

O sistema de comunicação é constituído por recursos físicos (*hardware*) e lógicos (protocolos) que permitem a troca de informação entre os membros do sistema. Actualmente as redes de campo são o sistema de comunicações dominante em aplicações de controlo. Estas encontram-se principalmente vocacionadas para interligar pequenos controladores com sensores e actuadores, e a sua operação é caracterizada pela troca de informação suportada por mensagens de tamanho reduzido que geralmente têm associadas restrições temporais rígidas.

A utilização da rede de comunicações tem contudo efeitos na transmissão da informação. Esses efeitos fazem-se sentir nomeadamente no domínio do tempo através da introdução de atrasos. Quando a informação é enviada na rede de comunicações, a informação é afectada por um atraso desde que é produzida até ao momento em que é entregue no nó a que se destina. Este atraso é influenciado por diversos factores, entre os quais se incluem a velocidade de transmissão, a dimensão das mensagens que transportam a informação, o tráfego na rede de comunicações e o processamento da informação pelo protocolo de comunicação. A não consideração deste atraso, ou sua incorrecta caracterização pode provocar degradação de desempenho do sistema de controlo ou levá-lo a um estado de operação incorrecto [Kim94] [Shin95] [Yook00] [Yook02]. A latência e o *jitter* são parâmetros que caracterizam estes atrasos.

A latência é a medida do tempo que medeia o início da transmissão de uma mensagem e a sua recepção no nó de destino, efectuado ao nível da camada de transporte do protocolo [Kopetz98] [ISO94]. O *jitter* é a medida da variação da latência na rede de comunicação ao longo do tempo.

O *jitter* é um parâmetro particularmente sensível a cenários em que a normal carga do sistema é alterada, nomeadamente a originada por solicitações significativas do sistema de comunicações que ocorrem de forma pontual. A sua origem pode estar relacionada com a resposta a eventos anormais como sejam erros que resultam de perturbações temporárias de origem electromagnética. Desta forma, as redes de comunicação de tempo-real devem apresentar baixa latência do protocolo e *jitter* reduzido.

A operação do sistema de comunicações obriga a sua componente física, cabos ou outro meio de suporte à comunicação a disporem-se ao longo da área de cobertura das comunicações. Desta forma o sistema coexiste com factores ambientais de natureza agressiva, que expõe a infra-estrutura e torna-a mais susceptível a ocorrência desse eventos.

Ao estabelecer a ligação entre os diferentes agentes do sistema de controlo, o sistema de comunicações torna-se assim num dos recursos mais críticos do sistema, uma vez que a sua falha ou desvios ao seu normal funcionamento, causa a inoperacionalidade do sistema ou impactos significativos na sua operação.

Uma vez que a operação dos sistemas distribuídos de controlo dependam fortemente da performance do sistema de comunicações, a sua avaliação torna-se um aspecto importante que deve ser considerado, particularmente em cenários onde as comunicações tenham requisitos de tempo-real e operem em ambientes propícios à ocorrência de erros.

1.3 Sistemas Tolerantes a Faltas

A cada vez maior importância assumida pelos sistemas de computacionais nos mais diversos domínios da sociedade, fez emergir um conjunto de disciplinas associadas à área da ciência dos computadores e de aplicação no domínio da engenharia. Em algumas dessas disciplinas começou a existir uma particular preocupação quanto à forma como os sistemas computacionais efectuam as suas tarefas, tendo sido dado particular ênfase a duas questões específicas:

- Qual o grau de confiança depositado no serviço fornecido;
- Qual a eficiência com que o serviço é fornecido quando o sistema é perturbado no seu normal funcionamento.

A primeira questão está relacionada com a confiança no funcionamento, nomeadamente em factores que se relacionam com aspectos como fiabilidade, disponibilidade e segurança (*safety*). A importância destes foi acrescida quando

na década de 60 se tornou evidente a necessidade dos sistemas computacionais trabalharem de forma contínua e sem erros, principalmente quando estes operam sistemas que possam pôr em causa a integridade de pessoas ou de avultados investimentos. Numa primeira fase estes requisitos eram tipicamente impostos por aplicações no domínio da indústria espacial, aeronáutica, em sistemas de controlo de tráfego ferroviário e no controlo de sistemas de produção de energia eléctrica de origem nuclear.

Este cenário teve evoluções significativas, quer por questões do aumento generalizado da complexidade dos sistemas, quer por questões de cariz sociológico, nomeadamente as relacionadas com a necessidades das empresas transmitirem uma imagem de qualidade dos seus produtos. Actualmente as questões relacionadas com a confiança no funcionamento são consideradas nos mais diversos tipos de aplicação, desde as mais sofisticadas às de utilização de uso quotidiano [Lee90].

Nos sistemas de controlo industrial, a confiança no funcionamento foi numa primeira fase, tida em consideração por motivos económicos relacionados com paragens da produção, ou do risco de danificar os equipamentos. Só mais tarde estas preocupações se estenderam à segurança de pessoas ou da protecção do ambiente [Groover00].

A redução da confiança no funcionamento está associada a um conjunto de factores com incidência nos diferentes estágios do ciclo de vida do sistema, ou seja da fase de concepção, até à de operação passando necessariamente pela implementação. Pode ter origem em factores que resultam da intervenção humana, ou de interacções de natureza física com elementos do ambiente em que o sistema opera.

A minimização dos impactos na confiança no funcionamento provocados pelos factores referidos requer, a utilização de forma combinada, de métodos, técnicas e tecnologias, que deverão ser usadas numa primeira fase no sentido da diminuição de todos os factores que potencialmente possam vir a interferir na normal operação do sistema. Isto passa pela utilização técnicas de fabrico e de outras metodologias que minimizem a introdução de erros de projecto e de implementação na estrutura do sistema.

Numa outra vertente assume-se a impraticabilidade de projectar *hardware* e *software* capaz de efectuar a cobertura da totalidade das situações que originam estados de operação incorrecta durante o funcionamento do sistema. Isto é conseguido pela introdução de mecanismos que permitem detectar, confinar e recuperar das referidas situações.

A segunda questão aponta para o conceito de desempenho na presença de faltas (*performability*)*:* utilizado para denominar, a eficiência com que um sistema disponibiliza os seus serviços, quando este vê a sua operação afectada por eventos externos ao seu normal funcionamento. Este conceito foi introduzido nos finais da década de 70, quando começou a ser incorporado nos sistemas, um determinado nível de tolerância a situações de operação incorrecta. Num estado de operação incorrecto, estes mecanismos são activados, e durante a sua actuação

ou até que o sistema seja reparado, o sistema sofre uma degradação da sua operação. Neste contexto o desempenho na presença de faltas utiliza de forma combinada o conceito de performance e da confiança no funcionamento, para obter medidas que caracterizam o comportamento do sistema quando é afectado por estados de operação incorrectos.

Embora seja uma medida muito específica, comparativamente a medidas de confiança no funcionamento, a sua obtenção é crucial para a avaliação do comportamento de muitos sistemas, nomeadamente daqueles que apresentam elevado requisitos de segurança (*safety*), mas que podem suportar a degradação de desempenho provocada pelas perturbações, e ainda assim continuar a operar dentro das especificações para que foram concebidos [Mayer95].

1.3.1 Técnicas de Validação e de Análise de Desempenho

A obtenção de níveis elevados de confiança no funcionamento de sistemas complexos é um processo difícil, que não é alcançável somente através da aplicação de técnicas e metodologias adequadas durante a fase de concepção e de desenvolvimento. A validade das opções tomadas tem de ser verificada, sendo assim necessário avaliar a operação do sistema nas mais diversas condições de funcionamento. Da mesma forma, a introdução de mecanismos que permitem tolerar estados incorrectos de operação (faltas), não deve ser analisada somente numa perspectiva do seu nível de cobertura, mas também deve abranger a análise da inerente degradação de desempenho introduzida pela sua operação.

O comportamento do sistema deve ser assim analisado de forma a permitir revelar erros introduzidos na concepção, ou a presença de potenciais vulnerabilidades que possam vir a afectar a correcta operação do sistema, e simultaneamente obter medidas de desempenho na presença de faltas que permitam caracterizar a degradação da função do sistema nos mais diversos cenários de operação.

Estas análises podem processar-se através da utilização, quer de métodos analíticos, quer de métodos experimentais.

1.3.2 Métodos Analíticos

Os métodos analíticos apresentam-se tipicamente como uma solução compacta e mais económica em relação aos demais métodos. Este método consiste na aplicação de técnicas que recorrem à matemática, à álgebra e à lógica, na modelação de sistemas, apresentando-se como uma ferramenta importante, quer na fase de concepção e de desenvolvimento, quer como ferramenta de análise do comportamento de sistemas.

No apoio à obtenção de confiança no funcionamento, a aplicação de métodos formais na fase de concepção e desenvolvimento, permite apresentar uma

linguagem formal sem ambiguidades, e assim, eliminar potenciais causas de erros. Os métodos formais podem também ser usados na verificação do sistema, de forma a provar que a sua concepção e implementação estão conformes com as suas especificações [Kopetz98] [Hedberg01].

Uma outra importante vertente dos métodos analíticos consiste na aplicação de técnicas de modelação na simulação de sistemas. Estes apresentam-se como uma ferramenta poderosa em aplicações como a previsão de faltas, ou na análise de desempenho. Diversas técnicas são aplicadas nesta área de entre as quais se destacam, pela sua grande utilização entre a comunidade científica, os métodos baseados na análise de espaço de estados, como é o caso das cadeias de Markov e outros que se baseiam em redes de Petri, nomeadamente as estocásticas, [Haverkort96], [Trivedi93], [Johnson88], [Bondavalli99].

1.3.3 Métodos Experimentais

Os métodos experimentais permitem obter informação relacionada com a operação de sistemas, podendo a sua aplicação ser efectuada com propósitos distintos, como sejam o teste, ou numa abordagem mais profunda com o objectivo de proceder à validação ou análise da confiança do seu funcionamento.

Tipicamente o teste consiste na exercitação do sistema estimulando as suas entradas e verificando se a sua resposta corresponde à função para a qual foi projectado. Este processo permite identificar erros no funcionamento do sistema que não foram detectados em estágios anteriores, e a sua utilização é muito frequente no teste de *hardware* e de *software*. No *hardware* os testes são usados na maior parte das vezes para detectar defeitos na produção de circuitos integrados. No *software* os testes são utilizados para identificar e remover erros de concepção [Folkesson99].

Embora este tipo de testes contribuam positivamente para aumento da confiança no funcionamento dos sistemas, os seus resultados são insuficientes para alcançar elevados níveis de confiança no funcionamento, uma vez que estes são focados numa perspectiva meramente funcional, desligada da dimensão temporal que caracteriza as tarefas de muitos sistemas. Assim, esta abordagem não se adequa à validação de sistemas de tempo-real onde a dimensão temporal e funcional estão interligadas.

Uma das formas mais utilizadas para verificar a conformidade de sistemas, passa por confrontá-los durante a operação com um conjunto de eventos representativos de situações anormais, nomeadamente de faltas que ocorram no seu ambiente. Uma das técnicas de eleição orientada para a validação é designada por injecção de faltas. Esta técnica consiste na realização de experiências controladas nas quais são intencionalmente injectadas faltas no sistema e o seu comportamento é analisado em relação a estas condições de funcionamento [Arlat90]. As técnicas de injecção de faltas estão vocacionadas para ser aplicadas em duas tarefas distintas de validação [Arlat93] [Kopetz98].

A primeira, para supressão de faltas, é aplicável durante a fase de implementação (teste e diagnóstico) como forma de acelerar a ocorrência de incidências que activem erros que estejam latentes no sistema. Um caso particular de aplicação é a verificação da eficiência dos mecanismos de tolerância a faltas, onde é necessário um número extremamente elevado de activações para a sua verificação.

A segunda, para previsão de faltas, cuja aplicação se processa de forma a obter informação relacionada com a confiança no funcionamento do sistema. A sua aplicação também pode ser usada na validação de modelos de previsão de faltas usados em simulação [Arlat93].

Em síntese, os métodos analíticos fornecem o suporte para o desenvolvimento de poderosas ferramentas oferecendo soluções de baixo custo e menos trabalhosas. Contudo, os resultados obtidos por estas ferramentas estão dependentes da qualidade do modelo utilizado para descrever a operação do sistema. Da mesma forma necessita de parâmetros que permitam definir pontos de funcionamento coerentes com a real operação do sistema. Neste contexto, em cenários de elevada complexidade os métodos experimentais são uma boa opção na validação do funcionamento do sistema, assim como na obtenção de informação do sistema que permita, alimentar, e validar modelos analíticos, apresentando-se as técnicas de injecção de faltas como uma das técnicas de eleição para o efeito.

1.4 Motivação

Com o aumento de aplicações com elevados requisitos de segurança (*safety*) implementados em dispositivos electrónicos programáveis, como microprocessadores e autómatos programáveis (PLC's), surgiu a necessidade de fazer com que os sistemas que incorporam esta tecnologia sejam também eles seguros. Nos últimos anos tem sido desenvolvido um significativo esforço para definir normas que estabeleçam procedimentos que permitam a aplicação da tecnologia sem prejudicar futuros desenvolvimentos tecnológicos nem pôr em causa a segurança global do sistema.

Neste esforço enquadram-se normas como a EN954 e ISO13849 (*Safety of Machinery – Safety Related Parts of Control Systems*) o IEC61508 (*Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems*) [ISO99] [IEC04].

A EN954 está direccionada para a segurança dos sistemas de controlo, com aplicação no domínio máquinas de ferramentas, excluindo especificamente o subsistema de comunicação.

O IEC61508 está orientado para a segurança funcional dos sistemas electrónicos programáveis, abrange os diversos aspectos da segurança do sistema

ao longo do seu ciclo de vida. A norma define 4 níveis de integridade de segurança (SIL *Safety Integrity Level*), assim como, os requisitos para os alcançar (Tab. 1.1). Os níveis de integridade resultam de uma avaliação dos perigos (*hazards*) envolvidos e da aceitação de riscos durante a operação do sistema, conferindo assim uma representação da probabilidade do sistema operar de acordo com as suas especificações durante um determinado intervalo de tempo.

Níveis de integridade de segurança – SIL	**Modo de operação**	
	Baixa utilização (falhas/hora)	**Elevada operação** (falhas/hora)
SIL 4	$\geq 10^{-5}$ a $<10^{-4}$	$\geq 10^{-9}$ a $<10^{-8}$
SIL 3	$\geq 10^{-4}$ a $<10^{-3}$	$\geq 10^{-8}$ a $<10^{-7}$
SIL 2	$\geq 10^{-3}$ a $<10^{-2}$	$\geq 10^{-7}$ a $<10^{-6}$
SIL 1	$\geq 10^{-2}$ a $<10^{-1}$	$\geq 10^{-6}$ a $<10^{-5}$

Tabela 1.1 - Níveis de integridade de segurança acordo com o IEC61508.

Apesar da IEC61508 constituir uma norma suficientemente abrangente que engloba o subsistema de comunicações, é também uma norma relativamente recente nomeadamente em relação à existência de aplicações distribuídas e das redes de campo que suportam uma grande parte destas aplicações. Dada a incapacidade de grande número de redes de campo satisfazerem os vários níveis de integridade requeridos pelos sistemas que as utilizam, durante muito tempo, houve a necessidade de se assumir as limitações da sua utilização no suporte de aplicações distribuídas críticas. Estas limitações tinham assim que ser supridas através da utilização de tecnologia convencional ou de redes de comunicação especiais.

Num esforço para ultrapassar estas limitações os diversos fornecedores de redes de campo, fizeram evoluir as suas redes de forma a permitir soluções abertas integradas para o suporte de aplicações seguras. O elemento comum a estas evoluções está relacionado com tipo de abordagem utilizada, que considera a rede de comunicação como intrinsecamente insegura. Assim, para que possam suportar aplicações seguras, os protocolos devem ser providos de meios que permitam fornecer serviços seguros para as aplicações, independentemente do tipo de rede em que as aplicações operam. Isto é conseguido através da inclusão de mecanismos nos protocolos que permitem que a rede falhe de forma segura. Vários mecanismos são empregues para resolver os vários cenários de erro que podem ocorrer durante a operação da rede de comunicações. Na tabela 1.2 apresenta-se um mapeamento da utilização de um ou mais mecanismos aplicáveis à resolução dos vários cenários de erro, recomendado pela BIA, uma organização Alemã responsável pela investigação e teste na área da segurança [Piggin00].

Actualmente existem já várias soluções de redes de campo seguras baseadas em redes intrinsecamente inseguras, como é o caso do SafetyBUS, CANopen e DeviceNet desenvolvidos sobre o CAN, e o PROFIsafe que opera sobre o

PROFIBUS-DP. Este último permite numa mesma rede a coexistência de nós de comunicação seguros, com outros não seguros. Recentemente no âmbito do IEC foi criado um grupo de trabalho WG12 para definir uma estrutura comum na abordagem no desenvolvimento de redes de campo seguras, que se baseiam em redes intrinsecamente inseguras [Felser04].

	Medidas						
Erros de transmissão	Runnig numbers	Time stamp	Time echo	echo	ID for send and receiver	Data CRC	Redundancy plus cross check
Mensagem repetida	√	√	–	–	–	–	√
Perda de mensagem	√	–	–	√	–	–	√
Mensagem inserida	√	–	–	√	–	–	√
Falha de sequência	√	√	–	–	–	–	√
Corrupção de dados	–	–	–	√	√	√	√
Atrasos	–	√	√	–	–	–	–

Tabela 1.2 - Mecanismos de prevenção de erros recomendada pela BIA.

Para garantir operação segura de um sistema é imperioso que o seu sistema de controlo seja capaz de reagir a eventos críticos num tempo que permita evitar *hazards*. Desta forma, funções críticas do sistema ficam dependentes do tempo que as mensagens possam demorar no sistema de comunicações, sendo vital o conhecimento das suas características de funcionamento, nomeadamente do máximo atraso que as mensagens possam sofrer. Embora a utilização de *time stamps* ou a análise do tempo de recepção de mensagens permita detectar a chegada de mensagens que não cumpram as *deadlines* e fazer entrar os nós de comunicação num estado de operação seguro, é necessário estar na posse de uma completa caracterização da operação da rede em cenários de faltas, de forma a poder ter uma perspectiva global das várias vertentes da confiança no funcionamento.

Neste contexto é motivação da presente dissertação avaliar o comportamento do PROFIBUS-DP, em cenários típicos do ambiente industrial em que opera. Nomeadamente avaliar o seu desempenho identificando características do funcionamento da rede que possam reduzir a sua disponibilidade. Situações que são importantes na operação do PROFIBUS-DP e que tendem a ser potenciadas quando o perfil PROFfIsafe é utilizado, designadamente pelo disparo mais frequente de mecanismos que podem ter com efeito secundário uma redução da disponibilidade.

Neste ambiente encontram-se muitos equipamentos que são fonte de radiação electromagnética capaz de corromper a informação transmitida através da rede. Este tipo de eventos pode ter impactos diversos que podem ir desde a simples retransmissão da informação afectada, até interferência na organização dos elementos que constituem a rede de comunicação. No PROFIBUS-DP as

estações formam um anel lógico e com base nesta organização fazem a gestão de acesso ao meio físico de comunicação. A perturbação do mecanismo que suporta esta organização virtual das estações, pode levar a que estas fiquem durante algum tempo sem possibilidade de efectuar os seus serviços na rede. A caracterização deste modo de funcionamento é portanto de grande importância no contexto da resposta temporal que as estações podem assegurar.

A reforçar a necessidade desta avaliação contribui também o facto de muita da análise temporal do PROFIBUS-DP ter sido efectuada até ao momento predominantemente com base nas características de desempenho dos protocolos, tendo sido dada pouca ênfase a aspectos do desempenho na presença de faltas desta rede.

Esta avaliação é efectuada numa perspectiva experimental tendo como elemento instrumental uma ferramenta de injecção de faltas desenvolvida para o efeito. Esta ferramenta permite alterar a informação das mensagens que circulam na rede de comunicação, de forma semelhante à que ocorre na realidade em consequência de interferências do meio. Não obstante, este método ser inerentemente mais trabalhoso, e obrigar a demoradas experiências, não tirando benefícios de aspectos como a compressão temporal que é possível efectuar em simulação, ou a utilização ferramentas matemáticas suportadas por essas técnicas, este método apresenta-se como a opção que mais se adapta a este tipo de análise. Esta adaptação é tanto maior quanto os sistemas envolvidos são de elevada complexidade, como acontece com os protocolos do PROFIBUS-DP. O mesmo é valido na identificação de modos de operação que resultam de um processo complexo, onde os mecanismos de tolerância a falhas podem ser activados de forma combinada.

Desta forma este método permite identificar e quantificar os eventos relevantes de acordo como eles ocorrem nos sistemas reais.

1.5 Contribuições da Dissertação

Englobado na avaliação do funcionamento do PROFIBUS-DP, são apresentadas nesta dissertação as seguintes contribuições:

1. Identificação de cenários de faltas que afectem a estabilidade organizacional das estações do PROFIBUS-DP, ou seja que afectam a integridade da anel lógico em que assenta muito do funcionamento do PROFIBUS-DP. Estes cenários podem surgir segundo duas vertentes. Uma resultante da remoção do anel lógico de estações. Outra resultante da perda do mecanismo que faculta o acesso das estações ao meio físico (*token*), impossibilitando toda a rede de operar até que esta recupere através da geração de um novo *token*.

2. Determinação da probabilidade da ocorrência dos eventos que originam instabilidade no anel. Análise qualitativa do comportamento dos mecanismos repensáveis pela recuperação das situações descrita no ponto anterior, consubstanciados em medidas qualitativas de desempenho na presença de faltas, relativas ao tempo médio de recuperação, assim como, medidas de desempenho da rede nestas condições.
3. Análise global do desempenho do sistema em cenários de faltas para diferentes configurações de carga. Determinação da probabilidade do não cumprimento de *deadlines* em função de parâmetros de configuração da rede e taxas de erros. Identificação do tempo resposta médio e pior caso de resposta (*Worst Case Response Time*), para as mesmas condições de operação.

1.6 Estrutura da Dissertação

A dissertação encontra-se estruturada em seis capítulos. Um primeiro de cariz introdutório do qual esta introdução faz parte.

No capítulo 2 é apresentada uma panorâmica das comunicações industriais com incidência nas redes de campo, onde é efectuada uma primeira abordagem à rede PROFIBUS-DP. Na descrição da rede são apresentados os vários perfis do PROFIBUS-DP, assim como, é apresentado o trabalho relevante no domínio da análise do desempenho de tempo-real da rede.

É efectuada uma incursão à análise da operação da rede em cenário de falhas onde é constatado o nível insuficiente de trabalho publicado na área, apontando para a necessidade de uma avaliação mais aprofundada da operação da rede em modo degradado.

No capítulo 3 são apresentadas técnicas para avaliação da confiança no funcionamento de sistemas, nomeadamente na vertente da verificação, supressão de faltas e previsão da sua incidência. É dada particular atenção às técnicas de injecção de faltas, sendo estas descritas de acordo com os principais vectores tecnológicos de aplicação: simulação; injecção física por hardware; injecção física por software, e técnicas híbridas.

A importância destas técnicas é realçada num contexto de aplicação à avaliação do funcionamento por injecção de faltas.

No capítulo 4 é descrito o ambiente de injecção de faltas utilizado para avaliar a operação do PROFIBUS-DP em cenários de faltas. Esta descrição centra-se na componente funcional e das soluções implementadas nos módulos que constituem o sistema de injecção de faltas.

É igualmente apresentada a fundamentação matemática e a metodologia subjacente às experiências de injecção de faltas utilizadas na avaliação do desempenho na presença de faltas e de confiança no funcionamento de sistemas e sua aplicação à avaliação do PROFIBUS-DP.

No capítulo 5 o PROFIBUS-DP é revisitado nomeadamente ao nível da descrição dos protocolos da camada de ligação de dados (FDL – *Fieldbus Data Link*). Esta descrição serve de suporte à avaliação efectuada ao protocolo, que é constituída por uma avaliação preliminar na qual são identificadas as principais causas de instabilidade na operação da rede, os seus impactos e probabilidades associadas aos eventos que as constituem.

A complementar esta avaliação é efectuada uma outra avaliação centrada no desempenho na presença de faltas da rede para cenários típicos de carga e de faltas.

A dissertação é finalizada com um capítulo 6 de conclusões.

1.7 Publicações Resultantes da Dissertação

- ***Assessment of PROFIBUS Networks Using a Fault Injection Framework***, Proceedings of 10th Conference on Emerging Technologies and Factory Automation, ETFA 2005, Vol. 1, pp. 415-423, IEEE, 2005.

  Prémio para o melhor *paper on Factory Automation* ETFA 2005.

- ***Experimental Analysis of Outage Times for PROFIBUS Networks***, Proceedings of 32nd Conference on Industrial Electronics Society, IECON 2005, pp. 421-426, IEEE, 2005.

- ***A Framework for Dependability Evaluation of PROFIBUS Networks***, Proceedings of the International Symposium on Industrial Electronics, IEEE, 2003.

- ***Dependability Modelling Techniques for Electronics Systems***. Proceedings of Sixth European Space Power Conference, pp335-342, 2002.

Esta página foi intencionalmente deixada em branco

Capítulo 2

Comunicações em Ambiente Industrial

2.1 Introdução

No início da década de 80, os índices de automatização relativos à capacidade instalada de equipamentos de automação em algumas empresas começou a tomar proporções assinaláveis. Contudo, não obstante os naturais benefícios inerentes à utilização deste tipo de equipamentos, começaram a ser identificados, problemas relacionados com a sua eficiente utilização, quando o seu desempenho era analisado numa perspectiva de utilização global. Na essência destes problemas residiam questões de ordem tecnológica, resultantes da dificuldade de integração de equipamentos de diferentes fabricantes, num ambiente caracterizado pela heterogeneidade de soluções.

Com o intuito de identificar soluções e de obter melhorias ao nível do desempenho global das indústrias, foram lançados nos anos 80 projectos como o MAP (*Manufacturing Automation Protocol*) e o TOP (*Technical and Office Protocol*): o MAP, com o objectivo de obter uma especificação assente no modelo de referência OSI para comunicações ao nível da fábrica [Schutz88], e o TOP, com objectivos semelhantes mas com aplicação na área técnica e administrativa.

Com o MAP, foi dado um forte impulso ao aparecimento de redes para aplicações industriais, que possibilitassem a interligação de equipamentos com as especificidades dos que são utilizados neste ambiente e assim reduzir ou eliminar as denominadas ilhas de automação [Daigle88].

Numa perspectiva mais ampla surgiu o conceito de CIM – *Computer Integrated Manufacturing*. Nesta, os fluxos de informação de uma indústria são

integrados quer ao nível horizontal, quer ao nível vertical ligando os vários sectores que a constituem, numa perspectiva da integração da totalidade das suas actividades. Independentemente do tipo de indústria, quer seja ela de manufactura ou de processos, a sua estrutura organizacional pode ser modelada por uma pirâmide na qual estão representados os vários níveis hierárquicos que a compõem, assim como, aspectos que caracterizam os fluxos de informação da própria estrutura (Fig. 2.1).

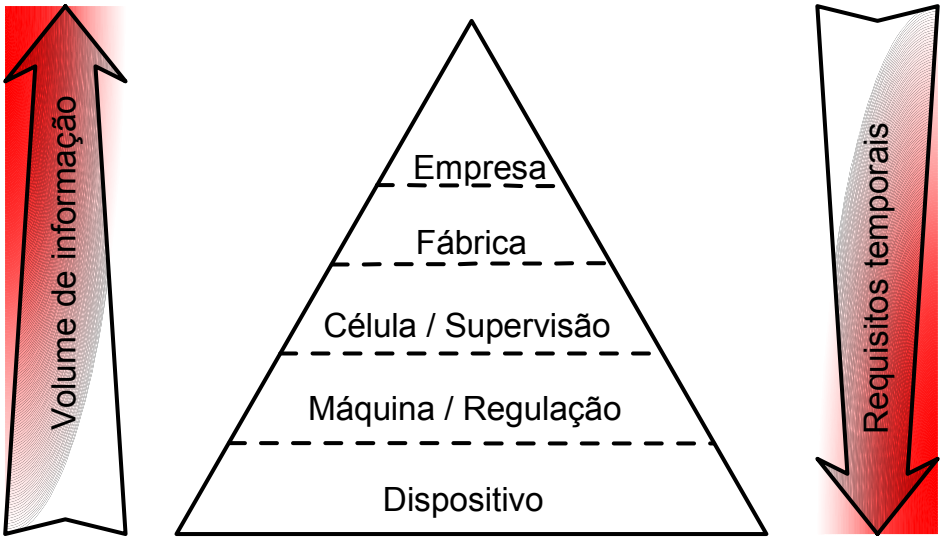

Figura 2.1 - Modelo hierárquico da organização de uma empresa e caracterização dos fluxos de informação.

A estes níveis correspondem funções distintas dentro da organização, que necessariamente fazem uso de informação com características igualmente distintas. Desta forma, quando se analisa a informação ao longo da estrutura hierárquica, esta apresenta atributos específicos que por vezes são antagónicos, nomeadamente no que respeita ao seu volume e aos seus requisitos temporais. Assim, nos níveis hierárquicos superiores decorrem actividades que envolvem o processamento de elevados volumes de informação sem particulares restrições temporais. Em sentido oposto, nos níveis inferiores são os requisitos temporais que imperam, associados a pequenos volumes de informação com periodicidade cíclica de elevada frequência (Tab. 2.1).

Sistemas baseados em computadores asseguram a execução das funções que são desempenhadas em cada nível. Estes por sua vez estão interligados por redes de comunicação que estabelecem a ligação entre equipamentos de um mesmo nível e simultaneamente asseguram as comunicações com outros que se situam nos níveis hierárquicos adjacentes [Decotignie05]. A existência de informação, com as condicionantes descritas inviabiliza a adopção de um único tipo de rede para aplicações industriais, obrigando necessariamente à existência de vários tipos de rede, com perfis que se adaptem ao contexto de aplicação. Com base nestas especificidades do ambiente industrial as redes de comunicação podem ser agrupadas em três grandes grupos [Cucej04] [Rembold93] [Pimentel90]:

	Volume informação	Tempo de transmissão	Frequência das transacções
Empresa	*Mbytes*	horas/minutos	dia / turno
Célula	*Kbytes*	Segundos	horas / minutos
Máquina	*bytes*	centenas microsegundo ... centenas milissegundo	dezenas de milissegundo ... centenas de milissegundo
Dispositivo	*bit*	milissegundo ...dezena de milissegundo	milissegundos

Tabela 2.1 - Requisitos para redes de comunicação.

- **Redes de Fábrica**

Este tipo de rede constitui o nível mais alto do sistema de comunicações da indústria, sendo constituído por redes capazes de suportar elevados débitos de informação sem contudo apresentar requisitos temporais relevantes (*backbone*). As comunicações são essencialmente utilizadas para interligar os principais centros de decisão estratégica como sejam: o departamento comercial, financeiro, de engenharia, planeamento e gestão da produção. Das actividades desenvolvidas nestes centros de decisão resultam ordens no sentido descendente da estrutura hierárquica que incluem informação relativa a ordens de fabrico e de escalonamento da produção. Com o objectivo de manter regulado todo sistema fabril os níveis de decisão são realimentados com informação relativa ao estado dos níveis operacionais, dos quais é enviado um fluxo de informação, que contém de entre outros, indicadores de qualidade da produção, necessidades de matérias primas e de ferramentas.

- **Redes de Célula**

Este tipo de rede é utilizado para estabelecer a ligação dos equipamentos responsáveis pela coordenação das operações de produção. Com o objectivo de aumentar a eficiência nas indústrias de manufactura, o equipamento fabril é por vezes agrupado segundo afinidades funcionais em *layouts* designados por células de fabrico. Desta forma, é possível reduzir o tempo de fabrico na componente associada ao tempo de transporte entre postos de trabalho. Quando esta configuração física não é implementada, as operações dos equipamentos são coordenados superiormente, de forma a obter em termos lógicos um agrupamento que permita um fluxo de materiais semelhante ao das células de fabrico. A coordenação e comando destes equipamentos é efectuado por sistemas computacionais que tomam a designação de controladores de célula e daí deriva que por vezes as redes a este nível de controlo sejam designadas por redes de célula [Rembold 93].

Estas redes situam-se nos níveis intermédios da estrutura hierárquica. Estabelecem com o nível superior uma ligação caracterizada quer no sentido descendente quer no sentido ascendente por um fluxo moderado de informação sem assinaláveis requisitos temporais. No sentido descendente obtém informação relativa ao processo de produção e seu escalonamento, ao passo que no sentido ascendente realimenta esse nível hierárquico com informação do estado da produção.

Com o nível inferior estabelece comunicações susceptíveis de estarem associadas, a funções que podem impor requisitos temporais críticos. No sentido descendentes o fluxo de informação inclui o despacho das ordens de fabrico, descarga de parâmetros e programas para configuração dos equipamentos utilizados na produção, assim como funções de controlo para a coordenação dos equipamentos na célula. No sentido oposto obtém informação utilizada na monitorização e realimentação de estado, necessária à coordenação dos equipamentos no processo produtivo.

- **Redes de Campo**

Este tipo de rede suporta as comunicações dos níveis inferiores da estrutura hierárquica, e é responsável pela interligação dos elementos que têm atribuída a função da execução das operações que compõem o processo de produção. Nestes elementos incluem-se os dispositivos que obtêm informação, ou que actuam no processo, assim como elementos de controlo a eles associados, isto é sensores, actuadores e pequenos controladores. A informação processada neste nível é tipicamente caracterizada pelo pequeno volume de informação de natureza cíclica com elevada frequência e requisitos de tempo-real normalmente críticos.

2.2 Redes de Campo

Na década de 80 a utilização de LAN (*Local Area Network*) na comunicação entre dispositivos de controlo de nível baixo começou a afirmar-se como alternativa às soluções pouco flexíveis e dispendiosas oferecidas pela tecnologia das ligações ponto a ponto. Estas redes configuradas para serem uma solução económica utilizando interfaces de comunicação série incluídas em muitos microprocessadores de baixo custo, foram especificadas para suportar quer tráfego cíclico com informação de estado, quer tráfego acíclico para indicação de ocorrência de eventos, garantindo deste modo limites temporais, e resposta de tempo-real necessária ao suporte de aplicações críticas.

Estas redes denominadas de redes de campo (*fieldbus*) seguem o modelo de referência OSI, estabelecendo uma arquitectura baseada em três camadas [Heffernan97]: Fig(2.2).

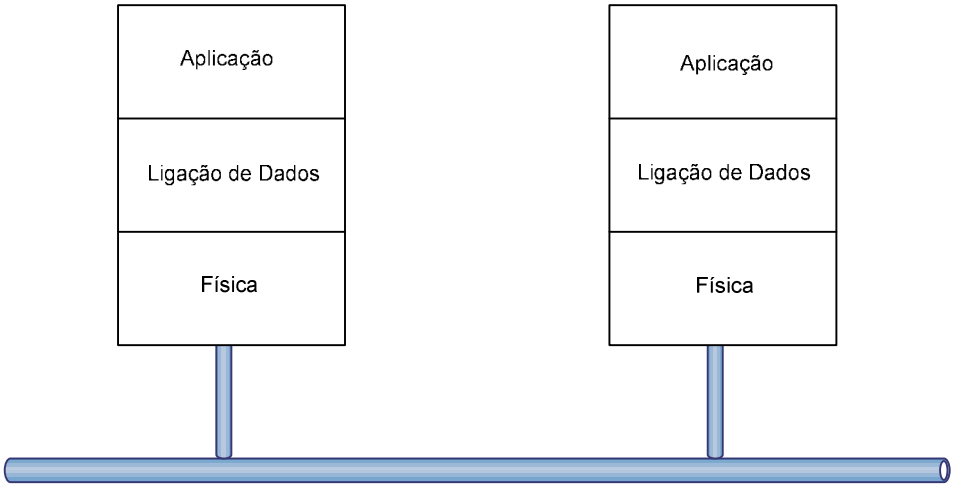

Figura 2.2 - Arquitectura de comunicações das redes de campo

Física: A camada física define a interface entre nós de comunicação, abrangendo os vários aspectos relacionados com o tipo de sinais utilizados na transmissão da informação, aspectos tecnológicos associados ao meio de transmissão e topologia da rede. Nas redes de campo, a simplicidade e a economia da solução são aspectos importantes, pelo que o par entrelaçado e a tipologia em barramento são as soluções mais frequentemente implementadas. O cabo coaxial, a fibra óptica e mais recentemente a transmissão em rádio frequência são outras soluções para o meio de comunicação. A configuração em estrela e anel são também configurações possíveis para topologia de rede.

Ligação de Dados: Na camada de ligação de dados são implementadas funções que permitem assegurar que a comunicação se efectue de forma fiável e isenta de erros. A detecção de erros, assim como os mecanismos de controlo de acesso ao meio, são funções que nas redes de campo são implementadas nesta camada.

Aplicação: Nas redes de campo esta camada integra os serviços de uma camada de utilizador, que inclui um conjunto de objectos do domínio de aplicação como sejam robots, controlos numéricos e controlo de processos. Esta camada estabelece uma interface do utilizador na qual os detalhes do sistema de comunicação, são escondidos transparecendo para este uma imagem virtual dos objectos com que opera.

A utilização das três camadas deriva da não necessidade de implementar a totalidade dos mecanismos do modelo de referência OSI, ou dadas as particularidades da rede, estes são concentrados nas restantes camadas [Thomesse05]. Assim, a camada de rede não está incluída na arquitectura porque, este tipo de rede não necessita de mecanismos específicos para efectuar o encaminhamento do tráfego na rede. Isto fica a dever-se à simplicidade da rede que geralmente estabelece um único caminho entre membros da rede, e mesmo que a sua complexidade aumente o encaminhamento de tráfego é efectuado por nós especializados, *bridges*.

A camada de transporte não está incluída pois muitas das funções como sejam de controlo de erros, segmentação de mensagens e controlo da sua transmissão não assumem requisitos importantes nestas redes ou estão integrados noutras camadas. No caso do controlo de erros, este é implementado na camada de ligação de dados e de aplicação. A segmentação de mensagens não é uma função muito utilizada devido à dimensão reduzida dos pacotes (*Protocol Data Unit* - PDU) neste tipo de rede e, quando em tarefas específicas, como seja a descarga de programa suportadas por alguns perfis de rede, esta função é implementada ao nível da aplicação.

A camada de sessão foi introduzida no modelo OSI para facilitar a troca de mensagens de elevada dimensão, o que não se aplica na maior parte das redes de campo.

A camada de apresentação é de fundamental importância para estabelecer uma sintaxe comum entre estações. Contudo, ao contrário de muitas das redes de comunicação, nas redes de campo os bits que compõem as mensagens não são codificados para obter caracteres, como é caso do ASCII e do EBCDIC, mas estão antes muitas das vezes associados à codificação de estado de sensores e actuadores. No caso das redes de campo, o significado do conteúdo da informação é frequentemente pré-configurado de acordo com os perfis das redes.

2.2.1 Que Rede de Campo?

No suporte de comunicações de tempo-real, a operação das redes de campo deve ter consideração a importância de requisitos como: baixa latência dos protocolos, *jitter* reduzido, capacidade detecção de erros e de capacidade para comportar alterações ao longo do tempo (flexibilidade) [Decotignie05]. Mas, a observação da totalidade destes requisitos apresenta conflitos fundamentais no projecto e implementação dos protocolos que compõem a rede [Kopetz98], ou seja, não é possível satisfazer de igual modo todos estes requisitos.

Neste contexto, é possível uma rede ser concebida para ter bom desempenho no processamento de tráfego cíclico, ou acíclico, mas não é possível garantir os dois simultaneamente sem que o *jitter* do protocolo aumente. Da mesma forma, a opção pelo método de controlo de acesso ao meio tem implicações no desempenho do sistema de comunicações. A utilização de métodos de controlo descentralizados evita situações da existência de um único ponto de falha. Estes são no entanto métodos complexos exigindo um considerável *overhead* em funções de gestão, que se traduz na redução do desempenho das redes de campo. Por outro lado, os métodos de controlo de acesso ao meio centralizados têm vantagens de desempenho, mas a sua falha provoca a interrupção das comunicações de tempo-real até que o sistema recupere desta situação. Analogamente, uma eficiente detecção de erros entra em conflito com a possibilidade do sistema sofrer alterações dinamicamente, obrigando a restrições na sua flexibilidade [Kopetz98] [Pimentel90].

Torna-se então inevitável que o projecto de protocolos de tempo-real tenha por base um compromisso entre diferentes premissas que melhor satisfazem a aplicação a que se destina. Por outro lado, o domínio de aplicação das redes de campo é extremamente abrangente cobrindo um leque bastante heterogéneo de aplicações, como aquelas que é possível encontrar na indústria de manufactura e na indústria de processos. Estes factores propiciam condições para a existência de diferentes abordagens no desenvolvimento de redes de comunicação, o que contribuiu para a proliferação de inúmeras soluções de redes de campo (Fig. 2.3).

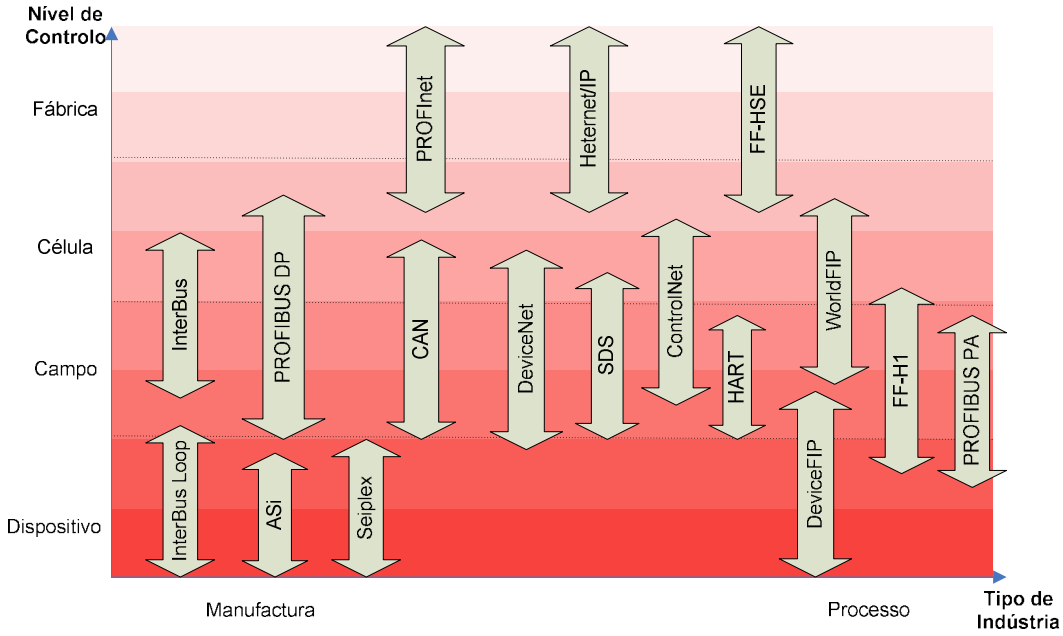

Figura 2.3 - Desenvolvimento das redes de campo.

Muito deste desenvolvimento foi impulsionado por factores de natureza estratégica fora do contexto de qualquer norma. Esta dinâmica demonstra a vitalidade desta área, mas também a importância das redes de campo como elemento estruturante dos sistemas de controlo. Já a não existência de uma norma reguladora fez com que surgisse um cenário pouco claro, com a existência de dezenas sistemas que vieram esbater todas as vantagens associadas aos sistemas abertos que estiveram na base do projecto MAP. As mesmas vantagens que se constituíram como referência de sucesso noutros tipos de rede.

Num esforço de alteração deste panorama teve início um processo de normalização. Esse processo foi encetado com os contributos de vários países em sede de organizações internacionais como a ISA (*Instrumentation Society of America*), CENELEC (Comité Europeu de Normalização Electrotécnica) e IEC (*International Electrotechenical Commission*).

Não obstante a inegável vantagem da existência de uma norma unificadora, o processo de normalização estendeu-se ao longo dos anos, com avanços mais aparentes que reais. Durante esse tempo esgrimiram-se argumentos e sobretudo extremaram-se posições na salvaguarda de interesses, alguns destes associados aos consórcios das soluções já estabelecidas no mercado [Felser02] [Leviti01].

Esta realidade teve como consequência a não obtenção de um consenso e os resultados obtidos pelos diversos grupos constituídos acabaram por ficar muito aquém do desejável.

A nível europeu, na busca de uma norma europeia que permitisse preencher o espaço normativo até à finalização do processo encetado pela IEC, a CENELEC optou pela não definição de uma norma com uma única arquitectura para rede de campo, mas antes pela integração de três standards nacionais. A norma EN 50170 designada por *General purpose field communication system*, inclui a definição e especificação do P-NET da Dinamarca, o World FIP da França e o PROFIBUS (FMS, DP (PA 50020)) da Alemanha [EN96a], [EN96b], [EN99a].

Neste domínio mais três normas foram publicadas pela CENELEC:

- A EN 50254 (*High Efficiency Communication Subsystem for Small Data Packages*) com o propósito de definição e especificação da rede INTERBUS e como adenda à EN 50170, para os perfis de rede PROFIBUS-DP e World FIP [EN98];
- A EN 50325 (*Industrial communication subsystem based on ISO 11898 (CAN) for controller-device interfaces. Smart distributed systems* (*SDS*)), para abranger os perfis DeviceNET, SDS e CANopen, suportados pelo protocolo CAN (*Control Area Network*) [EN02];
- A EN 50295 (*Low-voltage switchgear and controlgear. Controller and device interface systems. Actuator sensor interface* (*AS-i*)), com o propósito de especificar uma norma para redes mais básicas próximas do conceito original da rede de campo, sendo especificado o protocolo AS-i *Actuator and Sensor Interface* [EN99b].

O grupo de trabalho formado pelo IEC, personificou o esforço a nível mundial para obter uma arquitectura de rede globalmente aceite. Contudo, os resultados ficaram muito aquém do pretendido, dando origem à norma IEC61158 (*Digital Data Communications for Meassurement and Control – Fieldbus for use in Industrial Control Systems*) [IEC03a], com 10 redes completamente heterogéneas e incompatíveis entre si, baseadas em muitas soluções já existentes no mercado [Thomesse05].

Em complemento ao IEC61158, foi criada uma norma designada por IEC61784 (*Digital Data Communications for Measurement and Control – Profiles Sets for Continuous and Discrete Manufacturing Relative to Fiedbus use in Control Systems*) [IEC03b], com o propósito de definir perfis de comunicação (*CPF – Communication Profiles Families*), que integrem os diversos tipos de redes de campo especificados no IEC 61158 (Tab-2.2).

IEC 61784 Perfil	IEC 61158 Protocolo – Camadas			Nome Comercial
	Física	Ligação de Dados	Aplicação	
CPF-1/1	Tipo 1	Tipo 1	Tipo 9	Foundation Fieldbus (H1)
CPF-1/2	Ethernet	TCP/UDP/IP	Tipo 5	Foundation Fieldbus (HSE)
CPF-1/3	Tipo 1	Tipo 1	Tipo 9	Foundation Fieldbus (H2)
CPF-2/1	Tipo 2	Tipo 2	Tipo 2	ControlNet
CPF-2/2	Ethernet	TCP/UDP/IP	Tipo2	Ethernet/IP
CPF-3/1	Tipo 3	Tipo 3	Tipo 3	PROFIBUS-DP
CPF-3/2	Tipo 1	Tipo 3	Tipo 3	PROFIBUS PA
CPF-3/3	Ethernet	TCP/UDP/IP	Tipo 10	PROFInet
CPF-4/1	Tipo 4	Tipo 4	Tipo 4	P-Net RS-485
CPF-4/2	Tipo 4	Tipo 4	Tipo 4	P-Net Rs-232
CPF-5/1	Tipo 1	Tipo 7	Tipo 7	World FIP (MPS, MCS)
CPF-5/2	Tipo 1	Tipo 7	Tipo 7	World FIP (MPS, MCS, subMMS)
CPF-5/3	Tipo 1	Tipo 7	Tipo 7	World FIP (MPS)
CPF-6/1	Tipo 8	Tipo 8	Tipo 8	INTERBUS
CPF-6/2	Tipo 8	Tipo 8	Tipo 8	INTERBUS TCP/IP
CPF-6/3	Tipo 8	Tipo 8	Tipo 8	INTERBUS Subset
CPF-7/1	Tipo 6	Tipo 6	-	Swiftnet transport
CPF-7/2	Tipo 6	Tipo 6	-	Swiftnet full stack

Tabela 2.2 - Redes de campo de acordo com IEC 61158 e IEC 61178.

2.3 O PROFIBUS-DP

O PROFIBUS (*PROcess FIeld BUS*) é originário da Alemanha e foi desenvolvido para oferecer uma solução completa de redes de campo, que satisfizesse os diferentes requisitos de comunicações a nível industrial. Esta oferta era baseada em três variantes:

- **PROFIBUS-FMS** (*Fieldbus Message Specification)*: foi a primeira versão do PROFIBUS. Foi especificada para suportar uma diversidade serviços de comunicações que satisfizessem as necessidades de comunicação entre dispositivos inteligentes.

- **PROFIBUS-DP** (*Decentralized Periphery*): desenvolvida para aplicação em sistemas de produção e automação, está optimizado para comunicações rápidas de pequenas mensagens entre dispositivos de entradas e saídas.
- **PROFIBUS-PA** *(Process Automation)*: desenvolvida para aplicação na indústria de processos, capaz de suportar aplicações intrinsecamente seguras e de fornecer alimentação eléctrica aos dispositivos a ela ligados.

Em consequência da introdução de extensões à versão original do PROFIBUS-DP que lhe permitem, nomeadamente, suportar tráfego acíclico, actualmente o PROFIBUS FMS já não desempenha um papel relevante, não fazendo parte da norma IEC61158. Desta forma, o PROFIBUS-DP assume um papel importante na estrutura das comunicações industriais, assegurando as comunicações ao nível da célula e de campo. A PROFIBUS fornece uma solução de comunicações estruturada e integrada, acedendo ao nível inferior *sensor bus* através de interface ASi e com o nível superior através de redes *ethernet* suportadas pelo PROFInet (Fig. 2.3).

Uma rede PROFIBUS-DP é composta por três tipos de nós de comunicação: mestre (*master*) classe 1, mestre classe 2 e escravos (*slaves*). Os nós de comunicação são também designados em função da sua actividade na rede. Deste modo, os mestres, em consequência de serem as estações que por sua iniciativa produzem actividade na rede, são também designados por estações activas. Em contraste os escravos só têm actividade na rede a pedido dos mestres pelo que são também designadas por estações passivas. A comunicação de dados é efectuada essencialmente entre mestres classe 1 e escravos, tendo os mestres classe 2 uma função essencialmente de manutenção, configuração e gestão da rede, não sendo portanto necessário estarem permanentemente na rede [Jecht05].

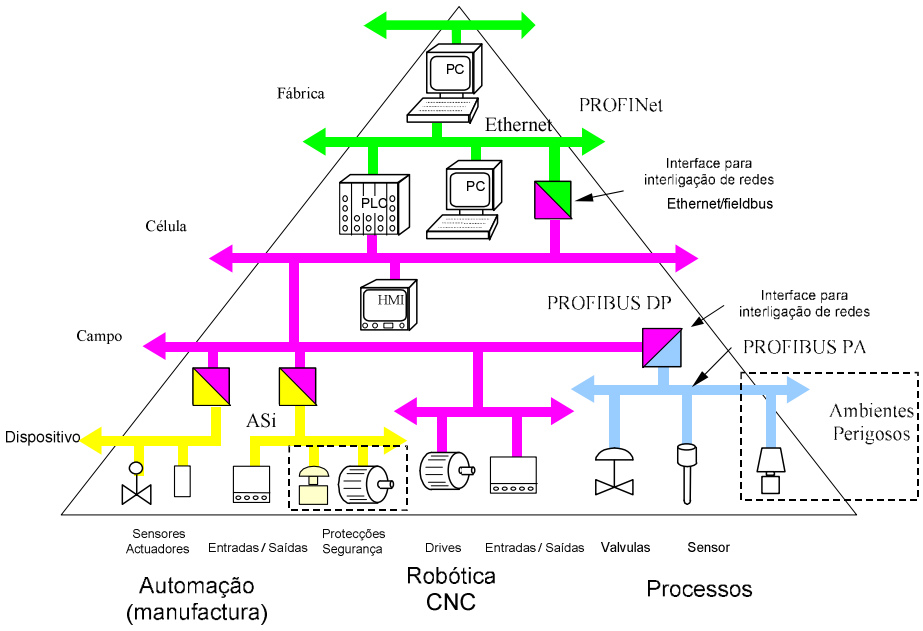

Figura 2.3 - Solução integrada de redes em ambiente industrial baseada na utilização da rede PROFIBUS-DP.

2.3.1 Arquitectura

Do ponto de vista da arquitectura do protocolo, o PROFIBUS-DP pode ser considerado como a aplicação da camada 2 de um protocolo baseado em duas camadas [Popp03]. No topo desta estrutura é fornecido um conjunto de perfis de forma a satisfazer requisitos específicos do domínio de aplicação (Fig. 2.4)

Na camada física, o PROFIBUS-DP pode fazer uso de diferentes tipos de tecnologia, onde se inclui: o RS-485, RS-485-IS intrinsecamente segura, a MBP-IS (*Manchester coded Bus Powered*) e fibra óptica [Jecht05].

A configuração mais económica, mais simples e mais difundida consiste na transmissão de dados através de interface RS-485, fazendo uso de cabo entrelaçado com blindagem. A rede tem uma topologia em barramento capaz de suportar até 126 nós de comunicação, divididos por segmentos de 32, sendo a ligação entre segmentos assegurada através de repetidores. Neste tipo de configuração os dados são codificados de acordo com o código NRZ (*Non Return to Zero*), ou seja o nível eléctrico associado ao correspondente valor lógico não é alterado durante a duração do bit. A taxa de transmissão pode ser estabelecida em valores da gama de 9.6Kbit/s a 12Mbit/s, sendo este valor condicionado designadamente, pelo comprimento dos segmentos que podem atingir 1200m.

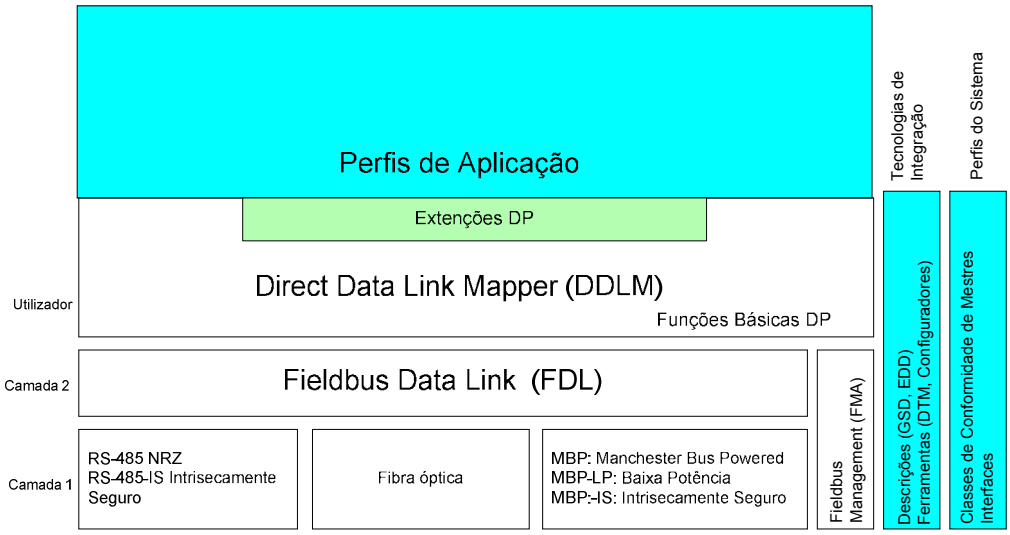

Figura 2.4 - Arquitectura do PROFIBUS-DP.

A camada de ligação de dados é designada por *Fielbus Data Link Layer* (FDL) e é responsável no processo de comunicação por assegurar ligações fiáveis. Para estabelecer as comunicações o PROFIBUS-DP utiliza um conjunto de tramas denominadas por telegramas que são suportadas em dois serviços: o SRD (*Send and Request Data with acknowledge*) e o SDN –(*Send Data with No acknowledge*).

Essas tramas englobam telegramas destinados à troca de dados propriamente dita e telegramas que estão associados a funções de gestão da rede – (*Token* e *Request FDL Status*). Nestas últimas assenta parte significativa do controlo de acesso ao meio implementado pelo MAC – *Medium Access Control*. No PROFIBUS-DP, este segue uma estratégia híbrida com o objectivo conciliar as vantagens ao nível do desempenho que advêm da centralização do controlo, e simultaneamente reduzir a possibilidade de um único ponto de falha, complementando-o com um mecanismo simples de descentralização. Numa primeira fase este método consiste na formação de um anel lógico. Neste as estações activas (mestres) estão inseridas partilhando o acesso ao meio através da utilização de um testemunho (*token*), que é passado entre estações de acordo com uma sequência ascendente de endereço na rede. Numa segunda fase, de acordo com o modelo mestre escravo, a estação que possui o *token* tem a opção de encetar transferência de dados com as suas estações passivas (escravos) durante um tempo especificado *Token Holding Time* (T_{TH}) (Fig. 2.5). Este tempo resulta da diferença do *Target Rotation Time* (T_{TR}), parâmetro de configuração da rede e do tempo total gasto na rotação do token pelo anel lógico *Real Rotation Time* (T_{RR}) medido na estação em causa.

Novas estações são admitidas no anel quando assinalam essa intenção na resposta a tramas *Request FDL Status*. Estas tramas são enviadas periodicamente por cada estação no espaço de endereçamento compreendido entre a estação que detém o *token* e a próxima estação no anel (GAP List). O período de tempo que rege a geração deste tipo de trama é especificado pelo parâmetro T_{GUD} que é um múltiplo de T_{TR}.

Para além de garantirem o controlo de acesso ao meio, as tramas de token têm uma outra função importante pois identificam as estações no anel, pelo que são permanentemente escutadas por todas as estações. Desta forma, é lhe possível efectuar e manter actualizado o mapeamento das estações no anel, e representar a estrutura da rede através de uma lista das estações activas (LAS - *List of Active Stations*).

Durante a fase de transferência de dados, o tipo de tráfego estabelecido entre mestre e escravo varia de acordo com a especificação da camada do utilizador. Inicialmente o PROFIBUS-DP (*DP-V0*) foi especificado essencialmente para suportar tráfego cíclico. Este consiste essencialmente na troca de dados e obtenção de diagnósticos. De forma a aumentar o campo de aplicação da rede, extensões ao protocolo foram sendo introduzidas, comportando actualmente, para além da base *DP-V0*, mais duas versões: a *DP-V1* e *DP-V2*.

A versão *DP-V1* acrescenta ao PROFIBUS-DP importantes funcionalidades para uso em aplicações de automatização de processos que englobam a utilização de tráfego acíclico em tarefas de monitorização, operação e gestão de alarmes.

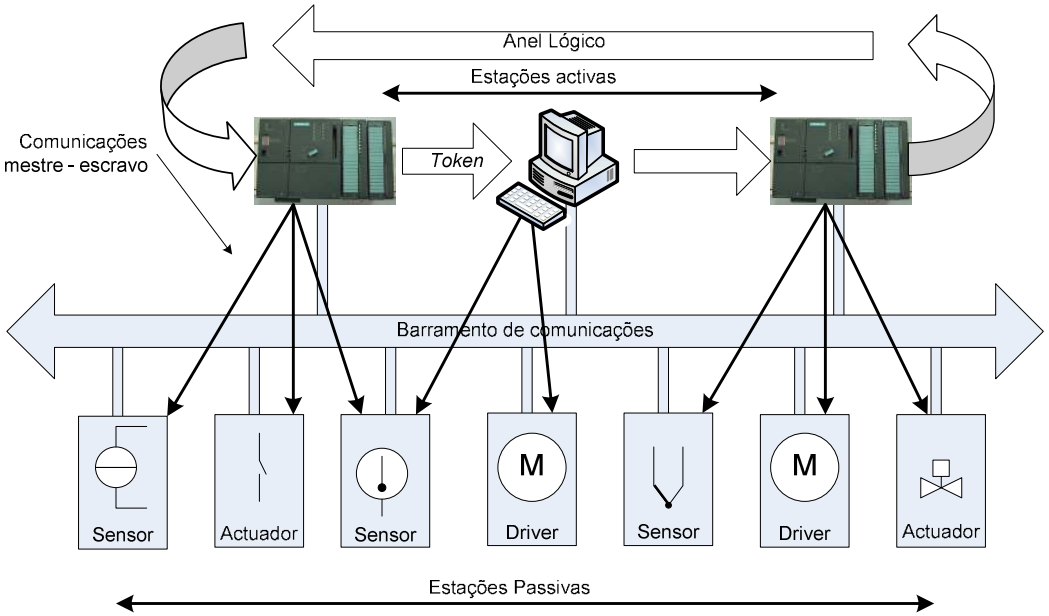

Figura 2.5 - Controlo de acesso ao meio numa rede PROFIBUS-DP.

A versão *DP-V2* é uma extensão à norma IEC61158 com a função de melhorar e adaptar o PROFIBUS-DP à tecnologia de controlo de servomecanismos [Popp03]. Esta versão introduz novas funcionalidades no funcionamento do protocolo nomeadamente:

- O modo *isochronous*: acrescenta determinismo ao protocolo permitindo aos mestres que implementam esta função manterem-se sincronizados durante a execução de ciclos de controlo, independentemente da carga no barramento*;*
- A difusão de dados (*Data Exchange Broadcast*): esta funcionalidade permite a transferência de dados entre escravos. Os ciclos de transferência de dados continuam a ser ditados pelos mestres. Contudo, a introdução do conceito de produtor (*Publisher*) e subscritor (*subscriber*) permite que escravos publiquem os seus dados e outros que os subscrevam, os recebam por um processo de difusão (*Broadcast*). A troca de informação é desencadeada nos ciclos de transferência de dados entre mestre e escravo produtor;
- A sincronização temporal: estabelece um serviço de relógio global que permite a sincronização de todas as estações na rede. Esta função é sobretudo útil em funções como a aquisição de eventos que necessitem de determinação da sequência temporal;
- A carga e descarga: permite funções de gestão da rede como carga e descarga de quantidades elevadas de informação sem necessidade de intervenção manual. A descarga de programas nos equipamentos é um exemplo da aplicação desta função;
- A redundância: suporte do conceito de redundância para aplicações com requisitos de segurança muito elevados.

Em síntese o PROFIBUS-DP apresenta três versões, em que duas surgem como fazendo parte do processo de evolução do PROFIBUS-DP, e que utilizam os serviços fundamentais da versão *DP-V0* (tráfego cíclico). Acresce-se que muitas das funções introduzidas pelas versões mais recentes estão associadas a tráfego acíclico não prioritário, ao contrário das funções com requisitos temporais críticos, que geralmente são suportados pelo tráfego cíclico [Popp03]. Neste contexto, esta dissertação aborda unicamente a componente cíclica do tráfego do PROFIBUS (*DP-V0*).

2.3.2 O PROFIsafe

Em 1998 foi iniciado um projecto com o objectivo de assegurar comunicações seguras através do PROFIBUS-DP. Este projecto assentou no pressuposto da possibilidade das comunicações serem efectuadas em redes onde coexistissem dispositivos normais e dispositivos seguros, sem haver necessidade de qualquer alteração ao nível de cabos, ASICs, *stack* de comunicação ou de qualquer outro dispositivo da rede [Stripf05].

O PROFIsafe é o resultado desse projecto e do ponto de vista da arquitectura do protocolo aparece como um perfil de aplicação onde os níveis inferiores são mantidos inalterados, sendo o seu comportamento tratado como uma caixa negra (*black box*). Desta forma, neste nível de aplicação são adicionados mecanismos que permitem tornar o protocolo independente da detecção de erro implementada nas camadas inferiores, sendo assim possível conferir níveis de funcionamento seguro até SIL3. Para completar esta abordagem, a estrutura das tramas de comunicações são mantidas, sendo os campos de controlo que suportam os mecanismos do PROFIsafe integrados no bloco de dados das tramas standard do PROFIBUS-DP. A estrutura das tramas do PROFIBUS-DP será alvo de análise no capítulo 5.

Em sistemas complexos como são os sistemas de comunicações e de que a rede PROFIBUS-DP é um exemplo particular, existe uma diversidade de factores que podem levar à ocorrência de erros e a estados de operação incorrectos. Nestes incluem-se factores externos como sejam a ocorrência de interferências electromagnéticas, factores de natureza física que levem a falhas do *hardware*, ou mesmo de deficiente operação dos equipamentos, nomeadamente pela sua incorrecta configuração. As consequências deste tipo de eventos manifestam-se tipicamente através erros de transmissão designadamente através de: perdas de mensagens; mensagens repetidas, inserção de mensagens, falha na sua sequência, corrupção de dados e atrasos nas comunicações. De forma a tratar estas situações, o PROFIsafe implementa um conjunto de medidas de recuperação (Tab. 2.3) [Stripf05]:

	Medidas			
Erros de transmissão	Sinal de validade	Time-out com confirmação	Código na comunicação entre emissor e recepto	Teste integridade de dados
Mensagem repetida	√		–	–
Perda de mensagem	√	√	–	–
Mensagem inserida	√	√	√	–
Sequência incorrecta	√	–	–	√
Corrupção de dados	–	–	–	–
Atrasos	–	√	√	√
Masquerade (tramas standards imitam *failsafe*)	–	√	–	–

Tabela 2.3 - Erros de transmissão e medidas de recuperação implementados pelo PROFIsafe.

- **Envio de sinais de validade (*sing-of-life*)**: manter a sequência de eventos e ordens de comando é de grande importância para o correcto funcionamento dos sistemas de controlo. Para garantir a sequência das tramas nas comunicações o PROFIsafe implementa um mecanismo que consiste no envio de um número em cada trama. Este número vai sendo incrementado, permitindo ao receptor detectar se recebeu todas as tramas e se estas se encontram na sequência correcta;
- **Time-out**: nos sistemas de tempo-real não interessa unicamente que os dados sejam recebidos sem erros. É necessário também que os mesmos cheguem num tempo limite que evite que o sistema falhe. Para evitar consequências graves resultantes da chegada para além do tempo de segurança é permitido aos receptores (escravos) encetarem medidas de segurança, como seja a paragem da operação e a transição para um estado de operação seguro. Estas medidas são comandadas com base em temporizadores (*watch-dog timer*) que monitorizam os tempos de segurança;
- **Utilização de códigos na comunicação entre emissor e receptor**: nas comunicações com requisitos de segurança é importante assegurar que a comunicação é estabelecida entre membros bem conhecidos. Para evitar que a relação 1:1 estabelecida na comunicação mestre-escravo seja quebrada, o PROFIsafe utiliza códigos (*passwords*) que permite verificar a autenticidade da trama;
- **Teste da integridade de dados**: um dos aspectos que assume grande importância nas comunicações é a verificação da integridade de dados. Um dos mecanismos mais utilizados para a função é a utilização de códigos de redundância cíclica (CRC - *Cyclic Redundancy Check*), que são adicionados à trama, e que resultam da aplicação de um polinómio ao conteúdo da mesma. Desta forma, aplicando a mesma operação matemática no receptor, é possível identificar alterações nos dados das tramas. O PROFIsafe implementa polinómios de CRC totalmente

independentes dos mecanismos de detecção de erros FCS (*Frame Check Sequence*), e testes de paridade usados no PROFIBUS-DP. Com esta configuração é possível assegurar uma cobertura até uma probabilidade de 10^{-9} falhas/hora. Para complementar este sistema de teste à integridade de dados, o PROFIsafe implementa o mecanismo designado por *SIL monitor* que examina a operação em modo seguro do sistema de comunicações**.** A sua utilização permite garantir níveis de integridade até SIL3 [Stripf05] [Popp03].

Em síntese, o PROFIsafe implementa uma estratégia de caixa negra "*Black Channel*", que lhe permite abstrair da tecnologia de comunicações utilizada e de detalhes de implementação dos protocolos usados nas camadas que lhe estão abaixo. Contudo, dado que o seu funcionamento assenta em serviços do PROFIBUS-DP, herda deste as características de desempenho, nomeadamente no que respeita ao comportamento temporal.

2.4 Análise Temporal do PROFIBUS-DP: Trabalho Relevante

Genericamente os sistemas de comunicação que suportam aplicações distribuídas de tempo-real têm como função assegurar a comunicação entre elementos do sistema, garantindo as *deadlines* das mensagens envolvidas no processo. A prossecução deste objectivo requer que o acesso dos nós de comunicação ao meio, assim como a correspondente troca de mensagens deve ocorrer em intervalos de tempos compatíveis com as especificidades temporais do sistema, e serem limitados superiormente no tempo [Joseph03] [Zhang99] [Agrawal94] [Chen92]. Este modo de operação deve ser garantido independentemente da ocorrência de condições que possam levar a alterações dos estados carga do sistema. Assim, a caracterização do controlo de acesso ao meio e o estudo das suas propriedades são de particular importância na análise das comunicações de tempo-real.

Nas redes de campo não existe um método de controlo de acesso ao meio que seja comum para todas as redes. Existe antes uma diversidade que vai de encontro às especificidades de cada rede. Para controlar o acesso ao meio, o PROFIBUS-DP implementa a uma versão simplificada da norma IEEE 802.4. Este método é atractivo dada a sua fiabilidade e o determinismo que confere no tempo de acesso ao meio. No entanto, ao contrário da norma IEEE 802.4, o método utilizado pelo PROFIBUS-DP não reserva largura de banda, ou seja no caso da recepção de *token* atrasado, só é garantida a transmissão de uma mensagem de alta prioridade em vez poder transferir o seu tráfego prioritário durante um tempo pré-configurado. A configuração do protocolo deve ser objecto de parametrização cuidada para que este se adapte a aplicações com requisitos temporais críticos [Cavalieri95] [Tovar98a] [Tovar99a].

O PROFIBUS-DP é uma rede direccionada para o *polling* cíclico de dispositivos de instrumentação e de controlo, pelo que é necessário assegurar a coerência temporal nestas acções. Assim, indicadores de desempenho como tempo de ciclo (tempo entre a recepção de dois *tokens* consecutivos) e a sua variação (*jitter*), são de grande importância na análise desta rede. Resulta assim, que muito do trabalho nesta área está orientado para o problema do escalonamento e condições de escalonabilidade para suporte de comunicações de tempo-real [Tovar99b] [Tovar99c] [Cavalieri02] [Wang03] [Vitturi04].

A caracterização do tempo de ciclo é alvo de vários estudos. Em [Tovar99d] é definido um limite superior para o tempo entre a recepção de dois *token* consecutivos numa dada estação (tempo de ciclo). Este limite considera um cenário onde coexiste tráfego cíclico e acíclico. Foi formulado com base num modelo de mensagens onde foram incluídas as componentes temporais associadas aos ciclos de *polling* e a possíveis retransmissões de mensagens, assim como, das mensagens de gestão do anel lógico. Com base neste limite superior foi formulada uma metodologia para calcular o parâmetro T_{TR} que pretende assegurar as condições de escalonamento de aplicações de tempo-real [Tovar98b].

Em [Cavalieri02] o trabalho anterior é expandido para suportar configurações de rede com um único mestre (mono-mestre). Nesta configuração não existe partilha do meio com outras estações. Assim, a generalização dos resultados obtidos para mais que um mestre implica assumir limites para o pior caso de tempo de ciclo não muito enquadráveis com a configuração de rede. No seguimento da análise mono-mestre é formulada uma nova proposta para o tempo de resposta no pior caso (*worst-case response time*), em configuração multi-mestre com intervalos de tempo mais apertados. A obtenção de limites mais apertados para o tempo de ciclo tem como primeira vantagem o aumento da probabilidade do cumprimento das *deadlines*, e também aumentar o espectro de aplicações escalonáveis num processo de escalonamento pré-execução (*pre-runtime*). Desta forma vários trabalhos foram sendo efectuados com este mesmo objectivo [Cavalieri02] [Monforte00].

O tráfego acíclico contribui também para a introdução de *jitter* nas comunicações. Uma abordagem à analise do efeito deste tipo de tráfego no tempo de ciclo é efectuada em [Vitturi04]. Esta análise considera a contribuição das diferentes componentes de tráfego acíclico, como sejam o resultado de tráfego gerado ao nível da FDL para tarefas de gestão do anel lógico, ou gerado nos níveis superiores nomeadamente em resposta a comandos da camada do utilizador. Como resultado da análise é apresentada uma configuração de um parâmetro do protocolo para que a componente do *jitter* introduzido pelo tráfego gerado ao nível da FDL seja eliminada. O processo consiste na configuração do parâmetro que regula a periodicidade das mensagens de gestão do anel (*GAP Update Time* T_{GUD}) para que estas não sejam geradas em múltiplos do ciclo do *token*, mas antes que se produzam em cada ciclo. Apesar de o parâmetro T_{GUD} no modo de operação multi-mestre influenciar o comportamento do *jitter*, não é possível remover a sua componente, nem tão pouco alterar a sua amplitude. De

facto não, é possível gerar indefinidamente tramas do tipo pretendido. De acordo com a norma da rede, ao fim de completar a verificação da GAP *List*, a estação terá que aguardar pelo menos um período igual a T_{TR} para poder iniciar de novo esta função. Assim, a utilização deste parâmetro unicamente modifica a frequência dos ciclos de actualização da GAP *List*. No caso de operação da rede no modo multi-mestre, os vários contributos das estações para este tráfego irão produzir um efeito de modulação que confere a esta componente características que são de difícil caracterização.

A um nível mais global e com o objectivo de reduzir o efeito das actividades acíclicas do PROFIBUS-DP, no tempo de ciclo é proposto um método para optimizar a operação da rede. Este consiste na reserva de tempo suficiente para as actividades acíclicas, e no caso deste tempo não ser utilizado na totalidade, o tempo de ciclo é prolongado artificialmente, para que este se mantenha constante [Vitturi04]. Este método contudo, tem o grande inconveniente de não se basear na alteração da parametrização do protocolo, mas antes requerer alteração da sua estrutura.

Uma particularidade das abordagens à análise temporal da rede PROFIBUS-DP que foram apresentadas, reside no facto de estas se basearem essencialmente nas descrições do funcionamento que constam na norma da rede. Por outro lado a necessidade de repetições provocadas por erros de comunicações não é prevista na sua totalidade. Na maior parte dos casos é unicamente considerada a possibilidade de repetições de mensagens que intervêm nos ciclos de *polling*, não sendo tida em consideração a necessidade de repetição de mensagens de *token* geradas ao nível da FDL. Mais, não são considerados outros atrasos que derivam de uma forma genérica da activação de mecanismos de recuperação de erros, que são implementados nos protocolos de comunicação. Em suma, estas análises estão mais de acordo com o desempenho do protocolo, o que de certa forma é válido na maior parte do tempo de operação da rede. No entanto, não são representativas de situações que derivam da ocorrência de perturbações que provoquem erros nas comunicações, ou de situações de picos de cargas que estão muitas das vezes associados a estes eventos.

2.4.1 Operação em Modo Degradado

A ocorrência de erros nas comunicações obriga à tomada de medidas ao nível dos protocolos de comunicações de forma a recuperar desse estado de operação. Como consequência é desencadeado um conjunto de acções que têm como efeito colateral a introdução de atrasos e sobrecargas (*overhead*) no sistema de comunicações. Entre as acções desencadeadas encontram-se nomeadamente: tempos de espera, envio de mensagens de manutenção e se necessário a repetição das mensagens afectadas. Estes eventos levam inevitavelmente a atrasos no envio de mensagens de dados no sistema de comunicações. Quando isto ocorre em redes que estabelecem comunicações em níveis superiores, os efeitos de tais acções são relativamente bem tolerados, mesmo que os tempos de recuperação resultem elevados. Contudo, nos níveis inferiores em que existe restrições de

tempo-real apertadas, os tempos de atraso produzem uma degradação do modo de operação que nem sempre pode ser tolerada. Isto releva a necessidade de analisar a capacidade do sistema para operar na presença de faltas [Mayer95] [Mayer89].

Neste contexto reveste-se de grande importância a caracterização do funcionamento da rede, identificando eventos típicos que estão relacionados com a ocorrência de erros, sua probabilidade e impacto no desempenho, nomeadamente em indicadores que estão associados aos tempos de acesso ao meio [Moon98].

Apesar da importância da análise de desempenho na presença de faltas para uma correcta percepção da operação das redes de comunicação, a maior parte das análises baseia-se unicamente em indicadores de desempenho.

Ao nível das redes de campo com domínio de aplicação na automação e no controlo de processos, o PROFIBUS-DP é uma das redes mais difundidas a nível mundial. Porém, existem poucos trabalhos em que seja feita uma abordagem à operação da rede na presença de erros, quer seja efectuada numa perspectiva da confiança do funcionamento, quer seja numa perspectiva do seu desempenho na presença de faltas. Isto, não obstante o ambiente em que opera ser considerado severo, nomeadamente pela existência de uma grande diversidade de fontes geradoras de ruído electromagnético, que aumentam de forma considerável a susceptibilidade de interferências nas comunicações. Os trabalhos a seguir referidos focam aspectos da operação das redes que não podem ser dissociados do ambiente que as envolve.

Como em qualquer outra rede de comunicações, o acesso das estações ao meio físico é condicionado pelas regras do protocolo. Desta forma, poderão ocorrer cenários nos quais as estações possam estar mais tempo inibidas de aceder ao meio do que aquele que ocorre em regime de funcionamento estacionário. Uma análise do pior caso para este tipo de impedimento na rede PROFIBUS-DP é efectuada em [Veríssimo97]. Este tempo designado pelos autores de inacessibilidade resulta de três tipos de eventos: inserção de novas estações, perdas de *token* e falhas de estação.

- **Inserção de novas estações**: a componente de inacessibilidade deriva da inserção de novas estações e representa o tempo de espera que uma ou mais estações, que pretendem ser admitidas no anel lógico, têm de esperar. De acordo com a norma da rede [EN96a] as estações são inseridas no anel lógico quando expressam essa intenção em resposta a uma mensagem de gestão do anel (*Request FDL status*). Com base nesta regra é identificado o tempo de espera para uma estação sendo depois generalizado para o caso da existência de mais que uma estação em espera.
- **Perda de *token***: a perda de *token* numa rede como a PROFIBUS-DP impossibilita o acesso das estações ao meio, até que de novo uma estação fique na posse do *token*. Numa situação de perda de *token* a rede recupera gerando um novo *token*, processo este que é desencadeado pela expiração de um temporizador (*timeout*), que

ocorre na estação com menor endereço. Para este caso é identificado como pior caso de inacessibilidade o tempo em que a estação com endereço mais elevado fica inibida de aceder ao meio.

- **Falha de estação**: neste caso a estação que sofre a incidência abandona o anel. De acordo com as regras do protocolo a tentativa de passagem de *token* é repetida duas vezes, após as quais o processo de passagem do *token* é efectuado para a estação imediatamente a seguir no anel. Com base nesta regra é identificado o tempo dispendido na tentativa de passagem do *token*, sendo este depois generalizado para um cenário de falha simultânea de mais do que uma estação.

As expressões analíticas obtidas para cada uma destas situações, resultam da aplicação das componentes temporais e comportamentais (regras) especificadas na norma do protocolo, e podem representar quer casos de operação normal da rede, como sejam inserção e remoção de estações, quer situações que decorrem da necessidade de recuperação de erros. Apesar destas expressões representarem indicadores que podem ser associados a uma análise de desempenho na presença de faltas, elas não resultam de uma efectiva análise da operação da rede nessas condições. Isto é consubstanciado no facto de os resultados serem derivados sem entrarem em consideração com a existência dos erros. O processo poderia ser efectuado quer pelo recurso a um modelo de faltas representativo dos cenários que levam à ocorrência dos eventos indicados, quer pela obtenção experimental de parâmetros que permitam incluir a influência dos erros nessa análise.

Uma grande parte dos sistemas podem apresentar operação em modo degradado. Em [Bello99] é efectuada uma análise ao funcionamento do PROFIBUS-DP, tratando-o como um sistema capaz de suportar operação em modo degradado. Nesta análise a operação da rede é simulada, tendo como suporte um modelo implementado em redes de petri estocásticas. O modelo inclui informação das faltas, sua frequência e impacto na operação dos três blocos que constituem o sistema: nós de comunicação passivos (escravos), nós de comunicação activos (mestres) e barramento de comunicação.

As faltas são especificadas quer para cenários em que a sua influência no sistema é transitória, quer em cenários que devido à persistência dos seus efeitos é necessária a intervenção do operador, e representam os seguintes casos de operação:

- **Falhas nos *transceivers***, que podem ser resultantes de avaria, mau funcionamento momentâneo que leve ao corte do canal de comunicação, ou à corrupção do conteúdo da informação transmitida/recebida;

- **Falhas no barramento**, que podem ser resultantes de interferências no barramento, ou de uma anomalia na estrutura do meio físico que impossibilite as comunicações. O comportamento do sistema na presença de falhas que não sejam permanentes é modelado através de retransmissões, quando os erros ocorrem em mensagens de dados, e saídas do anel lógico quando os erros ocorrem na transmissão do *token*.

Com base neste modelo foi analisada a tolerância a faltas do PROFIBUS-DP assim como foram obtidas medidas de confiança no funcionamento da rede, nomeadamente a disponibilidade e intervalos de tempo médio entre falhas consecutivas (*Mean Time Between Failures* - MTBT).

Um aspecto de crucial importância numa análise deste tipo prende-se com o grau de fidelidade com que o modelo representa a operação do sistema para as condições de análise. Neste particular, a parte nuclear do modelo que engloba a activação de mecanismos de recuperação de erros e consequente comportamento do protocolo, foi efectuada tendo como referência a norma do PROFIBUS-DP. Contudo dada a complexidade do sistema muitos dos efeitos provocados pelas faltas podem não ser facilmente identificados, ou identificados de forma imprecisa, quando se recorre simplesmente a este processo.

Um outro aspecto importante está relacionado com a certificação dos resultados fornecidos pelo modelo. Para se obterem resultados que permitam tirar conclusões em relação ao sistema em análise é necessário que estes sejam validados, ou que sejam obtidos por processos em que o grau da incerteza da sua validade seja reduzido. Neste trabalho não está explicito a que tipo de validação o modelo foi sujeito. Desta forma os resultados obtidos podem estar fortemente afectados por se basearem num modelo que hipoteticamente pode não representar fielmente a real operação do sistema na presença de erros.

Neste contexto na concepção de modelos de sistemas de elevada complexidade, é de grande utilidade o recurso a ferramentas que permitam a identificação do real comportamento do sistema. É igualmente importante proceder à identificação de forma precisa de quais os eventos associados à ocorrência de faltas, qual a configuração de faltas que está associada a esses eventos e sua probabilidade para um determinado cenário de faltas.

No PROFIBUS-DP coexistem com as tramas de dados, tramas que têm como única função a gestão do funcionamento da rede. Estas tramas são geradas ao nível da FDL sem qualquer intervenção do utilizador, e delas depende a correcta organização do anel lógico, assim como uma parte significativa do controlo de acesso ao meio.

Num contexto de operação em ambientes agressivos, do ponto de vista das interferências electromagnéticas, torna-se importante identificar qual o nível de perturbação no desempenho global do sistema provocado por erros que afectem as tramas de gestão da rede. Um trabalho que aborda este problema com grande profundidade é apresentado em [Willig99a], [Willig99b], [Willig99c], [Willig01], [Willig02]. Este trabalho enquadra-se numa perspectiva da análise da adequação do PROFIBUS-DP para operar tendo como meio físico a transmissão em radiofrequência. Concretamente na adequação do seu MAC, que se baseia numa versão simplificada do *Token Passing protocol*, a operar sobre meios que exibem elevadas taxas de erros.

Assim, um dos objectivos do trabalho consistiu na caracterização da estabilidade do anel lógico do PROFIBUS-DP e das principais causas que lhe provocam instabilidade. A análise foi suportada por duas ferramentas:

- Um modelo analítico baseado e cadeias de Markov [Willig99c];
- Uma ferramenta de simulação que permitiu avaliar a operação da rede, quer através da identificação de eventos geradores de instabilidade no anel, quer para obtenção de medidas de desempenho do protocolo.

A ferramenta de simulação implementa parte considerável da FDL, nomeadamente do MAC do PROFIBUS-DP, fazendo uso do CSIM, um software comercial desenvolvido para modelar e simular a operação de sistemas discretos [CSIM].

O modelo analítico foi validado pelo recurso à informação obtida pela ferramenta de simulação, ao passo que esta foi validada por inspecção.

O funcionamento da rede foi simulado para um cenário onde somente tramas de gestão eram transmitidas na rede, e submetido à incidência de erros de acordo com um modelo que emulasse os erros que ocorrem num canal de transmissão de radiofrequência como os das redes locais sem fios.

Dos resultados das simulações foram identificados eventos que provocavam a saída de estações do anel lógico. Esses eventos estão associados às três principais causas que o autor identifica como geradoras de instabilidade do anel, às quais foi atribuída a seguinte designação [Willig01] [Willig99b] [Willig99c]:

- ***Error skipping***, processo pelo qual uma estação é removida do anel. Este comportamento é provocado por erros que não são detectados pelo mecanismo de detecção de erros implementado na trama do *token* (*parity*), e quando, em resultado desse erro, o seu endereço no *token* é modificado para um de uma estação que se encontra acima de si no anel;
- ***Hearback removal***: processo pelo qual a estação detentora do *token* termina a sua operação no anel e abandona-o em resultado da detecção de erros em duas tentativas consecutivas de passagem do *token*. Como resultado, a estação descarta o *token* e perde todo o conhecimento da estrutura do anel lógico (LAS);
- ***Hearback removal with ring jacking***: trata-se de um processo idêntico ao anterior só que ocorre na estação com o menor endereço. Neste caso o processo de recuperação (geração de um novo *token*) é desencadeado na estação afectada. Como esta não tem conhecimento da estrutura do anel lógico, reinicia o anel provocando o colapso deste com a consequente remoção de todas a estações.

Não obstante, a existência de eventos que causam instabilidade no anel, os eventos reportados não estão de acordo com a norma da rede. Por outro lado não representam com fidelidade o que de facto ocorre quando a operação da rede é perturbada por erros, nomeadamente no que se refere às tramas de gestão.

O *Error Skiping* é um evento que na realidade não acontece como é descrito em [Willig01] [Willig99b] [Willig99c]. De facto existe a possibilidade de ocorrência deste erro, mas para dois cenários distintos daquele que é apresentado.

Um com uma probabilidade reduzida, e outro resultante da combinação de vários factores em que intervêm os mecanismos do PROFIBUS-DP, a estrutura física dos *transceiver* e do barramento de comunicações. Uma explicação detalhada e fundamentada será efectuada no capítulo 5 da dissertação.

De igual modo o evento *Hearback Removal*, não corresponde ao que de facto ocorre na realidade. A detecção de dois erros consecutivos leva a estação a suspender a sua actividade e a descartar o *token*, saindo do anel mas evoluindo para um estado de monitorização da rede sem perda do conhecimento da sua estrutura (LAS). De acordo com a norma, a situação descrita só ocorre quando o mecanismo de detecção de erros permanentes nos *transceivers* é disparado. Neste caso, basta que na análise do *token*, este erro se verifique uma única vez. Na prática este evento pode também ocorrer para uma combinação específica de erros no *token*, como é o caso implementado em ASIC certificado pela PROFIBUS-DP. Acresce que a ocorrência de um *Hearback Removal* como ele é descrito faz evoluir a estação para um estado de não operacionalidade que deve ser gerida com a intervenção do operador. Assim, o evento *Ring Jacking* produz-se essencialmente porque na simulação a estação é lançada na rede logo após a ocorrência deste evento, o que contraria o modo correcto de operação da rede. No entanto, o evento *Ring Jacking* pode ocorrer, mas com uma probabilidade muito reduzida como será descrito no capítulo 5.

Não obstante estas imprecisões, este trabalho tem a grande virtude de abordar o modo de operação da rede, quando ocorrem erros que afectam as tramas de gestão. Erros esses que têm um impacto no desempenho da rede que não pode deixar de ser considerado.

2.5 Definição do Problema

Na conexão de sistemas de controlo, as redes de campo assumiram um papel de destaque. A sua aceitação tem gerado uma dinâmica no sentido de estas assumirem igualmente as comunicações em aplicações com requisitos de segurança crítica, que estavam até então restritas a algumas topologias de *hardware*.

O PROFIBUS-DP apresenta-se como uma das redes de campo mais difundida em de aplicação da automação e controlo de processos. A sua evolução tem se produzido através da integração na sua estrutura de perfis que a adaptam a domínios de aplicação específicos, e a aplicações com requisitos de integridade de segurança. Neste último caso, através da utilização do *PROFIsafe* é possível obter níveis de integridade de segurança até SIL3. Comum à utilização de todos estes perfis está subjacente a utilização dos serviços de comunicação fornecidos pelo PROFIBUS-DP que por consequência do seu desempenho na presença de falta pode levar a rede para modos de funcionamento que se traduzam numa redução da sua disponibilidade.

Do apresentado, constata-se que para esta rede existe pouco trabalho publicado, acerca da sua operação em cenários de faltas. Da mesma forma os trabalhos apresentados baseiam a sua análise em modelos de simulação que são desenvolvidos tendo essencialmente como fonte de referência a norma da rede. Em sistemas complexos de que é exemplo o PROFIBUS-DP, este processo de desenvolvimento tende resultar em modelos que não representam de forma fidedigna o modo de operação da rede com aquele que ocorre na realidade na presença de faltas. Isto fica a dever-se essencialmente a dois factores:

- Identificações de eventos que não correspondem ao modo de operação real;
- Incompleta identificação dos modos de operação resultantes de erros.

Desta forma, a rede PROFIBUS-DP carece de uma avaliação da sua operação em cenários de faltas, que forneça informações precisas acerca do seu modo de operação, nomeadamente:

- Avaliar os efeitos das faltas no anel lógico, identificando os eventos que possam gerar instabilidade;
- Quantificar a probabilidade da ocorrência desse eventos para as condições de análise;
- Obter parâmetros de desempenho na presença de faltas da rede, nomeadamente tempo médio do ciclo do *token*, ou do tempo de interrupções momentâneas de serviço (*outages*) provocadas pela activação dos mecanismos de recuperação de faltas;
- Avaliar o impacto na resposta de tempo-real.

Uma forma de obter esta informação com um elevado grau de fidelidade, pressupõe a sua obtenção a partir de um sistema real (físico). Para que este processo seja efectuado em tempo útil é necessário o recurso à injecção de faltas como forma de acelerar a ocorrência dos eventos objecto de avaliação.

Neste contexto, torna-se necessário desenvolver uma infra-estrutura que suporte experiências de injecção de faltas, numa rede real, e que possua características que a tornem capaz de:

- Implementar nós de comunicação representativos daqueles que existem a nível industrial, e que estejam de acordo com a norma IEC61158;
- Permitir a injecção de faltas transitórias de forma controlada no barramento de comunicações;
- Permitir o registo de eventos, quer a nível do barramento de comunicações, quer nos próprios nós de comunicação;
- Possuir resolução temporal de forma a caracterizar os eventos e a poder obter medidas de desempenho na presença de faltas.

2.6 Síntese

As redes de comunicação assumem actualmente um papel preponderante na estrutura e no desempenho dos sistemas de controlo. A existência de uma organização hierárquica na qual os níveis de controlo apresentam requisistos substancialmente distintos propiciou o aparecimento de um tipo de rede de comunicação específico, designado por rede de campo.

O PROFIBUS-DP é uma rede com aplicação em sistemas de automação e controlo amplamente difundida que se insere nesta classe de redes.

Como parte integrante de sistemas de controlo, que podem apresentar requisitos de segurança críticos, ou que envolvam a utilização de equipamentos de capital intensivo, ou mesmo que da interrupção da operação se produzam perdas ao nível da matérias primas, a confiança no funcionamento do PROFIBUS-DP e de uma forma geral destas redes é um factor que não pode ser descurado.

A avaliação do funcionamento da rede quando esta é perturbada na sua operação é uma importante tarefa que permite caracterizar o seu comportamento, nomeadamente a sua susceptibilidade a determinados modos de operação e o contributo destes para a degradação do funcionamento.

Em síntese e com base nos contributos de trabalhos publicados na área, constata-se que a operação da rede PROFIBUS-DP é afectada de forma significativa, quer pela ocorrência de erros durante a troca de tramas de gestão, quer pela destruição de informação durante a troca de tramas de dados e consequente necessidade da sua repetição. Esta degradação de desempenho e a forma como a rede trata os erros poderá colocar questões relativas a possíveis impactos na operação do PROFIsafe. Do ponto de vista da segurança da operação do PROFisafe, a questão não se coloca, uma vez que são seguidas as recomendações de entidades certificadoras da segurança, designadamente da BIA [Piggin00] e da TÜV [TÜV05]. É também implementado um conjunto de mecanismos que tornam o tratamento dos erros independente dos mecanismos implementados nas camadas inferiores do PROFIBUS-DP. Desta forma o PROFIsafe é capaz de assegurar o SIL3. Existe contudo, uma interrelação entre os vários indicadores da confiança no funcionamento, e quando se privilegia um desses indicadores, poderá ter que existir uma cedência de um outro.

Neste cenário a ocorrência de erros poderá levar a pequenas interrupções de serviço das estações afectadas ou mesmo de todo o barramento, tornando impossível a comunicação com os escravos durante o período do seu efeito. Para salvaguardar o sistema de situações em que os intervenientes no processo de comunicação estejam para além de um intervalo de tempo seguro sem comunicações, o PROFIsafe implementa de um mecanismo de *timeout*. Este mecanismo permite aos intervenientes no processo desencadear acções de forma a poderem comutar para um estado de operação segura.

Este é um procedimento que proporciona um aumento da segurança de operação da rede mas que tem efeitos contrários ao nível da disponibilidade, podendo esta vir a ser reduzida e consequentemente levar a uma redução da confiança no funcionamento do sistema.

Do exposto fica evidenciado que o funcionamento da rede PROFIBUS-DP não está ainda completamente caracterizado, e muitas das análises baseiam-se em modelos que assentam em parâmetros de comportamento que têm unicamente como referência a sua norma. Desta forma os resultados obtidos podem diferir significativamente do que ocorre na realidade. Por outro lado questões com directa implicação na disponibilidade da rede são relevantes dado que esta poderá suportar aplicações onde este índice de confiança no funcionamento seja um requisito. Assim, assume particular importância que a caracterização da operação da rede seja efectuada tendo por base a recolha de informação a partir de estruturas reais. Esta abordagem permite obter as características de operação da rede na presença de erros, assim como de outras medidas relacionadas com a confiança no funcionamento e seu desempenho na presença de faltas, que estão de acordo com o que de facto ocorre para o cenário de erros considerado.

Esta metodologia ganha tanta mais importância quanto permite actuar de forma integrada com outro tipo de ferramentas, nomeadamente as de simulação, e assim funcionar como meio de validação dos seus modelos, potenciando todas as virtualidades dessas ferramentas [Portugal05].

Capítulo 3

Avaliação do Funcionamento por Injecção de Faltas

3.1 Introdução

A introdução de sistemas electrónicos na cadeia de controlo tornou-se possível devido aos crescentes níveis de fiabilidade apresentados pelos componentes que os constituem. Este atributo está associado ao uso massivo de electrónica no estado sólido (semicondutores), sendo actualmente expectável que em condições de operação normal, estes dispositivos assegurem o funcionamento durante um horizonte temporal na ordem das décadas [MIL91]. Este suporte tecnológico assume também maior relevo nomeadamente pela sua componente digital (electrónica digital) estar associada a tecnologias emergentes que contribuem para um aumento da capacidade de efectuar operações complexas que têm inerentemente um campo de aplicação mais amplo.

A integração desta tecnologia em aplicações de controlo levanta contudo algumas questões relacionadas com a caracterização dos modos de operação para os mais diversos cenários com que se pode deparar. Quando o nível de complexidade do sistema é baixo os vários modos de operação, incluindo aqueles que se traduzem em situações indesejáveis ou com perigosidade para pessoas e bens são facilmente identificados. Todavia, actualmente o panorama dos sistemas de controlo é dominado pela sua elevada complexidade, que se verifica ao nível lógico, funcional e da sua estrutura física, pelo que se torna extremamente difícil identificar e recrear cenários que levam à sua falha. A avaliação do funcionamento destes sistemas assume assim uma grande importância nomeadamente em aspectos relacionados com o estudo dos modos de operação que derivam da ocorrência de erros e de falhas.

3.2 Taxonomia da Confiança no Funcionamento

Nas últimas décadas produziram-se significativos avanços na área da confiança no funcionamento/confiabilidade. Tratando-se de uma área multidisciplinar, que requer uma interacção entre intervenientes que trabalham em diversas vertentes da confiança do funcionamento, tornou-se necessário fornecer uma ferramenta que permitisse a interacção e discussão sobre o tema da confiança no funcionamento, com base num conjunto de conceitos bem definidos. Neste sentido, e em resposta a esta necessidade, uma terminologia própria foi adoptada, tendo esta sido estabelecida pelo grupo de trabalho 10.4 do *Fault Tolerant Computing* do IFIP *International Federation of Information Processing,* [Laprie89] [Avizienis04]. Uma proposta para uniformização desta terminologia em língua portuguesa é apresentada em [Veríssimo89].

Nas suas diversas vertentes, a confiança no funcionamento, tem como objecto a qualidade com que um sistema fornece os seus serviços. Neste contexto, o comportamento do sistema está associado à percepção que os seus utilizadores têm da forma como o serviço é fornecido.

Assim, o serviço é considerado adequado se cumprir com as especificações do sistema, e é considerado inadequado quando as mesmas forem violadas. Com base nestes dois estados, é possível identificar se o sistema falhou o fornecimento do serviço, e em função dos seus requisitos de confiança no funcionamento, definir diferentes atributos de confiabilidade.

De uma forma genérica, o conceito de confiança no funcionamento assenta sobre três grandes vectores: atributos, impedimentos e medidas Fig. 3.1

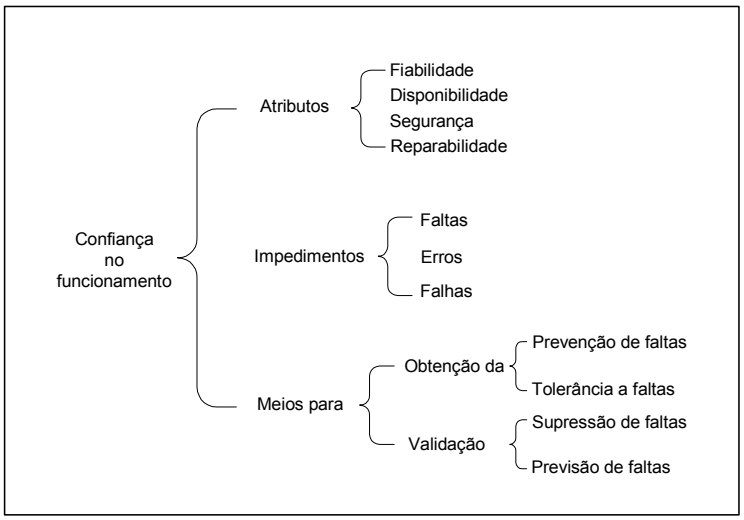

Figura 3.1 - Árvore da confiança no funcionamento.

3.2.1 Atributos

A confiança no funcionamento pode ser caracterizada por diferentes atributos relacionados com a aplicação específica do sistema, e dos seus requisitos de confiança no funcionamento. Assim, o grau de confiança no funcionamento, pode apresentar-se sobre diferentes medidas, tais como:

- Fiabilidade;
- Disponibilidade;
- Reparabilidade;
- Segurança.

Desta forma, se o factor mais importante no funcionamento sistema é a continuidade de fornecimento de serviço, então, a medida mais importante é a **fiabilidade** *R(t)*. Esta é expressa pela probabilidade do sistema fornecer o serviço ao longo de um intervalo de tempo *t*, e é usualmente expressa pelo tempo médio para a falha – *Mean Time To Failure* (MTTF), na terminologia inglesa.

$$MTTF = \int_0^\infty R(t)dt \tag{3.1}$$

O tempo necessário para reparar o sistema, após a ocorrência de uma falha, é dado pela **reparabilidade** *M(t)*. A reparabilidade, é medida pela probabilidade do serviço ser restaurado dentro de um limite de tempo *t*, sendo normalmente expressa pelo tempo médio para a reparação – *Mean Time To Repair* (MTTR) [Kopetz 98].

$$MTTR = \int_0^\infty M(t)dt \tag{3.2}$$

Por outro lado, quando o aspecto mais importante do funcionamento do sistema é o tempo que o serviço é disponibilizado adequadamente, em relação ao tempo em que o serviço é inadequado, a medida mais importante é **disponibilidade** *A(t),* que é expressa pela probabilidade do sistema fornecer serviço adequado num determinado tempo *t* . A disponibilidade é expressa por:

$$A = \lim_{t\to\infty} A(t) = \frac{MTTF}{MTTF + MTTR} \tag{3.3}$$

Quando, do adequado fornecimento do serviço, depende a integridade de pessoas ou de equipamentos, a característica mais importante é a segurança *S(t)*, que é expressa pela probabilidade do sistema permanecer seguro durante um período de tempo t, sendo normalmente expressa pelo tempo médio para a ocorrência de falhas catastróficas – *Mean Time To Catastrophic failure* (MTTC) [Kalbarczyk93].

$$MTTC = \int_0^\infty S(t)dt \tag{3.4}$$

3.2.2 Impedimentos

Durante a vida operacional de um sistema, este está sujeito a um conjunto de incidências que podem levar a sua resposta a desviar-se das suas especificações. Em última instância, pode mesmo fazer com que o sistema forneça um serviço inadequado, ou seja, levar a uma falha do sistema. Estas incidências, são compostas pelo conjunto de eventos que se opõem ao normal funcionamento do sistema, e, designam-se por impedimentos à confiança no funcionamento. Estes englobam quer as causas, quer as consequências dos referidos desvios e são caracterizados por três estados:

Falha – está associada à incapacidade de o sistema fornecer o serviço para que estava especificado;

Erro – é a sintomatologia da ocorrência de uma falta, que se revela através de um estado incorrecto no sistema. Numa primeira fase, o erro fica em estado latente podendo posteriormente tornar-se efectivo. Quando efectivo, a sua evolução no sistema poderá tornar-se benigna, não resultando daí consequências graves para a qualidade do serviço, ou, pode evolui de forma maligna e conduzir à falha do sistema;

Falta – é o fenómeno que origina o erro, e surge como o resultado de um conjunto de circunstâncias de origem diversa, as quais podem ser agrupadas em três grandes classes. Estas classes correspondem: aos fenómenos que as provocam, à sua localização e à sua característica temporal. Assim, as faltas podem ser classificadas quanto à:

- **Causa** – de causa física ou humana, estando a primeira associada a fenómenos físicos adversos, que podem ocorrer quer no interior do sistema, quer no seu exterior. A segunda é causada pela intervenção humana, podendo esta ser de natureza acidental ou intencional, que por sua vez pode ter origem na fase de desenvolvimento ou de operação;
- **Localização** – confinada ao interior ou exterior do sistema. A primeira resulta de faltas que se encontram latentes na estrutura do sistema. A segunda resulta da interacção do sistema com a sua envolvente, quer física quer humana;
- **Duração** – relativo ao domínio temporal e caracterização da sua persistência. Assim as faltas podem ser classificadas como permanentes ou temporárias. As primeiras persistem no sistema ao longo do tempo. As segundas têm uma existência limitada, assumindo por vez a designação de intermitentes quando ocorrem no interior do sistema, ou transitórias quando ocorrem no seu exterior.

A ocorrência de faltas, erros e falhas no sistema de uma forma geral, não se processa de forma simples. Dada a complexidade dos sistemas, estes três eventos tendem a interrelacionar-se, de forma que a falha de um subsistema é ela própria uma falta na entrada do subsistema que utiliza os seus serviços. Forma-se assim

uma cadeia que pode alterar de forma significativa a resposta do sistema, e tornar difícil a identificação da causa da sua falha.

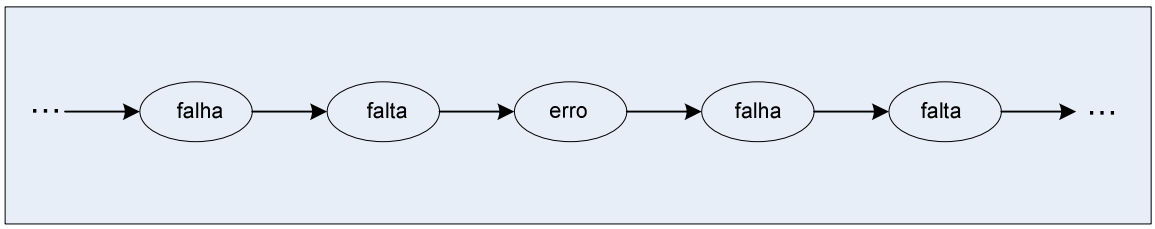

Figura 3.2 - Cadeia de propagação de falhas.

3.2.3 Meios

Diversos factores podem concorrer para que um sistema não forneça o serviço de forma adequada. Neste contexto, um conjunto de métodos e ferramentas deve ser empregue com intuito da obtenção de elevados índices de confiança no funcionamento. A aplicação destes métodos e técnicas pode ser efectuada através de duas abordagens complementares [Lee90]:

Prevenção de faltas – através desta abordagem procura-se eliminar todas as potenciais fontes de faltas, antes do sistema ser posto em serviço. Este processo é efectuado durante a fase de desenvolvimento, e passa pelo uso criterioso de técnicas e de tecnologias, de forma a procurar evitar a introdução de faltas, quer na fase de projecto, quer na fase de produção e implementação do sistema;

Tolerância a faltas – esta abordagem baseia-se na perspectiva de que é impossível garantir um sistema completamente livre de faltas, e assim, procura assegurar que o sistema consiga fornecer os seus serviços de forma adequada mesmo na sua presença. Desta forma, o sistema deve ser provido de um conjunto de recursos que lhe confiram, nomeadamente, capacidade de detecção de faltas, capacidade de as confinar e de usar processos que permitam compensar e recuperar dos efeitos das mesmas. Através do recurso à redundância é possível obter tolerância a faltas. A redundância é conferida através do uso, no sistema, de componentes e software adicional, para efectuar as tarefas referidas anteriormente, podendo a sua aplicação ser efectuada a três níveis [Kopetz 98]:

- Redundância de *Hardware*;
- Redundância de *Software;*
- Redundância temporal.

Como complemento a estas duas abordagens deve recorrer-se a outras técnicas com o intuito de alcançar maiores níveis de confiança no funcionamento. Essas técnicas podem ser aplicadas segundo duas perspectivas:

Supressão de faltas – Procura minimizar possíveis faltas que possam ter sido introduzidas no sistema numa das seguintes fases: concepção, desenvolvimento e produção. O processo de supressão de faltas envolve três funções [Folkesson99]:

- **Verificação** – para testar se o sistemas está de acordo com as suas especificações. A verificação pode ser efectuada através da análise estática ou de prova (*proof-of-correctness*), nas quais um conjunto de análises e inspecções são efectuadas ao sistema para verificar da sua conformidade, ou através de testes dinâmicos onde o funcionamento do sistema é verificado;
- **Diagnóstico** – para identificar faltas no sistema que contrariam as condições de verificação.
- **Correcção** – para efectuar as necessárias alterações e criar as condições para que o sistema fique conforme as condições de verificação.

Previsão de faltas – procura estimar a presença de faltas, futuras ocorrências e suas consequências. Este processo é efectuado através de dois tipo de métodos:

- **Qualitativos** – através de métodos qualitativos é estimado o comportamento do sistema na presença de faltas, recorrendo para tal quer a modelos analíticos, quer a métodos experimentais (teste);
- **Quantitativas** – através de métodos quantitativos são efectuadas medidas de confiabilidade do sistema, geralmente associadas à eficiência dos seus mecanismos de tolerância a faltas.

3.3 Injecção de Faltas

Para analisar a confiança no funcionamento é necessário estudar o comportamento do sistema na presença de faltas, verificar o efeito dos erros introduzidos por estas incidências e estudar as situações que possam levar a à sua falha. Efectuar estas análises durante a normal operação do sistema é uma tarefa de difícil realização. As principais dificuldades com que se depara estão associadas aos seguintes constrangimentos:

(i) **Grande latência das faltas:** os eventos que provocam as faltas têm um espaçamento temporal incerto com tendência para ser longo, o que inviabiliza a obtenção de resultados em tempo útil;

(ii) **Representatividade das faltas**: dificuldade de representação da totalidade do espaço de faltas (conjunto de faltas e locais onde ocorrem). A natureza aleatória dos fenómenos que podem afectar o sistema, aliado a um número extremamente elevado de combinações que englobam os locais onde as faltas podem ocorrer, os valores que podem tomar e os erros resultantes dessas mesmas faltas, tornam impraticável a validação de um sistema recorrendo somente à analise do seu funcionamento no seu ambiente;

(iii) **Natureza destrutiva das falhas**: existe a possibilidade das falhas provocarem a destruição do equipamento o que não é aceitável, quer em termos económicos, quer em termos de segurança.

Em função destes constrangimentos é necessário recorrer a um processo alternativo que permita acelerar a ocorrência de tais eventos, de modo a verificar o funcionamento do sistema na sua presença. Este processo denominado por injecção de faltas, consiste na introdução intencional e de forma controlada de um conjunto de faltas que seja representativo, daquelas que podem ocorrer durante todo o horizonte temporal para que o sistema foi projectado.

A injecção de faltas é uma técnica usada na análise da confiabilidade de sistemas, orientada quer à validação quer previsão de faltas. A sua aplicação está geralmente associada à fase de desenvolvimento do ciclo de vida do sistema, embora possa também ser também aplicada em fases posteriores à de protótipo [Clark95].

Tipicamente um sistema de injecção de faltas é constituído pelos seguintes módulos funcionais (Fig. 3.3) [Hsueh97]:

- **Controlador**: com a função de coordenar todas as tarefas efectuadas pelo sistema de injecção;
- **Injector**: bloco especializado para aplicar as faltas no sistema objecto de avaliação;
- **Monitor**: projectado para observar os eventos relevantes da operação do sistema objecto de avaliação e desencadear acções de recolha de dados;
- **Aquisição de Dados**: sistema que permite a recolha de dados que podem ser processados *online*, ou guardados para posterior processamento.

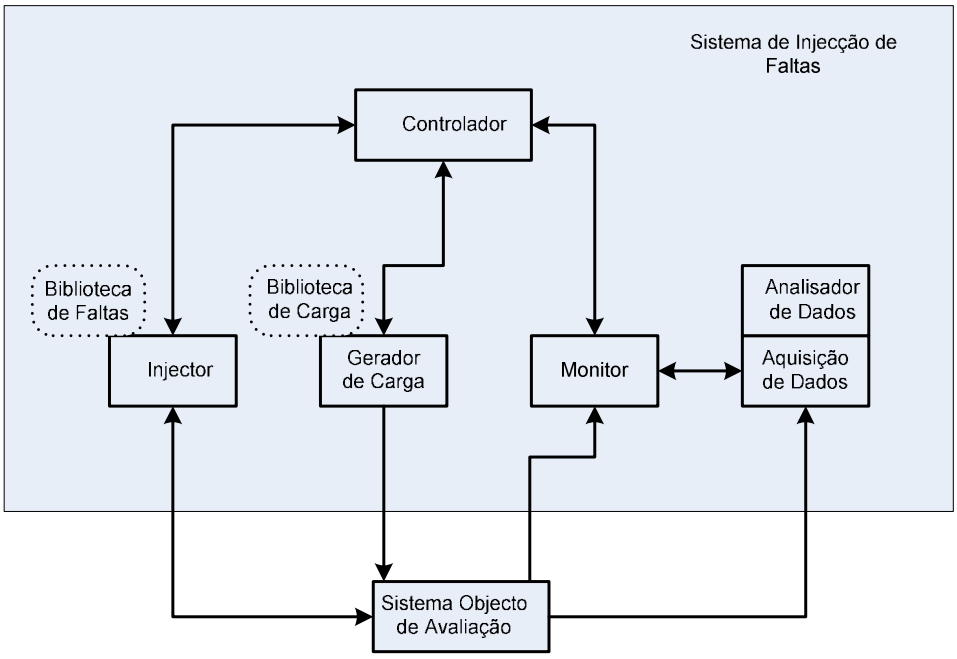

Figura 3.3 - Ambiente de injecção de faltas.

A injecção de faltas é efectuada em simultâneo com a operação do sistema. Durante a operação são executados comandos que representam o cenário de carga para os quais as experiências são válidas, sendo os diversos cenários armazenados nas bibliotecas do gerador de carga. O injector efectua a injecção de faltas de acordo com a informação armazenada na biblioteca de faltas. Este por sua vez pode ser especificado para suportar faltas de diferentes tipos, que podem ser injectadas em diferentes localizações do sistema, e activadas de acordo com uma ordem temporal, ou sincronizada por disparos baseados em eventos do mesmo.

Tipicamente os modelos de faltas mais frequentemente implementados são:

- **Forçar a (*Stuck at*)**: este modelo é usualmente utilizado para representar faltas permanentes. As faltas são estabelecidas fixando o sinal ao nível lógico um (*stuck at 1*) ou ao nível lógico zero (*stuck at 0*);
- **Inversão de bit (*bit–flips*)**: este modelo é tipicamente utilizado para representar faltas transitórias. As faltas processam-se através de inversões do valor lógico do bit ou dos bits afectados;
- **Passagens (*Bridging*)**: este modelo de faltas representa a ocorrência de um estado de condução simultânea não intencional (passagens), que normalmente ocorre em configurações de transístores. Este evento tem como consequência o estabelecimento de estados indeterminados diferentes daqueles que definem o nível lógico zero e um [Renovell95]. Este tipo de faltas também pode ocorrer para outros cenários em que exista condução eléctrica não intencional entre circuitos, nomeadamente entre circuitos adjacentes em que por alguma circunstância de funcionamento ocorre uma passagem entre ambos;
- **Circuito aberto (*Open*)**: representa faltas que decorrem da interrupção do circuito eléctrico.

As técnicas de injecção de faltas foram alvo de um profundo trabalho na perspectiva de fornecerem soluções que satisfizessem os exigentes requisitos de validação de sistemas. Esta evolução também resultou da necessidade de acompanhar os desenvolvimentos tecnológicos com o intuito da utilização dos recursos disponibilizados, ou como forma de suprir dificuldades impostas por essa mesma tecnologia.

Esse trabalho resultou numa grande diversidade de métodos de injecção de faltas. Em termos globais estes abordam a injecção de faltas segundo duas perspectivas: uma na qual a injecção é efectuada sobre modelos que permitem simular o funcionamento do sistema; outra em que a injecção de faltas decorre sobre um modelo real do sistema.

3.4 Injecção de Faltas em Modelos de Simulação

A injecção de faltas baseada em métodos de simulação tem como principal característica a modelação do sistema objecto de avaliação, através de uma representação não física do mesmo. Esta característica está associada a uma das principais vantagens deste método tornando-o aplicável nos primeiros estágios do ciclo de vida de um sistema. Desta forma é possível verificar e validar as soluções empregues no mesmo e simultaneamente analisar parâmetros de confiança no funcionamento avaliando o seu comportamento em cenário de faltas.

Actualmente os sistemas tendem a apresentar um nível de complexidade muito elevado. A modelação exaustiva que contemple os mais diversos detalhes do sistema, pode resultar numa dimensão extremamente elevada do modelo que torne impraticável a sua implementação. Numa perspectiva de racionalização e de controlo da dimensão do problema, muitos modelos baseiam-se em diferentes graus de abstracção do sistema. Existem tipicamente três níveis de abstracção que representam outros tantos graus de detalhe do modelo. A representação com maior detalhe situa-se ao nível eléctrico, seguindo um nível intermédio, com detalhe ao nível lógico, e um nível superior de abstracção assente sobre aspectos funcionais do sistema [Portugal02] [Folkesson99].

3.4.1 Nível Eléctrico

O nível eléctrico é o nível mais baixo de abstracção do sistema. A este nível as simulações são efectuadas tendo o transístor como elemento base de descrição do sistema, e como regras de análise as leis da física que descrevem o funcionamento dos componentes eléctrico/ electrónicos. As características deste modelo tornam-no indicado para a análise da resposta dos sistemas a faltas transitórias, com origem em variações de diferença de potencial e de correntes que se verificam nos componentes afectados.

O detalhe do modelo, permite que este possa ser aplicado a sistemas com diferente suporte tecnológico, como sejam os sistemas baseados em electrónica analógica, ou mesmo sistemas que integrem os dois tipos de tecnologia – analógica e digital. Esta é uma característica diferenciadora relativamente a técnicas baseadas em modelos lógicos e funcionais, as quais não podem, ou é difícil a sua aplicação a este tipo de sistemas.

A injecção de faltas baseada neste tipo de modelo é suportada por ferramentas de análise de sistemas eléctricos/electrónicos de que o SPICE é um exemplo. Dada a capacidade destas ferramentas para trabalharem com modelos que representam de forma muito fidedigna a constituição dos sistemas, são consequentemente capazes de fornecer resultados com um nível de rigor elevado Esta técnica foi aplicada para aferir o grau de precisão de modelos baseados na representação do sistema ao nível lógico.

Os resultados destas análises demonstram grandes discrepâncias entre os resultados obtidos através dos dois modelos, indiciando a necessidade de uma cuidada validação dos modelos de faltas baseados em graus de abstracção superiores [Ries94]. Em contraste com o grau de precisão dos seus resultados, esta técnica apresenta alguns inconvenientes de que se destacam: os tempos de simulação consideravelmente elevados, e a elevada necessidade de recursos, como sejam capacidade de processamento e espaço para armazenamento de informação.

Na tentativa de diminuir o tempo de simulação esta técnica foi submetida a melhoramentos, nomeadamente através do recurso a métodos que permitem usar de forma combinada os dois tipos de modelo – eléctrico e lógico. Desta forma, durante a simulação, a representação do circuito é comutada entre os dois tipos de modelos, permitindo obter ganhos de desempenho ao nível da velocidade de análise [Yang92].

O nível de utilização dos recursos está ligado ao grau de complexidade dos sistemas. Assim, o aumento da complexidade dos sistemas conduz facilmente à exaustação dos recursos de simulação, quer ao nível do espaço de armazenamento, quer relativamente à necessidade de capacidade de processamento. Com o intuito de minorar este problema foi proposta uma abordagem na qual se faz uso do conhecimento prévio do comportamento dos componentes do sistema em cenários de faltas. Esta informação é guardada em blocos designados de dicionários de faltas. Depois da informação ser pré-processada, os dicionários de faltas quando invocados permitem reduzir o grau de necessidade dos referidos recursos [Choi93].

3.4.2 Nível Lógico

Ao nível lógico, as simulações são efectuadas tendo como elemento básico a porta lógica e, como regras de análise, as funções da lógica que descrevem o funcionamento destes dispositivos. Desta forma, é possível obter modelos mais compactos comparativamente aos modelos do nível eléctrico. Esta vantagem torna-se mais evidente quando se modelam sistemas complexos de escala muito elevada de integração (VLSI – *Very Large Scale Integration*), de que são exemplo os actuais microprocessadores e microcontroladores.

O desenvolvimento de ferramentas suportadas por linguagens de descrição de hardware, capazes de suportar tarefas como projecto, simulação e validação, amplamente aceites pelos fabricantes de semicondutores, tornaram-se também elas próprias num suporte tecnológico importante para a área da injecção de faltas. Embora a validação efectuada pelos fabricantes de semicondutores não corresponda ao conceito de validação por injecção de faltas, sendo antes aplicada na perspectiva de teste para verificação da qualidade do processo de produção, as linguagens de descrição de hardware, apresentam características bastante favoráveis à sua utilização para tal tarefa. Assim, características como capacidade de descrever um sistema, quer ao nível da sua estrutura, quer do seu

funcionamento [Juan93], aliada à capacidade de representações hierárquicas de sistemas digitais, permite a modelação de sistemas de maior escala como por exemplo o computador. Estas características tornaram o VHDL, na mais difundida linguagem de descrição de hardware, em aplicação de injecção de faltas [Parrotta00] [Garcia02] [Zarandi03] [Baraza05].

A injecção de faltas em VHDL recorre tipicamente a duas técnicas. Uma das técnicas requer a modificação do modelo de descrição do hardware. A outra é implementada com base em comando fornecidos pela própria linguagem. Em ambos os casos os modelos de faltas usualmente aplicados são: forçar a 0 (*stuck at 0*), forçar a 1 (*stuck at 1*) *e* inversão de bit *(bit-flip)*.

3.4.2.1 Técnicas Baseadas em Modificações do Modelo

Nesta técnica as modificações são efectuadas com o objectivo de introduzir componentes que depois irão ser os responsáveis pela injecção de faltas. As modificações são efectuadas pela introdução de elementos especializados, ou pelo recurso à mutação de componentes.

Os elementos especializados são denominados por *sabotadores* [Jenn94]. Os *sabotadores* são blocos de código VHDL comandados, que quando inactivos não interferem no normal funcionamento do sistema, e quando activos, injectam faltas que se repercutem no sistema através da alteração do valor ou das características temporais do sinal que é afectado.

A inserção do *sabotador* no sistema pode ser efectuada recorrendo a uma das configurações: série ou paralela. Na configuração série o *sabotador* é inserido nas ligações entre as saídas de sinal (*drivers*) e os seus receptores (*inputs*). Na implementação paralela o *sabotador* é ele próprio um *driver* que por sua vez é ligado em paralelo a um conjunto de outros *drivers* que fazem parte do circuito objecto de avaliação. Através das funcionalidades da linguagem é depois possível seleccionar o sinal desejado da configuração resultante de forma a ter controlo sobre o processo de injecção de faltas [Jenn94].

A injecção de faltas, recorrendo a mutações no código, é efectuada através da utilização de elementos designados por *mutantes*. Um *mutante* é um elemento no qual foram feitas alterações na sua descrição gerando assim uma mutação na sua função. Quando inactivo, este bloco de VHDL desempenha a sua função normal, mas quando comandado, o seu comportamento é alterado emulando assim um comportamento em presença de faltas. A mutação pode ser efectuada de diversas formas [Jenn94]. Nestas incluem-se alterações estruturais da descrição do componente, como por exemplo a substituição de uma porta NAND por uma NOR e a alteração das declarações ao nível funcional, nomeadamente através da geração de operadores errados, ou pela modificação dos identificadores de variáveis.

3.4.2.2 Técnicas Suportadas em Comandos da Linguagem

As técnicas baseadas em comandos da própria linguagem têm como principal vantagem, o facto de não obrigar à introdução de alterações no modelo do sistema, tendo contudo como limitação o grau de funcionalidade disponibilizada pelos comandos da linguagem.

A aplicação desta técnica pode ser efectuada de acordo com duas abordagens:

- **Manipulação de sinais**: nesta abordagem o comportamento do sistema é simulado, mudando de contexto em pontos temporais preestabelecidos onde são injectadas as faltas. Assim, nesses pontos os sinais são desligados dos *drivers* e o seu valor é alterado de acordo com o modelo de faltas;
- **Manipulação de variáveis**: esta abordagem é em tudo idêntica à manipulação de sinais no que diz respeito ao processo de alteração dos valores. Contudo esta é efectuada em variáveis do modelo e não nos sinais.

3.4.3 Nível Funcional

No nível funcional os modelos representam o sistema de acordo com o funcionamento de um conjunto de objectos que o constituem. A principal vantagem desta abordagem está relacionada com a possibilidade de analisar a confiança no funcionamento de sistemas de grande complexidade, de que são exemplo os sistemas distribuídos. Contudo, para um bom desempenho desta técnica é necessário obter modelos de faltas que sejam representativos do funcionamento de uma grande diversidade e heterogeneidade de componentes. Assim, torna-se necessária uma correcta modelação das faltas a níveis inferiores de abstracção, como seja o nível lógico, de componentes como microprocessadores, memórias interfaces de comunicações, e demais componentes do sistema.

Uma ferramenta de injecção de faltas que recorre a este conceito é a DEPEND [Goswami97]. Esta ferramenta foi concebida para fornecer um ambiente de desenvolvimento integrado, permitindo o projecto de arquitecturas de sistemas tolerantes a falhas e o suporte para execução de testes de injecção de faltas, para avaliar as capacidades das arquitecturas projectadas.

3.5 Injecção de Faltas em Protótipo

Quando o ciclo de desenvolvimento de um sistema atinge a fase de implementação, abre-se a possibilidade da análise do sistema poder ser efectuada

através de técnicas injecção de faltas aplicadas ao próprio sistema. Este tipo de técnica é designado por injecção física de faltas [Hsueh97] [STSARCES00].

A injecção de faltas num sistema físico (protótipo) tem como principal vantagem o facto de se poder avaliar o comportamento de um sistema sem necessidade de recorrer a um modelo, no qual são necessariamente assumidos comportamentos que se consideram representativos da realidade. Esta característica reveste-se de grande importância, uma vez que, os resultados assim obtidos reflectem com elevado grau de rigor, o que de facto ocorre para as condições de operação representadas pelos cenários utilizados nas experiências.

Do ponto de vista da implementação este tipo de técnica é caracterizado pela existência de uma elevada quantidade de variantes. Estas por sua vez podem ser classificadas em dois grandes grupos, tendo como afinidade o processo de injecção:

- Um grupo no qual as faltas são injectadas directamente no *hardware* do sistema;
- Um segundo grupo em que as faltas são injectadas através do *software* do sistema (*injecção de faltas por emulação*).

3.6 Injecção Física de Faltas

A injecção física de faltas (*Hardware based fault injection*) processa-se através da alteração de sinais eléctricos do hardware do sistema. Este processo é desempenhado por dispositivos especiais designados por injectores, cuja configuração varia de acordo com as especificidades dos sistemas e dos locais onde as faltas são injectadas.

Três grupos tecnológicos suportam as técnicas de injecção física de faltas:

- **Injecção por contacto eléctrico**. Neste grupo incluem-se as técnicas nas quais durante o processo de injecção existe contacto eléctrico entre injector e o sistema objecto de avaliação.
- **Injecção por radiação**. Neste grupo são incluídas a técnicas que recorrem a qualquer processo onde não ocorra contacto entre injector e o sistema objecto de avaliação.
- **Injecção com suporte a instrumentação no próprio circuito**. Este grupo de técnicas emergentes é suportado por tecnologias associadas ao desenvolvimento de produtos na área do micro electrónica.

3.6.1 Injecção de Faltas com Contacto Eléctrico

Dentro do espectro das técnicas de injecção física de faltas, a injecção por contacto e mais concretamente a técnica de injecção ao nível do pino (*pin level fault injection*) é uma das mais difundidas. Esta técnica tem como princípio de funcionamento a alteração do potencial eléctrico nos pinos dos circuitos electrónicos.

Tipicamente existem dois processos para efectuar a alteração do referido potencial eléctrico: forçar o nível do sinal (*forcing*) e a inserção de sinal (*insertion*).

3.6.1.1 Forçar o Nível do Sinal

Quando a injecção de faltas se processa forçando o nível dos sinais (*forcing*), estas são injectadas directamente nos pinos dos circuitos integrados por intermédio de pontas de prova activas. As pontas de prova aplicam potenciais eléctricos nos pinos e demais componentes ligados a eles através de linhas equipotenciais que são constituídas pelas pistas dos circuitos impressos (Fig. 3.4). Tipicamente o modelo de faltas implementado para forçar o nível do sinal é o *stuck at*. Com este modelo as pontas activas estabelecem um potencial eléctrico que corresponda ao nível lógico um ou zero.

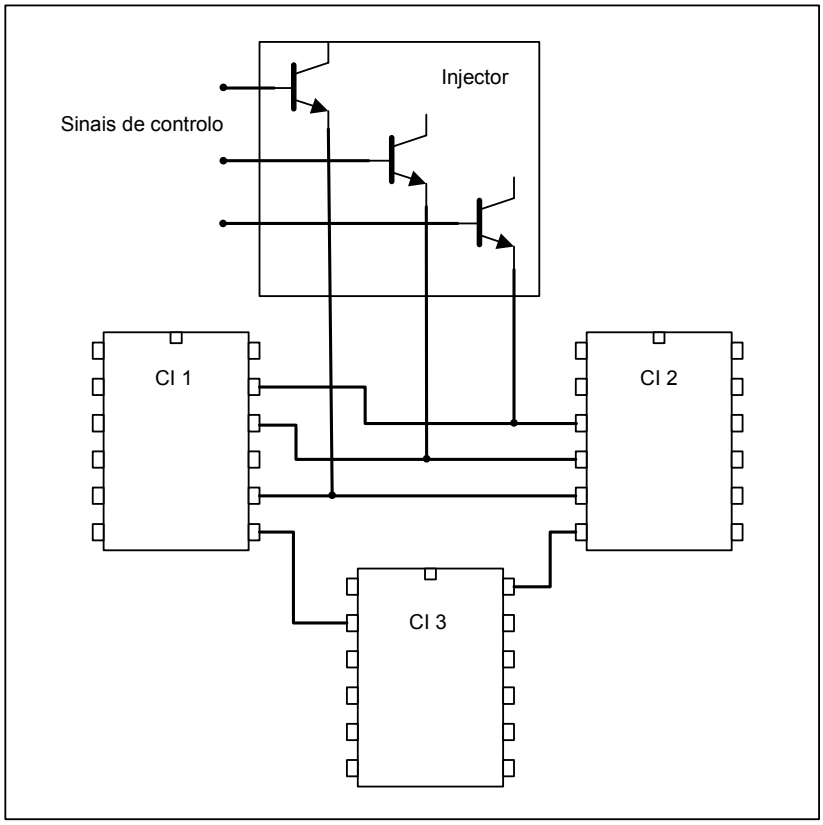

Figura 3.4 - Injecção de faltas nos pinos dos circuitos electrónicos, através da técnica que força o nível do sinal.

Esta técnica foi utilizada como configuração do injector em diversas ferramentas, algumas das quais foram aplicadas na validação de sistema tolerantes a falhas com requisitos de segurança críticos.

Demonstrativo da importância desta técnica e a sua adequação à validação de sistemas, foi a sua utilização na ferramenta MESSALINE. Uma ferramenta que foi desenvolvida com o objectivo de avaliar o funcionamento de um sistema centralizado de controlo de tráfego ferroviário baseado em computador, que foi utilizado pela empresa de caminhos-de-ferro Francesa SNCF. A ferramenta foi igualmente utilizada para validação de uma arquitectura de sistemas de computadores distribuídos tolerante a faltas, desenvolvida no âmbito do projecto ESPRIT Delta-4 [Arlat90]. A concepção desta arquitectura visou a sua utilização em aplicações de automação e controlo, nomeadamente na integração de sistemas de fabrico.

A ferramenta MESSALINE foi desenvolvida no *Laboratory for Automatics and Systems Analysis (LAAS)* Toulouse. A arquitectura do injector utilizado na ferramenta suporta até 32 pontos de injecção, os quais podem efectuar a injecção de faltas de acordo com uma técnica que força o nível do sinal ou de outra que faça a inserção de sinais no circuito. A inclusão de hardware adicional para suportar as duas técnicas, foi implementado com o objectivo da ferramenta suportar um espectro mais largo de modelos de faltas e assim mais representativo daquelas que podem ocorrer no hardware. Com esta configuração é possível injectar faltas do tipo: forçar a, inversão de bits., passagens e circuitos abertos.

Um outro exemplo de validação por este tipo de técnica é apresentado em [Walter90]. Neste trabalho foi desenvolvida uma ferramenta para validar um protótipo de uma arquitectura distribuída para aplicações com elevada necessidade de confiança no funcionamento MAFT (*Multicomputer Architecture for Fault Tolerance*) [Walter90] [Kieckhafer88]. Na validação foram efectuados ensaios envolvendo mais de 2000 experiências de injecções de faltas, que foram correctamente detectados pelos mecanismos de tolerância a faltas. Contudo devido aos resultados obtidos terem sido inconclusivos, muito possivelmente por causa de uma insuficiente dimensão das experiências, a análise experimental foi complementada analiticamente. Dada a complexidade da avaliação deste tipo de sistemas, o autor considera como benéfica a utilização combinada de métodos experimentais com métodos analíticos, neste tipo de análise.

Actualmente os sistemas tendem a ser desenvolvidos sobre hardware cada vez mais complexo e com frequências de trabalho elevadas. Esta tendência coloca acrescidos problemas à implementação das técnicas de injecção física de faltas e em particular à técnica de injecção ao nível do pino. Um dos problemas está relacionado com a necessidade de injectar faltas com duração extremamente curta. Neste contexto um grupo de trabalho da Universidade de Valência desenvolveu uma ferramenta capaz de trabalhar a frequências mais elevadas. A ferramenta denominada AFIT (*Advanced Fault Injection Tool*) implementa a técnica que força o nível dos sinais, e o seu injector è constituído por hardware capaz de injectar faltas com resolução temporal até 25ns [Martínez99].

Um outro problema associado ao aumento da frequência de trabalho dos circuitos electrónicos está relacionado com o comportamento dos elementos activos que constituem as provas de injecção. O *hardware* em causa é constituído por transístores que trabalham no modo de comutação. Estes componentes electrónicos possuem características físicas como a existência de capacidades que a frequências elevadas podem provocar atrasos e oscilações indesejáveis. Este comportamento tem como consequência a possibilidade de ocorrência de cenários em que a injecção não produz o efeito pretendido podendo levar a um maior nível de interferências indesejáveis, ou mesmo que a injecção não seja efectuada com sucesso.

Na ferramenta AFIT estes eventos indesejáveis são monitorizados através de um sistema de aquisição cuja operação está condicionada à verificação de diferentes disparos (*triggers*) dos quais depende a evolução do processo de injecção. Com base neste método é possível detectar experiências inválidas e filtrar os seus resultados.

De forma a demonstrar a sua aplicabilidade à validação da confiança no funcionamento de sistemas, foi efectuada uma análise da confiabilidade ao sistema FASST (*Fault Tolerant Architecture with Stable Storage Technology*). O FASST é um sistema multiprocessador tolerante a falhas, composto por vários módulos *fail silent*. Da avaliação do comportamento do sistema foi possível obter medidas de factores de cobertura dos mecanismos de tolerância a faltas e dos mecanismos de reconfiguração, tendo por base um conjunto de experiências de injecção de faltas, injectadas de forma aleatória em locais como: barramento de dados, endereços e de controlo.

3.6.1.2 Inserção de Sinal

A técnica de inserção de sinal (*insertion*) recorre a hardware adicional que durante o processo de injecção desliga partes do circuito substituindo-se a estes. Desta forma, as saídas dos circuitos electrónicos são desligadas do circuito e nas entradas são impostos sinais correspondentes às faltas desejadas (Fig. 3.5).

Este procedimento apresenta duas vantagens quando comparado com a técnica que força o sinal dos circuitos.

- **Menor risco de danificar o hardware.** Quando se força um potencial eléctrico a probabilidade de danificar a electrónica associada a um pino que apresente baixa impedância (saídas), é consideravelmente superior à da técnica de inserção de sinal. A avaria da porta decorre da circulação pela mesma de um valor de corrente superior àquela para que foi projectada;
- **Maior facilidade de inversão de sinais.** A facilidade de inversão de sinais simplifica a implementação de modelos de faltas do tipo inversão de bits. Contudo a técnica de inserção de sinal é consideravelmente mais difícil de aplicar a sistemas que operam a frequências mais elevadas.

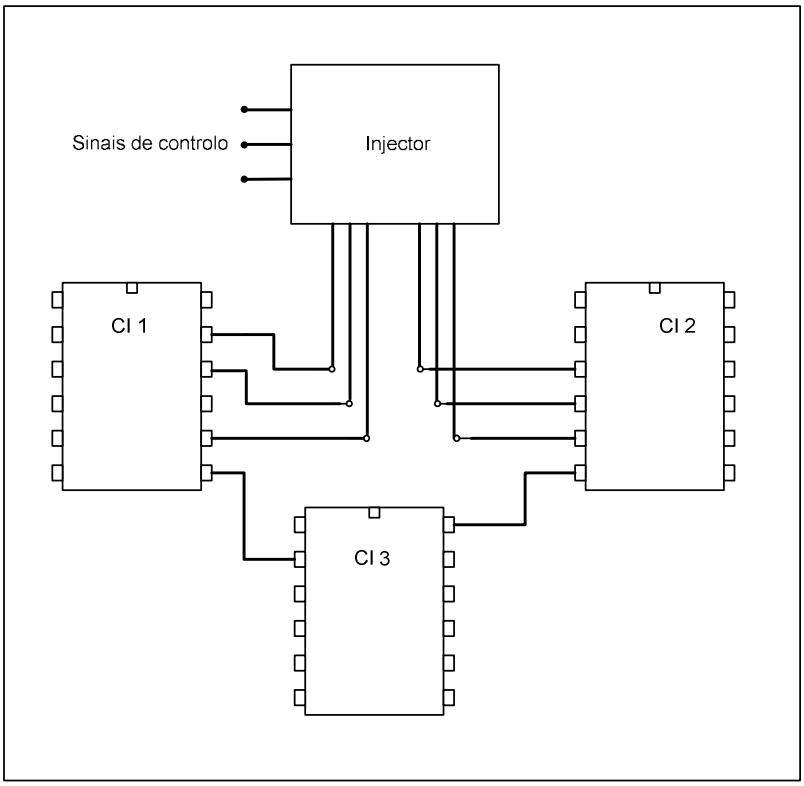

Figura 3.5 - Injector de faltas implementando a técnica de inserção de sinal.

A técnica de inserção de sinais foi utilizada em diversos trabalhos como a configuração base do injector ou fazendo parte da arquitectura de injectores com capacidade para injectar faltas de acordo com técnicas que forçam ou inserem sinais nos circuitos. São exemplo da aplicação destas duas configurações a ferramenta RIFLE e uma ferramenta de injecção de faltas aplicada à arquitectura SCRIBO (*Self-Checking RISC Board*) [Steininger97] [Steininger02].

A RIFLE é uma ferramenta desenvolvida na Universidade de Coimbra (Portugal) que implementa a técnica de inserção de sinal [Madeira94]. O injector foi concebido para funcionar como uma ferramenta multi-função, capaz de injectar faltas de acordo com vários modelos, e de ser adaptável à validação de sistemas implementados com base em diferentes microprocessadores. Nas configurações de hardware suportado encontram-se os microprocessadores da família 68000, Z80, 486DX e o InmosT800.

A técnica de inserção de sinal está também presente na ferramenta utilizada para avaliar o comportamento dos mecanismos de detecção de erros da arquitectura SCRIBO. A arquitectura SCRIBO é constituída pelo processador MC88100 da Motorola com sistema de gestão de memória (MMU) e *caches* externas. Esta arquitectura foi submetida a faltas no barramento de dados e nos sinais do relógio do sistema. Com base nos resultados obtidos foi possível identificar a combinação mais eficiente de mecanismos de detecção de erros, de forma a obter uma optimização baseada não só em critérios de factores de cobertura mas também relacionados com a disponibilidade (*availability*) e custos de implementação [Steininger97] [Steininger02]. Estes resultados são

demonstrativos da utilidade desta técnica na validação da confiança no funcionamento e na medida de desempenho dos sistemas na presença de faltas.

3.6.1.3 Restrições à Implementação da Técnica de Injecção ao Nível do Pino

Apesar da técnica de injecção ao nível do pino ser uma das técnicas mais utilizadas na injecção física de faltas, os desenvolvimentos verificados na indústria de produção de circuitos electrónicos apontam para soluções que dificultam a implementação desta técnica. As tendências de evolução são:

- **Complexidade dos circuitos**. Actualmente o nível de integração tem vindo crescer drasticamente, atingindo um nível de integração no sentido da inclusão do sistema num único circuito integrado (SOC – *System-On- a-Chip*). Esta configuração impõe algumas limitações à avaliação do sistema, nomeadamente por dificultar o acesso a componentes que se encontram no seu interior;
- **Configuração dos circuitos**. Associado ao nível de integração, o formato dos circuitos integrados tem vindo sofrer um processo de miniaturização verificando-se um aumento considerável da densidade de pinos. Da mesma forma tem se assistido a uma evolução no tipo de encapsulamento dos circuitos. Actualmente fabricam-se circuitos integrados em que os pinos não estão expostos e o seu número ascende às centenas, e em alguns casos já atingem o milhar (Fig. 3.6)
- **Frequência de trabalho**. O aumento da frequência de trabalho exige injectores mais rápidos. Esse aumento da frequência impossibilita em certos cenários a utilização da técnica de inserção de sinal, e mesmo a técnica que força o sinal torna-se de difícil implementação.

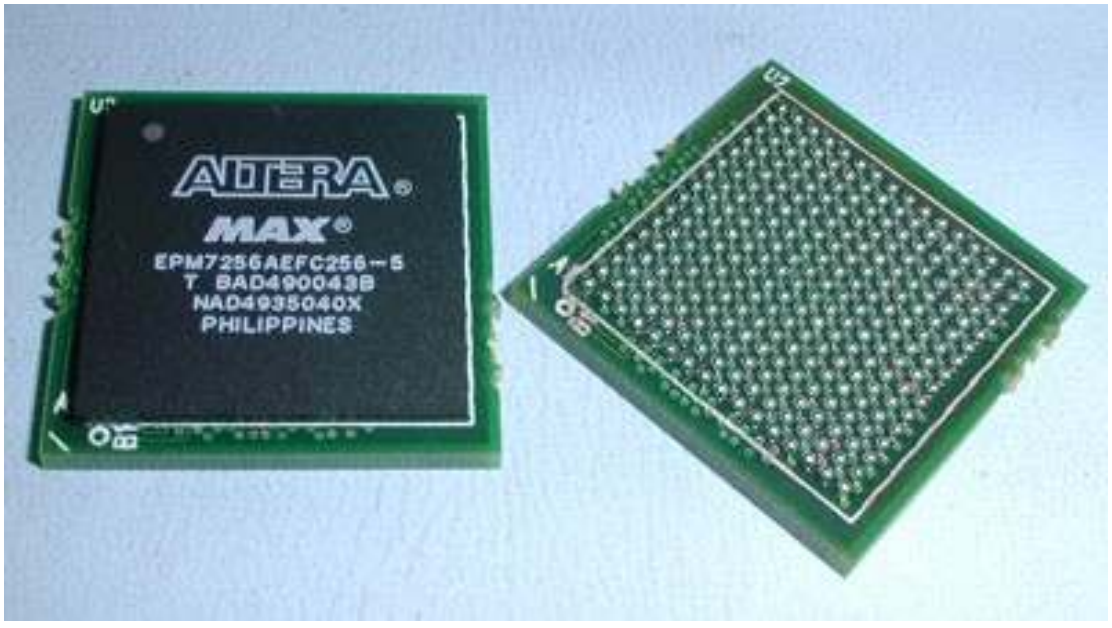

Figura 3.6 - Encapsulamento BGA de circuitos electrónicos de muito elevada escala de integração.

3.6.2 Injecção de Faltas por Radiação

As técnicas de injecção de faltas por radiação são técnicas em que não existe contacto entre injector e o hardware do sistema. Neste tipo de técnica as faltas podem ser injectadas no exterior ou no interior dos circuitos electrónicos. O processo de injecção consiste na emissão de uma radiação sobre a parte do circuito onde se pretende injectar faltas.

Os processos de radiação estão associados à existência de campos eléctricos e magnéticos (radiação electromagnética), que dependendo da sua frequência poderão tomar a forma de luz visível, ondas de rádio, raio X entre outras. Não obstante a necessidade de existência de rigor físico quando se descreve estes fenómenos, nesta dissertação, por uma questão de simplicidade, o termo radiação electromagnética será restrito à frequência de ondas rádio. Isto independentemente dessa radiação ser emitida por emissores de radiofrequência, ou seja derivada a fenómenos associados a grandes variações de correntes eléctricas, como as que resultam da operação de equipamentos eléctricos industriais.

Neste contexto, as técnicas de injecção de faltas que baseiam o seu princípio de funcionamento em processos de radiação serão agrupados nas seguintes classes:

- Electromagnéticas;
- Bombardeamento de partículas;
- LASER.

3.6.2.1 Injecção de Faltas por Radiação Electromagnética

A ocorrência de erros em sistemas electrónicos devido a interferências electromagnéticas é um fenómeno que ocorre com alguma frequência. Estes eventos são tanto mais frequentes quando no seu ambiente de trabalho coexistem equipamentos que são fontes de radiação electromagnética. Exemplos típicos são os ambientes industriais ou outros em que exista a comutação de cargas indutivas (ex. motores), ou então onde exista fontes de radiação electromagnética (ex. radares ou outros emissores de rádio frequências).

Tirando partido do mesmo processo que provoca interferências na operação real dos sistemas, a interferência electromagnética (EMI) pode ser utilizada de forma intencional para perturbar o funcionamento de sistemas electrónico e estudar o seu comportamento nestas situações. A utilização da EMI como técnica de injecção de faltas foi implementada na Universidade Técnica de Viena (Áustria). Uma ferramenta de injecção de faltas foi desenvolvida baseada nesta técnica, com o objectivo de avaliar o comportamento do sistema MARS (*Maintainable Real-Time System*). O MARS é uma arquitectura distribuída tolerante a faltas capaz de suportar aplicações com restrições temporais criticas,

sendo constituída por nós distribuídos autónomos com capacidade de processamento e com comportamento *fail safe*.

A ferramenta de injecção de faltas é composta por um gerador de impulsos e uma ponta de prova emissora de radiação electromagnética (Fig. 3.7).

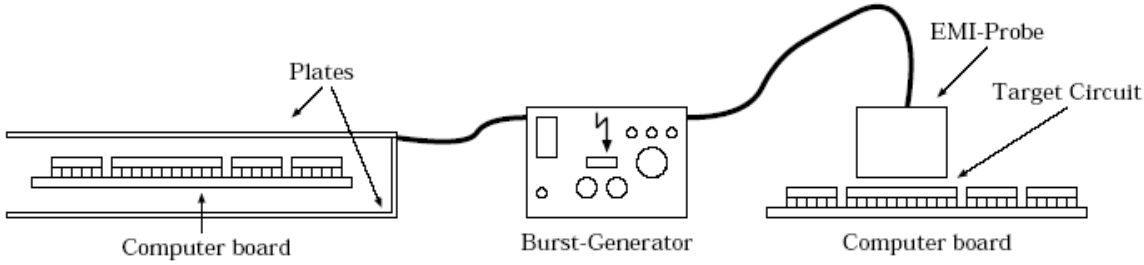

Figura 3.7 - Ferramenta de injecção de faltas por interferência electromagnética.

O gerador de impulsos tem a capacidade de gerar impulsos de acordo com a norma IEC801-4, ou seja impulsos que permitem representar o comportamento electromagnético provocado pela comutação de cargas, nomeadamente as de natureza indutiva [IEC88]. Estes impulsos tomam a forma de *burst* com a duração de 15 ms, e com um período entre *burst* de 300ms. A frequência de oscilação da onda que os constitui é configurável e pode assumir os seguintes valores: 1.25, 2.5, 5 e 10kHz (Fig. 3.8). Os *burst* são gerados por impulsos de corrente cuja intensidade é uma função da diferença de potencial podendo esta ser configurada para valores entre 225V e 4.4kV [Karlsson95].

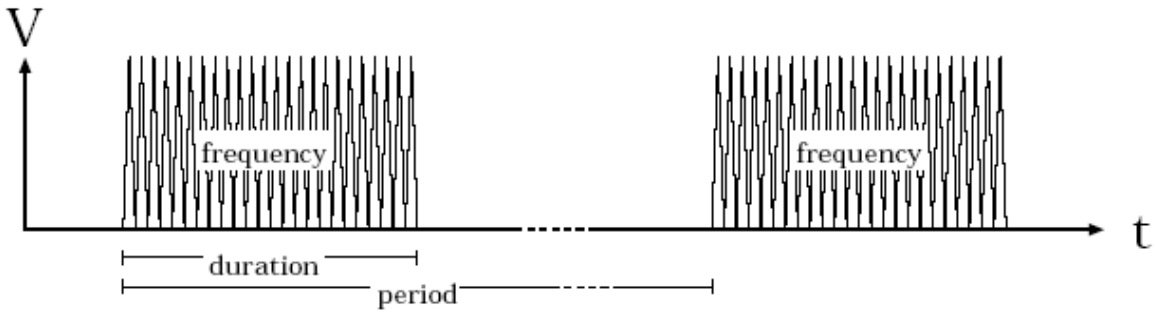

Figura 3.8 - Configuração das da onda electromagnética.

As pontas de provas emissoras podem ter duas configurações. Uma constituída por dois planos condutores (emissores) que envolvem todo o circuito sujeito a avaliação, ficando assim o circuito no centro do emissor de radiação electromagnética. Uma outra constituída por uma ponta de prova de dimensões reduzidas que permite a aplicação das radiações electromagnéticas de forma mais localizada.

Do ponto de vista da aplicação esta técnica apresenta algumas dificuldades que se manifestam a diversos níveis.

- **Difícil observação da activação das faltas e seus efeitos**. Esta dificuldade obriga à utilização de um sistema idêntico a trabalhar de forma síncrona, mas que não seja afectado pelas faltas. Este sistema

adicional funciona como nó de referência para obtenção de resultados por comparação.

- **Difícil controlo do processo de injecção**. Isto decorre não só da dificuldade de injectar faltas em locais precisos, mas também do seu controlo temporal. Relativamente à localização das faltas, a utilização de provas que permitam uma emissão mais direccionada, assim como a utilização de pequenos condutores orientados para aos pinos do circuito, são soluções que procuram melhorar a controlabilidade desta técnica. Esta última solução é usada com o intuito de os condutores funcionarem como antenas e aumentarem a probabilidade de os sinais desses pinos serem afectados.
- **Estatisticamente as experiências são de difícil reprodução**. Pequenas variações da disposição das provas ou do circuito objecto de avaliação geram resultados estatisticamente diferentes.

3.6.2.2 Injecção de Faltas por Bombardeamento de Partículas

O impacto de partículas com elevada energia na estrutura semicondutora dos circuitos electrónicos tem efeitos que se reflectem no seu funcionamento. Durante o impacto, as partículas perdem energia, principalmente através de um processo de ionização, que se traduz numa concentração pontual de carga na estrutura semicondutora. Esta concentração de carga induz um conjunto de efeitos que podem ter carácter transitório ou permanente. Neste último caso pode ter consequências catastróficas do ponto de vista da integridade do dispositivo electrónico. Na perspectiva de análise do funcionamento dos circuitos electrónicos, este processo de ionização traduz-se no seguinte conjunto de eventos [LaBel96]:

- ***Single Event Upset* (SEU).** Está associado a uma alteração de estado transitória, induzida pela ionização de partículas como raios cósmicos ou de protões. Esta alteração de estado pode ocorrer em circuitos analógicos, digitais ou mesmo ópticos. Em termos digitais o seu efeito pode traduzir-se pela inversão de bits (*bit flip*);
- ***Multiple Bit Upset* (MBU)**. Este tipo de evento partilha as características fundamentais do SEU. Contudo em circuitos de elevada densidade, a carga gerada pelo processo de ionização pode afectar partes do substrato que são partilhadas por diferentes elementos do circuito, e desta forma produzir múltiplas mudanças de estado;
- ***Single Event Latchup* (SEL)**. Este evento ocorre quando a carga gerada pela energia das partículas provoca uma disrupção da normal operação dos transístores que constituem o circuito. Este processo provoca a perda parcial da funcionalidade do circuito, e gera correntes elevadas no local onde o evento ocorreu. Isto está geralmente associado à existência na estrutura semicondutora de transístores bipolares parasitas. Através do processo de ionização estes transístores

parasitas podem originar o aparecimento de uma configuração semelhante a um rectificador controlado. Quando activado no modo de condução, esta configuração pode produzir efeitos catastróficos no circuito electrónico que o levem à destruição. Embora a maior parte dos actuais circuitos possuam circuitos especialmente concebidos para proteger destas situações - *latchup* (*clamp circuits*), estes são aplicados unicamente nas entradas e saídas do circuito. Assim, a ocorrência destes eventos no seu interior pode ter graves consequências para a sua integridade.

O impacto que estes eventos têm no funcionamento dos circuitos electrónicos, associado à possibilidade do seu carácter destrutivo, obriga que a sua ocorrência seja considerada e objecto de estudo. Este estudo é efectuado em duas vertentes:

- Uma relacionada com a análise da susceptibilidade dos circuitos electrónicos a estes eventos, e estudo das técnicas que minimizem a probabilidade de ocorrência;
- Uma outra inserida numa perspectiva da análise da confiança no funcionamento do sistema quando este é afectado por este tipo de eventos.

Embora em aplicações no domínio da indústria aeroespacial os SEU's causados pelos raios cósmicos sejam um problema com uma importância significativa, o seu estudo em aplicações de âmbito terrestre não representam um problema de igual grandeza. Isto, não obstante as evoluções tecnológicas que se verificaram no sentido do aumento de frequência de trabalho e a diminuição da escala dos semicondutores aumentar a possibilidade de interferências que produzem eventos semelhantes. Contudo nestas últimas aplicações o estudo dos efeitos dos SEU's é importante na perspectiva da avaliação da capacidade dos mecanismos de detecção e de recuperação de faltas.

Duas das técnicas mais usadas para efectuar este tipo de análise são:

- Técnicas baseadas em aceleradores de partículas;
- Técnicas baseadas em fontes radioactivas.

3.6.2.2.1 Técnicas Baseadas em Aceleradores de Partículas

A aplicação combinada de campos eléctricos e de campos magnéticos com o propósito de fornecer energia a partículas com carga eléctrica, produz nestas um efeito de aceleração. Este processo designado por aceleração de partículas é usado nos mais diversos domínios de aplicação, incluindo no estudo dos efeitos da colisão de partículas de alta energia na estrutura semicondutora de circuitos electrónicos.

O ciclotrão (*cíclotron*) é um equipamento que pode ser usado para este efeito. A sua aplicação consiste na geração de um feixe de partículas orientado para zonas vulneráveis do circuito objecto de avaliação. A incidência deste feixe num circuito, a que previamente foi removida parte do encapsulamento e colocado em

câmara de vácuo, produz na sua estrutura SEU's, que se traduzem em inversões de bits (Fig. 3.9).

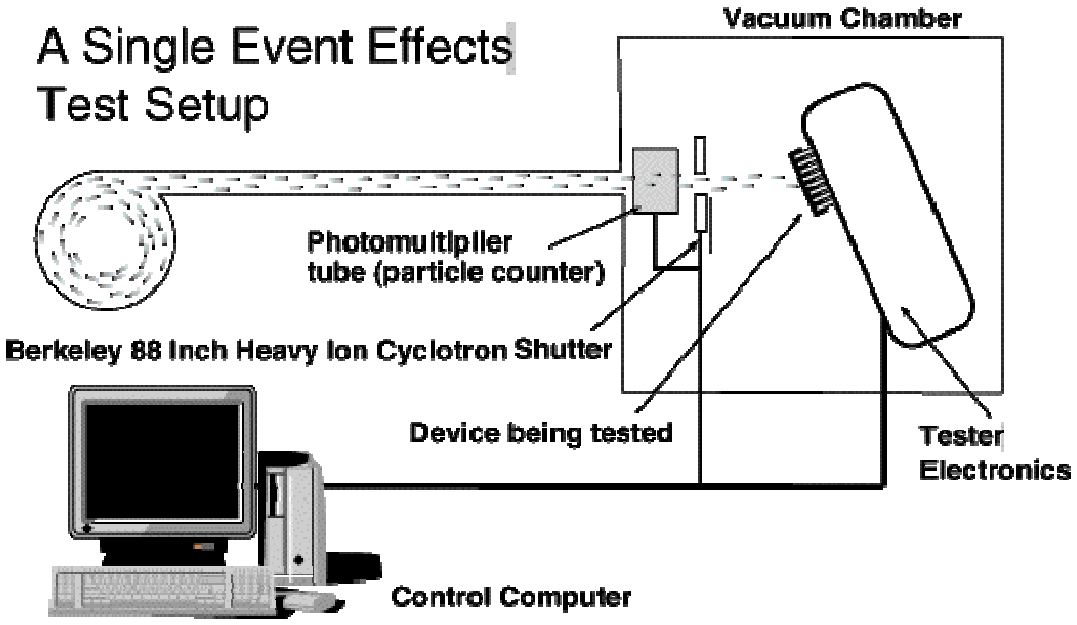

Figura 3.9 - Bombardeamento da estrutura semicondutora por partículas de alta energia.

Sustentado num panorama de grande desenvolvimento da indústria aeroespacial, com vertentes quer comerciais, como sejam a utilização de satélites de comunicações, quer para fins científicos, de que é exemplo a construção da estação espacial internacional, tem surgido uma cada vez maior solicitação de sistemas baseados em microprocessadores, cuja arquitectura permita elevado desempenho na execução de operações complexas, e sobretudo que tenham funcionamento seguro, mesmo neste tipo de ambiente.

Entre essas arquitecturas encontra-se a do microprocessador ERC32, um processador de 32 bits desenvolvido pela Agencia Espacial Europeia. O ERC32 baseia a sua arquitectura no processador Cypress CY601 SPARC V7, e inclui mecanismos para detectar e isolar erros causados por SEU [Gaisler97]. Integrado num processo evolutivo de cariz tecnológico foi desenvolvido o LEON-FT, um processador também ele de 32 bits tolerante a faltas que implementa o conjunto de instruções SPARC V8. O processador foi concebido para tolerar erros causados por SEU, com recurso a mecanismos de tolerância a erros baseados em técnicas nas quais é implementada: redundância modular tripla em registos (TMR - *Triple Modular Redundancy*), detecção e correcção de erros no próprio circuito, protecção de integridade de dados por bits de paridade, etc. [Gaisler02].

Estas arquitecturas foram submetidas a um processo de validação baseado na realização de experiências de injecção de faltas, com o objectivo de avaliar a forma como os seus mecanismos detectam e toleram a ocorrência de SEU's. No processo de validação foram utilizadas ferramentas de injecção de faltas tendo sido o ciclotrão escolhido como configuração para o injector.

As experiências demonstraram a eficiência dos mecanismos implementados, no processador ERC32. No caso do LEON-FT as experiências mostraram a

validade da arquitectura implementada no processador na detecção de erros, embora tenham sido detectadas algumas anomalias quando o processador era exposto a um elevado fluxo de partículas.

Estes dois casos mostram a importância e a aceitação da técnica de aceleração de partículas para este tipo de análise. Contudo, a utilização do ciclotrão como injector de faltas, coloca algumas dificuldades na execução das experiências. Estas dificuldades estão relacionadas com o facto da técnica não garantir o controlo sobre o processo de injecção. Significa isto, que não é possível assegurar que a falta é efectivamente injectada, dado que a ocorrência de um SEU depende da energia das partículas e do consequente processo de ionização. Da mesma forma o controlo temporal e o local onde os eventos ocorrem também é de difícil controlo. Para suprir estas dificuldades é utilizada uma técnica semelhante à referida na técnica de radiação electromagnética, em que os dados da experiência são obtidos por comparação com o funcionamento de um circuito de referência, que executa de forma síncrona a mesma aplicação (*software*).

3.6.2.2.2 Técnicas Baseadas em Fontes Radioactivas

A utilização de aceleradores de partículas, como injectores de faltas, é um processo dispendioso quando comparado com outras técnicas de radiação. Uma técnica muito mais favorável do ponto de vista económico consiste na utilização de uma fonte radioactiva.

A aplicação desta técnica pode apresentar algumas restrições, quando o objecto de estudo se prende com a análise da susceptibilidade dos circuitos à ocorrência de SEU, uma vez que nem todos os semicondutores são igualmente sensíveis a esta fonte de radiação, e o tipo de partículas em jogo pode não ser representativo das várias partículas que levam à ocorrência de SEU. Não obstante, esta limitação, a sua utilização numa perspectiva de exercitação dos mecanismos de detecção de erros e avaliação da confiança no funcionamento, não é influenciada de forma significativa por esses factores [Karlson94].

Uma fonte radioactiva que pode ser aplicada para esta função e que se encontra disponível comercialmente é o *Californium*-252 (Cf^{252}). A sua aplicação como injector de faltas processa-se através da radiação directa de partículas sobre o circuito objecto de avaliação. A fonte radioactiva é inserida numa câmara selada, na qual, por uma questão de aumento de eficiência do processo, é produzido vácuo, e o encapsulamento do circuito é removido, ficando assim exposto às partículas de alta energia provenientes da fonte radioactiva [Gunneflo87] [Gunneflo89]. De forma a permitir que as experiências decorram de forma controlada, são também implementados mecanismos na câmara, que permitem variar ou mesmo anular a quantidade de partículas a que o circuito é exposto (Fig. 3.10) [Karlson94].

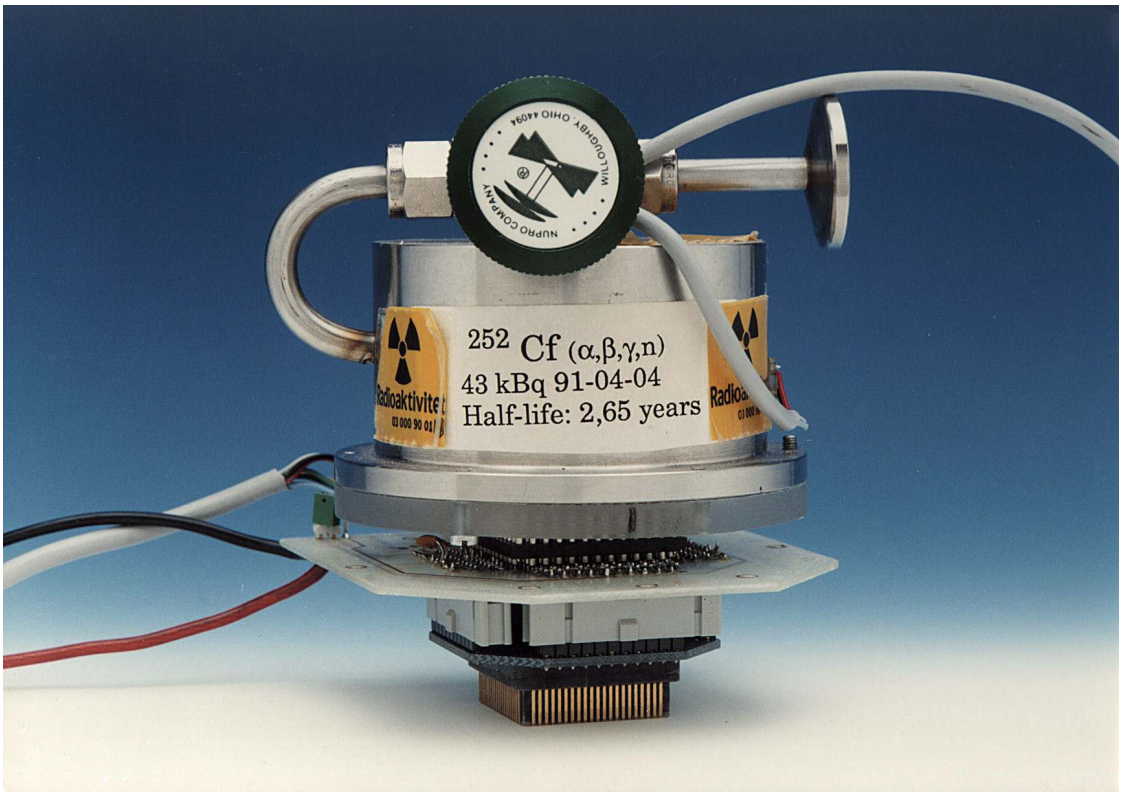

Figura 3.10 - Câmara de vácuo para injecção de faltas através de partículas radioactivas.

A ferramenta FIST (*Fault Injection system for Study of Transient fault effects*) desenvolvida na Universidade de Chalmers (Suécia) aplica esta técnica, e foi utilizada com dois propósitos.

Um, o de analisar a viabilidade da sua aplicação à injecção de faltas no interior de circuitos electrónicos. Neste sentido foi realizado um conjunto de experiências no microprocessador MC6809E da Motorola, tendo sido verificado que as faltas são injectadas por todo o circuito, e apresentando como resultado uma grande diversidade de erros, com a correspondente exercitação dos vários mecanismos de detecção de erros do processador. Estes resultados revelam características que conferem à técnica uma grande utilidade na avaliação da confiança no funcionamento de sistemas, nomeadamente de aplicações baseadas em microprocessadores. Foi também possível identificar alguns problemas associados à técnica, nomeadamente relacionados com questões de segurança, que derivam da manipulação de uma fonte radioactiva. Desta forma, foram sugeridas configurações para que a fonte radioactiva seja incorporada dentro do encapsulamento do circuito integrado que se pretende testar [Karlsson94].

Consubstanciado no facto de ter sido demonstrada a sua utilidade para proceder à injecção de faltas no interior de circuitos, esta técnica foi utilizada com um segundo propósito, o de meio complementar de avaliação de uma arquitectura de comunicação de tempo-real tolerante a faltas. Arquitectura essa que faz parte integrante do sistema MARS (*Maintainable Real-Time System*) [Kopetz89] [Arlat03] [Karlsson95] [Folkesson99].

3.6.2.3 Injecção de Faltas por Radiação LASER

A tecnologia LASER tem se afirmado como alternativa às técnicas baseadas em aceleradores de partículas e fontes radioactivos para gerar SEU e SEL no interior de circuitos integrados. A utilização de impulsos de feixes de LASER na estrutura semicondutora, apresenta-se como menos dispendiosa e consideravelmente menos destrutiva que os métodos baseados em aceleradores de partículas, e não requer a utilização de câmaras de vácuo e de protecções especiais contra a exposição à contaminação radioactiva [Buchner90].

Outra vantagem associada à utilização desta técnica reside no facto das dimensões do feixe permitirem provocar SEU's em localizações bem precisas no circuito. Esta característica aumenta de forma significativa o controlo espacial do processo de geração destes eventos. O impulso de LASER pode também ser sincronizado com sinais eléctricos do circuito, como por exemplo sinal do relógio ou sinais de controlo, o que evidencia importantes propriedades para o controlo temporal das experiências. Estes factores permitem assim conferir um elevado grau de repetibilidade das experiências que são suportadas por esta técnica.

Quando o domínio de estudo é centrado na análise da sensibilidade do circuito electrónico a fenómenos naturais como os que decorrem do impacto de partículas de elevadas energia, devem porem ser consideradas algumas limitações desta técnica. Estas limitações estão associadas ao facto do processo físico produzido pelo LASER não ser o mesmo que decorre da ionização provocada pela colisão de partículas de elevada energia. O LASER não penetra nas zonas de metalização como os feixes de partículas. Também não está identificada de forma precisa uma relação entre a energia dos impulsos de LASER e a energia dos feixes de partículas, que defina o limiar de energia necessária para produzir eventos do tipo SEU e SEL [Moss95]. Da mesma forma que no caso da técnica baseada em fontes radioactivas, estas limitações não assumem uma grande relevância, quando a técnica é utilizada numa perspectiva de validação da arquitecturas, desde que as faltas introduzidas sejam representativas dos cenários que se pretendem analisar.

A incorporação da técnica LASER em injectores de faltas foi demonstrada através do desenvolvimento de uma ferramenta na Universidade da Florida do Sul [Samson98]. A ferramenta é constituída por um LASER que é montado sobre uma mesa de translação que permite movimentos segundo 6 graus de liberdade em relação ao circuito objecto de avaliação. O controlo do LASER e da mesa de translação é completamente automatizado (Fig. 3.11).

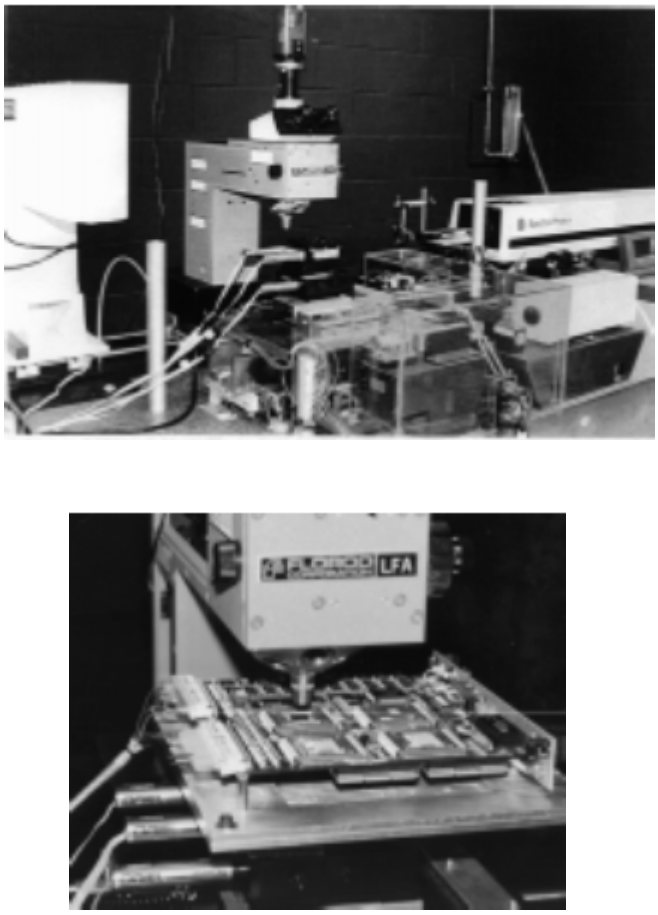

Figura 3.11 - Infra-estrutura de injecção de faltas baseadas na técnica de radiação LASER.

Para efectuar a validação dos mecanismos tolerantes a faltas ou a validação da aplicação suportada pelo circuito integrado objecto de avaliação, a infra-estrutura de injecção de faltas deve, entre outras funções, identificar a parte do circuito que irá ser perturbada e o local exacto onde a falta é injectada. Estas funções são efectuadas através do recurso a um sistema de CAD (*Computer Aided Design*). Com base num desenho que descreve na totalidade a implementação do circuito integrado, são geradas coordenadas para a mesa de translação. Após posicionado o LASER injecta a falta através de um pequeno impulso com a energia suficiente para produzir a falta na posição desejada.

A validade desta técnica foi demonstrada através de um conjunto de experiências nas quais foi comprovado que o método é exequível, e realçada a elevada repetibilidade das experiências efectuadas por esta técnica. Foram também apontadas algumas direcções para futuros desenvolvimentos da ferramenta, nomeadamente no sentido do aumento da automatização das experiências e sua adaptação a uma gama heterogénea de circuitos, de forma a tornar esta técnica uma ferramenta prática na validação de sistemas tolerantes a faltas [Samson98].

3.6.3 Injecção de Faltas com Suporte a Instrumentação no Próprio Circuito.

O desenvolvimento verificado na área da micro electrónica tem-se alicerçado num cada vez maior nível de integração. Actualmente muitos dos microprocessadores, microcontroladores, ou circuitos para aplicações específicas, integram num único circuito um conjunto de interfaces, memória e blocos funcionais, que lhes permitem assegurar muitas das funcionalidades dos actuais sistemas. Por outro lado, as necessidades de um cada vez menor ciclo de desenvolvimento, requer ferramentas que permitam suportar o inerente aumento de complexidade destes sistemas. Desta forma muitos circuitos de última geração têm um desenho orientado à testabilidade.

Numa perspectiva de adaptação a esta evolução, têm surgido técnicas de injecção que fazem uso deste suporte para teste, ou mesmo da facilidade de reprogramação concedido por dispositivos que permitem um rápido desenvolvimento de protótipos.

Duas técnicas que partilham este conceito são: a injecção de faltas na cadeia de registos de teste e diagnóstico (*scan chain fault injection*) e injecção de faltas em sistemas baseados em FPGA (*FPGA based fault injection*).

3.6.3.1 Injecção de Faltas na Cadeia de Registos de Teste e Diagnóstico

O aumento da complexidade dos circuitos integrados coloca várias dificuldades nomeadamente ao nível dos testes físicos, e de teste ao nível da aplicação. Os primeiros estão relacionados com a verificação de placas de circuito impresso, ou do funcionamento de circuitos com elevada complexidade, suportados em técnicas de montagem superficial de elevada densidade. Os segundos estão ligados à dificuldade de efectuar o diagnóstico (*debug*) em tempo-real das aplicações suportadas por este tipo de circuitos.

De forma a facilitar o teste dos circuitos foi definida uma interface segundo a norma IEEE1149.1. A norma define um conjunto de hardware que deve ser incluído em cada circuito integrado para permitir:

- Teste das ligações do circuito após a sua instalação na placa de circuito impresso;
- Teste do funcionamento do circuito;
- Observar ou modificar a actividade do circuito durante a sua operação.

O hardware consiste em cadeias de registos (*bondary–scan register*) e outros blocos funcionais que são acedidos através de uma porta denominada *Test Acess Port* (*TAP*) (Fig. 3.12) [IEEE01].

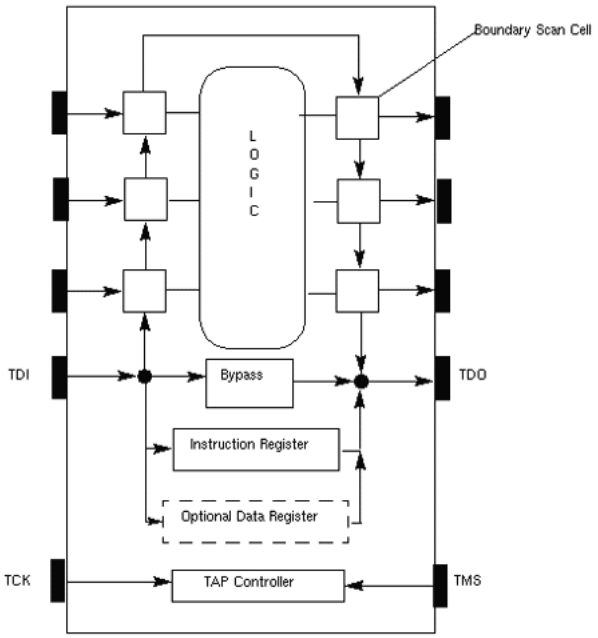

Figura 3.12 - Interface IEEE 1149.1 (JTAG).

A *TAP* é uma porta série síncrona, com uma configuração base de quatro terminais, 3 dos quais são entradas e o restante é uma saída, cujos nomes e funções são as seguintes:

- ***TCK***: serve de entrada ao sinal de relógio que define a frequência de deslocamento dos dados nos registos, e de forma síncrona com o seu sinal, toda a evolução do estado da *TAP*;
- ***TMOD***: serve de entrada aos comandos de selecção de modo, utilizado para configuração e controlo da porta;
- ***TDI***: entrada de dados da cadeia de registos da TAP;
- ***TDO***: saída de dados da cadeia de registos da *TAP*. Os dados são transferidos numa relação de um bit na entrada (TDI) para um na saída (TDO), por cada ciclo de TCK.

A cadeia de registos que se encontra associada à entrada *TDI* e saída *TDO* instrumenta cada nó, (pino), e é controlado pela *TAP*. A cada nó está associada uma macro célula com capacidade de adquirir o estado do nó, e de seleccionar através de *multiplexer* o valor que deve ter esse pino, ou seja se o seu valor deve reflectir o valor real, ou outro, que foi enviado pela cadeia de registos.

Embora esta porta tenha sido especificada para teste, ela possui algumas características que permitem o seu uso em outras funções, como por exemplo a de injecção de faltas. Uma primeira vantagem da utilização desta porta na injecção de faltas está relacionada com a dispensa da utilização de complexas ligações, que possibilitem o acesso aos pinos, nomeadamente de circuitos integrados com elevado número de entradas e saídas. Permite alterar e capturar o estado dos pinos, sem interferir no normal funcionamento do sistema.

Apesar da utilização desta porta apresentar algumas vantagens para o processo de injecção, o facto dela não ter sido especificada com objectivo de injecção de faltas, faz com que apresente naturais limitações na aplicação para este efeito. A injecção de faltas nos pinos, obriga ao envio de comandos e dados de injecção, que têm que ser serializados à frequência de trabalho imposta por TCK. Isto coloca questões de resposta temporal deste método, principalmente se a cadeia de registo for extensa. Outro inconveniente reside no facto de não ser possível trabalhar de forma assíncrona o estado dos pinos, ou seja, são enviados dados com o estado para a totalidade da cadeia endereçada pela TAP. Não é desta forma possível modificar o estado de um pino e deixar os restantes com o seu estado inalterado. Algumas das limitações da utilização da cadeia de registos para teste e diagnóstico em injecção de faltas são apresentadas em [Ke96], [Dostie95], sendo também apresentadas algumas configurações alternativas para melhorar o seu desempenho para esta aplicação

Estendendo o conceito da cadeia de registos de teste ao interior do circuito integrado é possível instrumentar as suas partes mais importantes e assim ter acesso a registos da CPU, barramento de dados, barramento de endereços, sinais de controlo, interrupções, etc. Desta forma, esta porta é muitas vezes utilizada para suportar diagnóstico (*debug*) em tempo-real de forma não intrusiva. A Figura 3.13 apresenta esquematicamente a instrumentação de um core da ARM e de todos os restantes periféricos utilizados na implementação de um ASIC.

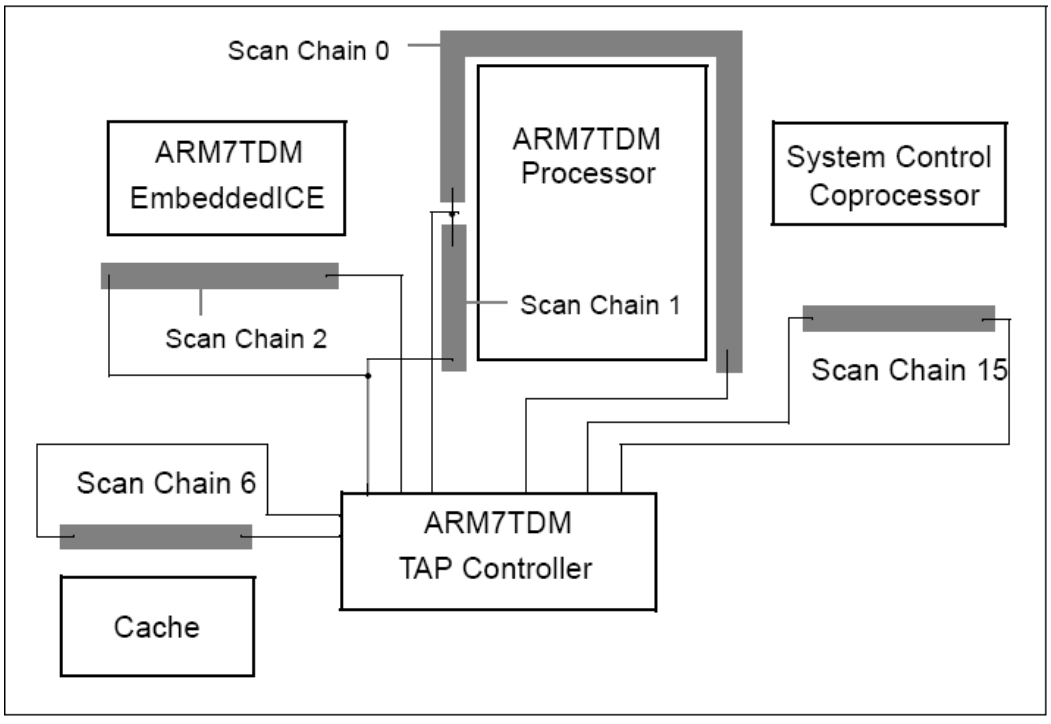

Figura 3.13 - Instrumentação de um circuito de elevada complexidade.

A aplicação desta técnica à injecção de faltas foi efectuada na Universidade de Chalmers, através do desenvolvimento de uma ferramenta, denominada FIMBUL (Fault *Injection and Monitoring using BUilt in Logic*) [Folkesson99] [Folkesson98]. A ferramenta utilizou a TAP de um processador com domínio de

aplicações na indústria espacial, o *Thor* da SAAB, para injectar faltas do tipo inversão de bit em vários locais do interior do processador, que estão acessíveis através das cadeias de registos de teste. A execução temporal das faltas é controlada através de disparos efectuados por *break-points* usando para tal registos de diagnóstico do processador. Os resultados das experiências foram comparados com outros, obtidos por simulação, utilizando um modelo VHDL detalhado do processador *Thor* na ferramenta MEFISTO-C. Com base nessa comparação foi demonstrado que a técnica injecção de faltas na cadeia de registos de teste e diagnóstico é cerca de 100 vezes mais rápida, produzindo resultados semelhantes.

3.6.3.2 Injecção de Faltas em Sistemas Baseados em FPGA

O aparecimento de dispositivos de lógica programável com capacidade para suportar milhar de portas lógicas, veio tornar exequível a implementação de sistemas complexos nestes dispositivos. As FPGA's fazem parte destes circuitos programáveis e aliam à elevada densidade de portas lógicas, outras características que as transformam em circuitos extremamente versáteis. Estas características incluem a fácil reprogramação, existência de ferramentas de suporte à sua programação baseadas em linguagem de descrição de hardware amplamente difundidas como é o VHDL. Desta forma, as FPGA's tornaram-se numa opção económica quer na implementação de circuitos, quer como plataforma de teste na fase de produção de dispendiosos circuitos de aplicação específica ASIC's.

A versatilidade evidenciada pelas FPGA permite também a sua utilização na validação de mecanismos de tolerância a faltas, ou de aplicações que correm sobre uma réplica do sistema real implementado na FPGA. A injecção de faltas em FPGA's utiliza o conceito da instrumentação no próprio circuito para suporte à injecção.

Com recurso a uma descrição detalhada do sistema é possível com esta técnica utilizar vários tipos de modelos de faltas, e injectar faltas de forma não intrusiva em praticamente qualquer lugar do circuito. Várias abordagens são possíveis:

- As faltas são introduzidas no código VHDL, modificando-o nos pontos onde as faltas são injectadas. Neste processo a FPGA é programada para cada conjunto de faltas pretendidas [Antoni00]. Neste método as experiências estão geralmente limitadas ao modelo forçar a (*stuck at*). Dada a necessidade de recompilação e programação da FPGA, para cada experiência, este método consome bastante tempo reduzindo a eficiência global do processo.
- Recorrendo a mutações na estrutura lógica. Estas mutações são introduzidas no código VHDL mantendo-se inactivas durante o funcionamento normal do circuito, podendo ser activadas em tempo-real através de sinais de controlo. Este processo não requer a recompilação do código VHDL, para cada experiência, melhorando de forma significativa o desempenho temporal do processo [Antoni00].

Contudo, para que seja possível suportar este tipo de funcionamento, é requerido um número significativo de portas, reduzindo os recursos da FPGA, o que desta forma limita a dimensão dos circuitos que podem ser implementados.

- Através da implementação de instrumentação auxiliar, criando uma interface que permite o acesso aos diversos locais de injecção (registos, barramentos, etc). Pelo recurso a esta interface é possível o envio de comandos para a FPGA, que são descodificados, e em função destes, procede-se à injecção da falta correspondente [Civera01], [Baback04].

Uma aplicação desta técnica à injecção de faltas num microprocessador é apresentada em [Civera01]. Neste trabalho, um processador da família 8051 é implementado numa FPGA e instrumentado de forma a possibilitar o acesso, e alteração dos sinais em locais com relevância para a análise. As faltas são injectadas em resposta a comandos enviados para a interface que dá acesso à instrumentação implementada na FPGA. Esta aplicação permite mostrar a versatilidade da utilização das FPGA's como método de emulação de sistemas bem como das suas potencialidades para aplicações de injecção de faltas.

3.7 Injecção de Faltas por *Software*

A injecção de faltas por software é uma técnica que tem vindo a ganhar expressão na validação de sistemas físicos. A sua atractividade está associada à flexibilidade, e ao facto desta não exigir hardware específico, por vezes de difícil implementação, e substancialmente mais dispendioso.

O modo de operação desta técnica, consiste na introdução de alterações que corrompam a informação associada a dados e/ou código do sistema, emulando assim erros resultantes de faltas físicas, ou mesmo de erros de concepção do software.

As principais limitações desta técnica estão associadas aos seguintes factores:

- As faltas não podem ser injectadas em locais que não estejam acessíveis pelo software.
- Alteração do objecto de avaliação. A injecção de faltas é um processo que envolve a execução de tarefas com requisitos temporais apertados e elevada necessidade de monitorização e recolha de dados. Assim, é necessário código adicional e afectação de recursos (CPU e memória) a estas tarefas, o que irá interferir com a utilização normal destes últimos, alterando a carga do sistema, ou eventualmente a sua estrutura.
- Fraca resolução temporal. O processo de injecção depende dos ciclos de máquina do processador, assim como, da disponibilização destes

pelo processador, o que dificulta o processo de injecção designadamente em modo assíncrono. A técnica exibe também dificuldades em monitorizar e adquirir erros com baixa latência, de que são exemplo erros nos registos da CPU [Folkesson99], [Hsueh97].

As técnicas de injecção de faltas baseadas em software podem ser classificadas de acordo com o momento em que estas são inseridas no sistema. De acordo com este índice de classificação, é possível agrupar as técnicas em duas classes. Uma em que as faltas são inseridas antes do processo de compilação do software do sistema – pré-execução (*pre-runtime*), e uma outra em que o processo ocorre durante a execução (*runtime*) [Folkesson99].

3.7.1 Técnicas de Injecção em Pré-Execução

Nas técnicas pré-execução (*pre-runtime*), antes do o sistema ser colocado em operação, o seu software é editado e modificado, para conter as faltas que irão ser activadas posteriormente. As mudanças processam-se através da alteração da informação que pode incidir quer no segmento de dados, quer no segmento de código do software. A base de trabalho para estas técnicas é o código fonte do sistema, ou quando este não está disponível, as faltas poderão ser introduzidas no código máquina.

Desta forma é possível emular erros no software, falhas de hardware, ou de faltas transitórias, sendo necessário para tal, descarregar uma imagem que inclua os erros relativos aos cenários que se pretendem analisar.

Esta abordagem à injecção de faltas, tem na minimização do efeito de intrusão, característico das técnicas de injecção por software, a sua principal virtualidade, uma vez que não requer a utilização de software adicional para suportar a sua realização.

A infra-estrutura de injecção de faltas DOCTOR (*integrateD sOftware fault injeCTiOn enviRonement*) foi desenvolvida na Universidade de Michigan (EUA), e integra o conceito de injecção de faltas em pré-execução. A infra-estrutura utiliza como injector a ferramenta SFI (*Software Fault-Injector*), que foi utilizada na validação do sistema distribuído de tempo-real HARTS [Han95], [Rosenberg93]. Esta ferramenta permite a injecção faltas transitórias, intermitentes ou permanentes em locais como: posições de memória; registos da CPU; e nas comunicações.

A técnica pré-execução é utilizada para a injecção de faltas na CPU. Desta forma, as faltas são previamente inseridas no código do programa que serve de carga ao cenário de avaliação. A mesma ferramenta utiliza uma abordagem que está acordo com a técnica de injecção durante a execução (*runtime*), para proceder à injecção de faltas nas comunicações e na memória do sistema.

3.7.2 Técnicas de Injecção Durante a Execução

A técnica de injecção durante a execução (*runtime*) apresenta-se como uma solução, mais versátil que a anterior, permitindo não só, a injecção de faltas no código do software, mas também aceder a todas as outras partes do sistema que sejam abrangidas pelo endereçamento do processador, como sejam: registos da CPU, memória, e outras interfaces. Neste caso, o processo de injecção requer a utilização de módulos especializados, que terão como função a execução das tarefas de injecção. Durante a operação, estes módulos de software são activados através de mecanismos implementados no sistema e procederão à injecção de faltas.

Os mecanismos mais utilizados para efectuarem a activação dos módulos de injecção são [Hsueh97]:

- ***Timeout***. Este é o processo mais simples de activação do módulo de software de injecção de faltas. Genericamente o módulo de injecção é invocado por uma interrupção que é gerada após a expiração de um temporizador, que representa o tempo que deve decorrer até que uma falta seja injectada.

  Neste processo, o evento que desencadeia a injecção de faltas não está directamente associada a um estado específico do sistema, existindo unicamente uma relação temporal entre as faltas. Assim, é difícil sincronizar a injecção de faltas com estados do sistema, e em consequência não é possível determinar à priori qual a incidência exacta da falta. Todavia, este método permite emular a ocorrência de faltas transitórias ou intermitentes ao nível do hardware.

- ***Exception/Trap***. Este método recorre a mecanismos implementados em hardware e/ou, em software, que permitem comutar a execução do programa aquando da ocorrência de determinados eventos. Assim, sempre que ocorra um evento associado a estes mecanismos, é produzida uma interrupção que transfere a execução do programa para os módulos responsáveis pela injecção de faltas. Ao contrário do *timeout*, este método permite injectar faltas em resposta a estados do sistema. Exemplo disso é a possibilidade do mecanismo efectuar disparos, quando uma determinada posição de memória é acedida (*hardware*), ou antes que uma determinada instrução seja executada (*software*).

- ***Code insertion***. Este método tem uma filosofia semelhante à implementada nas técnicas pré-execução, permitindo a injecção de faltas antes da execução de uma determinada instrução. Contudo difere quanto à forma de implementação. Na primeira as faltas são injectadas através da modificação do código. No método de inserção de código, a instrução que efectua a injecção de faltas é inserida durante a execução do programa. Desta forma, e ao contrário do método de *exception/trap*,

o software de injecção de faltas tem que fazer parte da aplicação do sistema.

Desde as primeiras abordagens à injecção de faltas, através técnicas de software, de que a ferramenta FIAT (*Fault Injection Based Automated Testing Environment*) é um dos seus resultados, tem-se verificado uma ampla utilização desta técnica para análise da confiança no funcionamento de sistemas [Segall88]. A acompanhar esta dinâmica, uma grande diversidade de ferramentas de injecção tem sido desenvolvida. As ferramentas, *FERRARI*, *FINE*, *DEFINE*, *FTAPE*, *Xception*, *MAFALDA*, são alguns exemplos de ferramentas baseadas na aplicação desta técnica.

A ferramenta *FERRARI* (*Fault and Error Automatic Real-time Injection*) foi desenvolvida na Universidade Texas (EUA), e recorre à função *ptrace* do sistema operativo UNIX, de forma a modificar a imagem do processo e inserir *software trap'*s [Kanawati92]. Estas são usadas para suportar a injecção de faltas, nos barramentos, na memória e na CPU. São activadas através da alteração do *program counter,* quando este atinge localizações específicas do código, ou através temporizadores (*timeout*), para emular a ocorrência de eventos transitórios e intermitentes.

A Universidade Illinois (EUA) tem um considerável historial de desenvolvimento de ferramentas de injecção de faltas, e em particular daquelas que baseiam o seu funcionamento em técnicas de software. A ferramenta *FINE* (*Fault Injection and moNitoring Environment*) recorre ao método de *software traps* para injectar faltas em sistemas operativos emulando faltas, quer ao nível do hardware, quer ao nível do software [Kao93]. Esta ferramenta foi utilizada para analisar a propagação de erros em sistemas UNIX. A ferramenta *DEFINE* (*Destributed Fault Injection and moNitoring Environment*) surgiu como extensão da anterior para suportar a injecção de faltas em ambiente distribuído [Kao94]. Seguindo este contexto de evolução, a ferramenta *FTAPE* (*Fault Tolerance And Performance Evaluator*), foi desenvolvida, não somente para a análise da confiança no funcionamento, mas também como suporte à medida de desempenho (*Benchmarking*) de sistemas tolerantes a faltas, e de análise do seu desempenho na presença de faltas [Tsai96].

A ferramenta *Xception* foi desenvolvida na Universidade de Coimbra (Portugal), e baseia o seu funcionamento na utilização das facilidades de diagnóstico e de monitorização de desempenho, disponibilizadas por muitos dos modernos processadores [Carreira98]. Através, da utilização de excepções implementadas no hardware do processador, a ferramenta é capaz de injectar faltas sem recurso a *software traps,* e sem necessitar de modificar o código da aplicação. Desta forma, é minimizando o efeito de intrusão provocado pela ferramenta.

A *Xception* é capaz de suportar alguns dos mais frequentes modelos de faltas como sejam o forçar a (*stuck at*) e inversão de bits (*bit*-flip*s*). Estes são activados por eventos de natureza espacial ou temporal, que envolvem a manipulação de

dados através de acesso à memória, ou disparos de temporizadores que podem ser configurados pelo utilizador

A existência de hardware dedicado, implementado pelo próprio processador, permite a recolha de informação detalhada acerca do seu estado de operação, facilitando as tarefas de monitorização, nomeadamente as que estão relacionadas com a identificação do efeito das faltas. Desta forma, combinando a capacidade de diagnóstico (*debug*) e de monitorização é possível programar o sistema para capturar eventos, de que são exemplo a activação de erros latentes na memória.

A ferramenta *MAFALDA* (*Microkernel Assessment by Fault injection Analysis and Design Aid*) foi desenvolvida no LAAS-CNRS (França) [Fabre00]. Recorre igualmente a funcionalidades de diagnóstico, implementadas no hardware, para definir disparos com resolução temporal e espacial (código e segmento de dados), que activam a injecção de faltas.

O seu desenvolvimento teve como objectivo a avaliação do comportamento de COTS (*Commercial Off-The-Shelf*) *microkernels* na presença de faltas e de suportar a sua integração em sistemas com requisitos de elevada confiança no funcionamento. Designadamente em aplicações de segurança crítica no domínio aeroespacial, ou do controlo de sistemas de transporte ferroviário.

Neste contexto, é também analisado o grau de confinação de erros conferido pela concepção de *wrappers,* mecanismos implementados através de software extra, que têm a função de testar as funções de chamadas ao sistema, e de fornecer os resultados das mesmas, integrando-os numa estratégia de controlo de erros.

Como evolução da anterior ferramenta foi desenvolvida a MAFALDA-RT, que tem como principal motivação a avaliação da confiabilidade de sistemas de tempo-real, e que apresenta uma abordagem no sentido da minimização dos efeitos intrusivos associados à técnica de injecção de faltas por software [Rodriguez02].

Isto é alcançado através do "congelamento" da actividade do sistema durante a fase de injecção de faltas. O sistema real é substituído por um modelo de software, que emula o comportamento dos elementos externos ao processador (ex. sensores). Durante a fase de congelamento da actividade do sistema, a ferramenta assume o controlo das interrupções, nomeadamente a que se encontra associada ao relógio. Desta forma, controla a comutação entre tarefas que constituem o sistema de tempo-real, tendo assim possibilidade de congelar a evolução do estado do sistema.

Este método elimina o efeito de intrusão da ferramenta de injecção. Contudo apresenta como principal inconveniente o facto de não poder ser aplicado a um sistema físico real.

3.8 Sistemas Híbridos

Existem ferramentas que fazem uso de mais que uma das técnicas de injecção apresentadas nas secções anteriores. A utilização combinada das técnicas é efectuada numa perspectiva complementar, para melhorar características de operação que não sejam consideradas satisfatórias quando executadas de forma separada. Esta configuração é denominada de técnicas de injecção híbridas.

Várias configurações são possíveis na utilização combinada de técnicas de injecção. As mais usuais são:

- Técnicas híbridas envolvendo injecção física por hardware e software;
- Técnicas híbridas combinando injecção de faltas por software com técnicas de simulação.

Técnicas híbridas baseadas em técnicas de hardware e de software foram utilizadas na ferramenta FERRARI e LIVE. Na ferramenta FERRARI, esta configuração surgiu como uma extensão da versão anterior da ferramenta §3.72 que se baseava unicamente na injecção de faltas por software [Kanawati95]. Com a introdução da técnica híbrida foi possível melhorar o funcionamento da versão anterior da ferramenta, nomeadamente através da incorporação de mecanismos que permitem a sincronização da injecção de faltas com os eventos que ocorrem no sistema.

A ferramenta LIVE (*Low Intrusion and Validation Environment*) foi desenvolvida na Universidade Nápoles (Itália) para ser aplicada na validação de sistemas de controlo ferroviário [Amendola03]. O injector da ferramenta é programável e capaz de usar de forma complementar a técnica de injecção ao nível do pino e mecanismos de técnicas de *software*. A técnica de injecção ao nível do pino é usada para alterar o valor dos sinais dos barramentos do microprocessador. No processador são geradas interrupções para activar módulos de *software* que injectam faltas nos registos da CPU e de outros dispositivos que estão no alcance do seu espaço de endereçamento.

A utilização combinada da técnica de injecção por software, com outra baseada em simulação, é proposta em [Güthoff96]. Esta configuração melhora o desempenho da primeira no que concerne à injecção de faltas em locais inacessíveis ao software. Para tal o sistema executa a aplicação, normalmente até que a injecção de faltas é activada através de uma interrupção. Em resposta a este evento o conteúdo dos registos e posições de memória mais representativos do estado do sistema são transferidos para o simulador. Este executa um modelo detalhado do sistema, para efectuar a injecção de faltas. Após concluída a operação, o novo estado do sistema é transferido do simulador para o de sistema real.

3.9 Comparação das Técnicas de Injecção de Faltas

Uma tão grande diversidade de ferramentas, suportada em diferentes técnicas de injecção pressupõe que em cada uma delas sejam potenciadas características que confiram vantagens ao processo de avaliação. Desta forma, pode-se proceder a uma classificação qualitativa das técnicas de acordo com um conjunto de atributos que as caracterizam [Arlat03] [Folkesson99] [STSARCES00].

- **Controlabilidade**: representa a capacidade de levar a cabo as experiências de forma controlada. A controlabilidade pode ser vista segundo duas perspectivas: uma relativa à capacidade de controlar o local onde a falta é injectada – **controlabilidade espacial**; e, uma outra relativa à capacidade de controlar o instante da ocorrência da falta – **controlabilidade temporal**;
- **Reprodutibilidade**: capacidade de reproduzir as experiências, que pode corresponder à capacidade de repetição das experiências individuais (**repetibilidade**), ou que dessas experiências se obtenham resultados estatisticamente reprodutíveis;
- **Atingibilidade (*Reachability*)**: caracteriza a capacidade de aceder aos locais do circuito onde se pretende injectar as faltas;
- **Resolução temporal**: representa a capacidade para efectuar medidas temporais associadas aos efeitos das faltas, como por exemplo tempos de latência;
- **Intrusividade**: caracteriza o grau de interferência não desejável provocada pela técnica na operação do sistema objecto de avaliação.
- **Custo**: indicador de quão dispendiosa é a infra-estrutura de injecção de faltas;
- **Eficiência**: representa o esforço a despender, associado ao número de experiências que são necessárias e do tempo a elas associado.

Na tabela 3.1, é apresentada uma classificação genérica numa perspectiva qualitativa, realçando as características mais favoráveis e aquelas que são consideradas como menos favoráveis à sua utilização.

	Simulação	Hardware	Software
Vantagens	• Controlabilidade elevada. • Repetibilidade muito elevada. • *Reachability* elevada. • Custo de implementação baixo.	• Injecção de faltas como ocorre na realidade. • Controlabilidade elevada (*pin-level Built in fault injection* e radiação LASER). • Repetibilidade elevada (*pin-level*, *Built in fault injection* e radiação LASER). • *Reachability* média a alta. • Resolução temporal elevada (*pin-level*). Eficiência elevada (*pin-level*), e média para as restantes configurações.	• Injecção de faltas num sistema real. • Controlabilidade elevada. • Repetibilidade elevada. • Custo de implementação baixo. • Resolução temporal média alta.
Desvantagens	• Injecção em tempo-real impossível. • Resultados dependem do detalhe do modelo. • Eficiência baixa: tempo de execução de tendencialmente muito elevado.	• Custo de implementação elevado ou muito elevado (excepto *Built in fault injection*). • Controlabilidade baixa • Reproducibilidade baixa (EMI). • Resolução temporal genericamente baixa (radiação). • Custos de implementação tendencialmente muito elevados.	• *Reachability*: inacessibilidade a locais não endereçáveis pelo software. • Eficiência baixa: número de experiências elevadas e elevado tempo de execução. • Intrusividade: em cenários de validação de sistemas, a técnica pode alterar a configuração do software ou eventualmente a carga do sistema.

Tabela 3.1 - Características das técnicas de injecção de faltas.

3.10 Síntese

Nas secções anteriores foi apresentada uma panorâmica das técnicas de injecção de faltas utilizadas na validação e avaliação do funcionamento de sistemas. Nessa descrição fica patente a existência de uma grande diversidade de técnicas para este domínio de aplicação.

Esta diversidade é caracterizada não só pela existência de várias classes de técnicas, mas também por variantes dentro de uma mesma classe. A existência de uma grande heterogeneidade de soluções implementadas nos sistemas, quer ao nível do hardware, quer ao nível do software, é imposta por requisitos do

processo de avaliação, que na sua essência são antagónicos e consequentemente conduzem a diferentes abordagens, contribuindo assim para este panorama.

A necessidade de um ambiente de validação e avaliação do funcionamento, mais homogéneo, cuja aplicação seja possível estender a um número considerável de sistemas, e ainda que os resultados obtidos possam ser utilizados para formular comparações, nomeadamente quanto ao seu comportamento na presença de faltas, tem conduzido à procura de uma solução unificadora. A NFATE (*Network Fault Tolerance and Performace Evaluator*) é um exemplo típico do desenvolvimento de uma ferramenta que resulta desse esforço [Stott00]. A NFATE implementa uma camada de *software* que faz a separação entre o nível de injecção das faltas com a estrutura superior da arquitectura da ferramenta. Desta forma, é possível isolar aspectos de implementação de injectores e usar uma estrutura comum, nomeadamente na análise de resultados.

Este é um método vantajoso, numa perspectiva de obtenção de uma uniformização da análise de resultados e da possibilidade de reutilização de um considerável número de componentes de *software* da infra-estrutura. Principalmente quando se pretende estender a aplicação das ferramentas à avaliação de novos sistemas. Contudo, no que concerne aos injectores, e mais concretamente às técnicas implementadas, dadas as particularidades das tecnologias envolvidas e da diversidade de problemas a endereçar, o cenário não é alterado, tendo sempre de ser a estrutura dos injectores a adaptar-se aos requisitos da avaliação.

Neste contexto não é possível assumir a primazia de uma técnica em particular, devendo antes estas serem utilizadas numa perspectiva complementar. Assim, a selecção de uma técnica deve obedecer a critérios fundamentados em atributos que as diferenciam e que mais se adaptem às necessidades de análise e às particularidades de cada sistema.

Capítulo 4

Arquitectura do Sistema de Injecção de Faltas

4.1 Caracterização do Ambiente de Injecção de Faltas

Actualmente a utilização de arquitecturas distribuídas em sistemas de controlo está perfeitamente consolidada. A sua utilização tem-se generalizado em aplicações onde a resposta de tempo-real é um requisito, assim como, se tem verificado uma expansão do seu domínio de aplicação para abarcar aplicações nas quais a segurança é mandatória.

As redes de comunicações apresentam-se como um elemento estruturante destes sistemas, os quais fazem assentar muito do seu funcionamento nos serviços de comunicação. Contudo a operação destas redes está sujeita a variações que resultam geralmente da interacção com componentes ambientais associadas aos locais onde operam.

A instalação destes sistemas é efectuada muitas das vezes em locais caracterizados pela existência de fontes geradoras de radiação electromagnéticas de intensidade considerável. A necessidade de estabelecer a cobertura de comunicações entre equipamentos que se encontram dispersos, obriga os elementos constituintes da camada física do sistema de comunicações a ter de ser por vezes dispostos ao longo de extensas áreas, que são problemáticas do ponto de vista destas interferências. Assim, o barramento de comunicações é o elemento mais exposto a este fenómeno, e aquele que devido à sua extensão é mais difícil a aplicação de técnicas que permitam reduzir estes efeitos sem que a relação custo benefício não se degrade.

Estes são condicionalismos que aumentam a probabilidade das comunicações serem afectadas por erros. Erros esses, que resultam tipicamente de faltas de natureza transitória com incidência nos *transceivers* e sobretudo ao nível dos cabos que constituem o barramento de comunicações. Quando isto se verifica a normal operação do sistema é perturbada manifestando-se esta através de desequilíbrios de carga no sistema, que resultam da activação de mecanismos de recuperação de erros que acabam por adicionar carga (*overhead*) suplementar no sistema. Em virtude da importância das redes na operação global do sistema e também pelos significativos impactos que variações do seu desempenho produzem, esta componente do sistema deve ser alvo de particular atenção.

A avaliação aprofundada do comportamento das redes de comunicação requer a verificação de como reagem os protocolos a um espectro alargado de incidência que perturbem a sua operação.

A utilização de ferramentas baseadas em técnicas de injecção de faltas é um método que se adequa à avaliação do funcionamento das redes para estas condições de operação. Para que esta avaliação seja possível é necessário criar um ambiente de injecção de faltas capaz de reproduzir um conjunto de experiências que sejam representativas da operação da rede neste ambiente.

Como em qualquer ferramenta de injecção, isto pressupõe que ela seja capaz de desempenhar tarefas segundo três vectores fundamentais, ou seja, ter capacidade de injectar faltas, recolher dados relacionados com os seus efeitos, e com base nestes efectuar análises ou disponibilizá-los para esse propósito.

Num contexto de aplicação à avaliação de redes de comunicação e mais concretamente no caso a que se propõe esta dissertação, a ferramenta tem como pré-requisitos:

- Ter capacidade de alterar o conteúdo da informação que circula no barramento de comunicações, injectando faltas de natureza transitória que possam ser configuradas e sincronizadas com a unidade de informação elementar – o *bit*;
- Activar as faltas de acordo com um processo estocástico incidindo em cenários de carga que podem variar da simples existência de informação de gestão do anel (tramas de gestão) até cenários de carga elevada com requisitos de tempo-real;
- Recolher informação global do sistema, obtida ao nível do barramento de comunicações, e informação individual relativa à operação de cada estação da rede;
- Possibilitar a análise da informação, dissociada da necessidade da execução simultânea de experiências de injecção, recorrendo a um método que permita construir um capital de informação sobre o qual possam ser efectuadas diversas medidas num processo pós-experimental.

Todas estas operações decorrem em ambiente distribuído, pelo que existe uma necessidade acrescida de coordenação entre os componentes da ferramenta. Isto é importante nomeadamente para manter todo o processo sincronizado, assim como para assegurar a consistência temporal da informação recolhida. Deve igualmente ser assegurado que na execução de cada experiência o sistema seja levado para um estado inicial bem definido que permita obter resultados com relevância estatística.

4.2 Arquitectura

O desenvolvimento de um ambiente de injecção de faltas é um problema de elevada dimensão, envolvendo tarefas de cariz multidisciplinar tendo algumas delas uma complexidade considerável. Estas podem ser classificadas de acordo com a sua função primeira. Neste contexto, existem tarefas que envolvem a criação de funcionalidades base, que estabelecem os alicerces fundamentais para a operação de toda a estrutura. Essencialmente estas consistem no desenvolvimento do hardware da infra-estrutura e de software que garante sua operação.

Outras de natureza complementar estão relacionadas com todas as tarefas que asseguram que a infra-estrutura desempenha correctamente a sua função. Isto inclui a reunião dos elementos: que governam as experiências, a coordenação da estrutura base e a análise de resultados. Este processo envolve o recurso a ferramentas e metodologias que permitam construir os cenários de análise, assim como extrair os resultados das experiências.

Neste contexto, e na prossecução de um ambiente de injecção de faltas que se adeqúe aos pré-requisitos enunciados, foi desenvolvida uma infra-estrutura, cuja arquitectura assenta sobre uma estrutura modular, que opera de acordo com uma organização hierárquica [Carvalho03] [Carvalho05a]. A infra-estrutura representa a componente operacional do ambiente de injecção, ou seja, a plataforma onde irão decorrer as experiências de injecção de faltas.

A característica modular da estrutura visa ajustar-se ao ambiente distribuído em que decorrem as experiências. Os módulos são concebidos como entidades autónomas com existência física independente dos demais, desempenhando cada um uma função específica neste sistema inerentemente distribuído.

A organização hierárquica permite balancear o sistema, através da divisão das tarefas repartindo-as pelos vários recursos do sistema, conferindo-lhe assim uma resposta mais adequada aos requisitos da aplicação. Uma representação esquemática da estrutura do ambiente de injecção de faltas proposto é apresentada na figura 4.1.

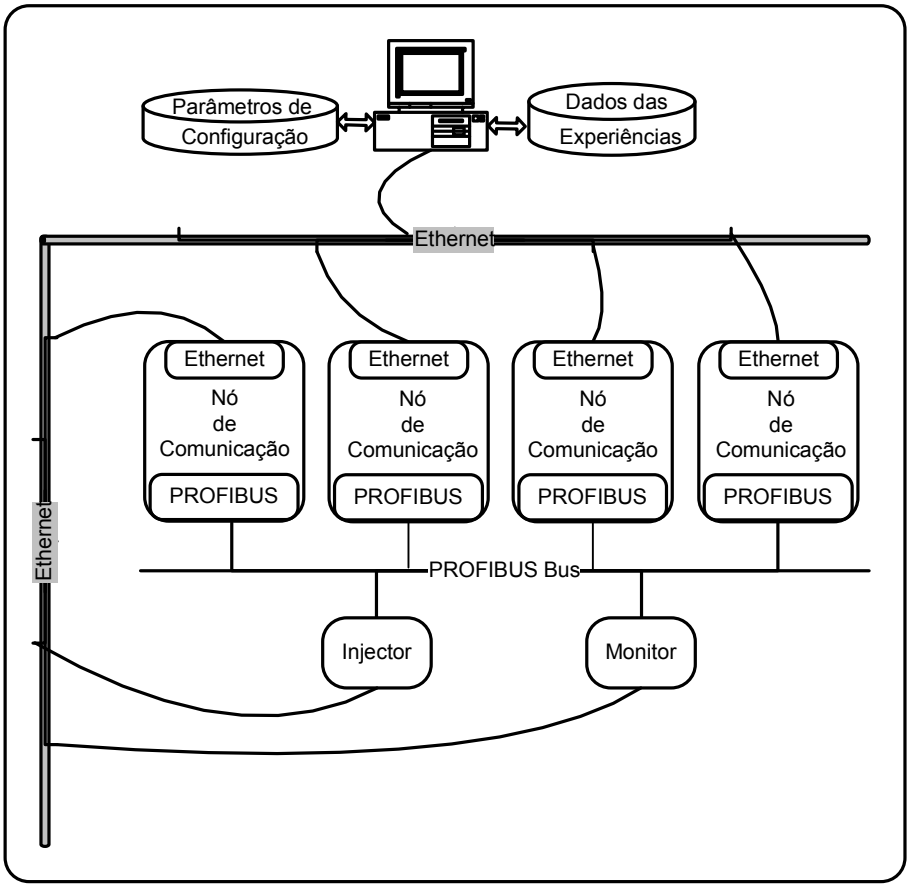

Figura 4.1 - Arquitectura da infra-estrutura de injecção de faltas.

A estrutura é constituída pelos seguintes módulos:

Infra-estrutura de Comunicações: que recria o ambiente de comunicações objecto de avaliação, sendo constituída por uma rede real onde os nós de comunicação estão de acordo com a norma IEC61158;

Injector: desempenha as funções necessárias à injecção de faltas na infra-estrutura de comunicações;

Monitor: responsável pela aquisição do estado global do sistema, através da recolha dos eventos relevantes que ocorrem no barramento de comunicações;

Unidade de gestão: desempenha as tarefas de gestão reunindo as condições que viabilizam a execução das experiências, assim como, armazena a informação que resulta dessas mesmas experiências. Desempenha igualmente, no nível hierárquico superior, um papel de coordenação de toda a infra-estrutura.

A infra-estrutura de comunicações, o injector e o monitor, representam a parte nuclear da infra-estrutura, suportando as funcionalidades base do ambiente de injecção de faltas. A unidade de gestão agrega as funcionalidades complementares, sendo estas implementadas na sua estrutura ou através dos serviços fornecidos por ferramentas, para as quais comporta interfaces que permitem a troca de informação entre ambas.

Ao nível das estruturas base de suporte, ao ambiente de injecção de faltas são requeridas consideráveis componentes de hardware e de software. No intuito de manter a diversidade de hardware a um nível reduzido, e simultaneamente assegurar um conjunto significativo de funcionalidades, foi desenvolvida uma configuração base de hardware, que assenta numa filosofia que tem subjacente a minimização das restrições à sua utilização. Esta configuração de hardware deve assim satisfazer as necessidades fundamentais dos três módulos da estrutura base, devendo as partes não suportadas ou que requeiram uma operação mais especializada ser implementadas em hardware específico que opere subordinado à configuração base.

Desta forma, para além da natural capacidade de processamento foi dado um especial enfoque ao projecto de interfaces que permitam conferir um elevado grau de versatilidade a estes módulos (Fig. 4.2).

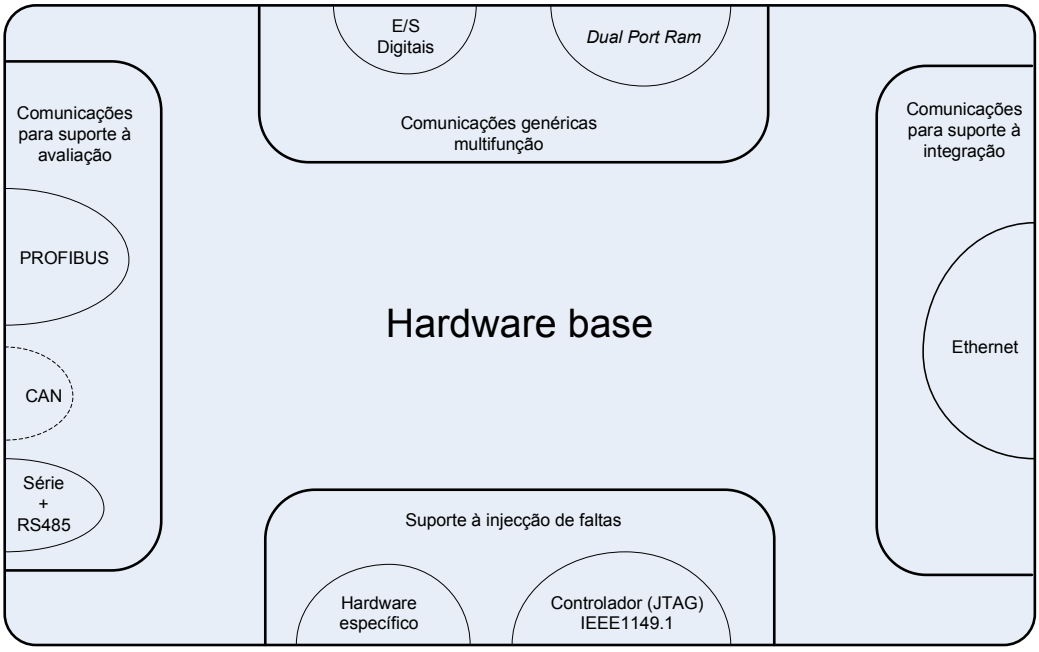

Figura 4.2 - Funcionalidades suportadas pela configuração base de hardware.

Neste contexto, o desenvolvimento do hardware fundamentou-se nas seguintes linhas orientadoras:

- Facilitar a expansão de funcionalidades, através da possibilidade de operação em coordenação com hardware externo;
- Suportar comunicações cuja aplicação esteja orientada à avaliação de redes de campo e em particular ao PROFIBUS-DP;
- Desenvolver interfaces que facilitem as tarefas de integração de toda a estrutura;
- Capacidade para disponibilizar suporte à injecção de faltas.

4.2.1 Comunicações Genéricas Multi-Função

A capacidade de um sistema baseado e microprocessadores, não pode ser unicamente caracterizada pelo poder de cálculo ou pela rapidez com que estas unidades efectuam as operações. Um sistema é algo mais complexo que pressupõe uma interacção com a sua envolvente. Desta forma, têm de existir meios que permitam a essas unidades obter informação e actuar nos objectos que estão na sua esfera de controlo. Isto requer o estabelecimento de interfaces de comunicação por mais elementares que elas sejam.

Com o objectivo garantir o acesso ao exterior do módulo base, nomeadamente para interagir com hardware externo, a configuração base foi provida de uma interface de comunicação paralela configurável. Esta consiste num conjunto de entradas e saídas que podem ser utilizadas segundo dois modos de operação distintos:

- Entradas e saídas configuráveis, com possibilidade de utilização independente ao bit, em acesso assíncronos de escrita ou de leitura. Esta interface pode ser igualmente configurada para trabalhar de acordo com requisitos de outras interfaces paralelas assegurando a conectividade com o exterior.
- Acesso directo ao conteúdo de memória do sistema por intermédio de *Dual Port Ram*. Através da utilização desta interface é possível utilizar dois módulos numa configuração mestre-escravo expandindo as suas capacidades de forma significativa. De igual modo é possível estabelecer comunicações com equipamentos providos de interfaces amplamente difundidas como seja a interface PC-ISA.

4.2.2 Comunicações para Suporte à Avaliação

O desenvolvimento de uma infra-estrutura de injecção de faltas, do tipo proposto nesta dissertação, requer a utilização de sistemas de comunicação, baseados em nós que implementem os protocolos da rede objecto de avaliação. Tipicamente, os equipamentos que utilizam este tipo de rede estão associados a domínios de aplicação específicos, sendo portanto muitas das vezes soluções proprietárias cuja implementação não permite o acesso a muita da informação relevante para o tipo de análise proposta. Normalmente, estes equipamentos são constituídos por autómatos programáveis que permitem simplesmente o acesso a funções de programação do equipamento, ou por vezes através de módulos de entradas e saídas que geralmente não permitem qualquer tipo de acesso ao subsistema de comunicações.

Assume assim, particular importância o desenvolvimento de nós de comunicação que permitam obter informação nos vários níveis da camada do protocolo.

De forma a poder efectuar uma avaliação dos efeitos das faltas no sistema, e estando estes contidos em informação que é adquirida em diferentes pontos da infra-estrutura é necessário garantir a coerência temporal entre a operação dos intervenientes no processo. Assim, a sincronização da infra-estrutura é um importante requisito. Esta sincronização é necessariamente efectuada através de mecanismos assentes na troca mensagens entre módulos. Isto obriga ao recuso a comunicações que garantem baixo valor em parâmetros com a latência e o *jitter*, em serviços de difusão (*broadcast*), ou para os quais o comportamento relativo a estes parâmetros esteja bem identificado à priori.

Neste contexto, é importante disponibilizar na estrutura base de hardware interfaces de comunicação que suportem comunicações de tempo-real, nomeadamente aquelas que implementem os protocolos da rede objecto de avaliação.

Para que os nós de comunicação possam utilizar a configuração base de hardware como estrutura de suporte à sua operação, no seu hardware foi incluída uma interface PROFIBUS-DP, cujo *driver* foi configurado para a solução típica constituída por *driver* RS-485 em conjunção com cabo coaxial.

Outras interfaces de comunicação foram igualmente implementadas no hardware da configuração base. Nestas, inclui-se uma porta série com capacidade para suportar comunicações com elevadas taxas de transmissão, configurada para operar em barramentos RS-485. Esta configuração pode ser utilizada em várias aplicações. Contudo, o principal objectivo da sua implementação está relacionado com a possibilidade de assegurar a observação de estado em barramentos RS-485, como aquele que é implementado pelo PROFIBUS-DP.

Enquadrado pela filosofia não restritiva que esteve subjacente ao desenvolvimento da estrutura base de hardware, e embora não se enquadrando nos propósitos propostos nesta dissertação, optou-se igualmente por disponibilizar suporte para CAN. Neste contexto, são disponibilizadas ligações para duas interfaces de CAN. Desta forma, através do recurso a circuitos de *driver* externos, é possível suportar comunicações de protocolos assentes numa das redes em que se tem verificado uma aceitação e utilização ascensional.

4.2.3 Comunicações para Suporte à Integração

O ambiente de injecção tem características inerentemente distribuídas, e como tal, a grande maioria das operações dos seus módulos está estruturada na troca de informação. Essa troca engloba fluxos de informação importantes, quer no sentido ascendente, quer no sentido descendente da estrutura hierárquica, nos quais está envolvida informação de configuração, de coordenação e armazenamento dos resultados, associados às experiências de injecção de faltas.

A comunicação entre componentes da infra-estrutura assume assim um papel de relevo, que deve ser correctamente considerado no hardware da estrutura base. Numa perspectiva não só de estabelecimento de canais de comunicação entre

componentes da estrutura, mas também como um meio que facilite a sua coerente integração no sistema foi implementada na estrutura base de hardware uma interface de comunicação ethernet.

A ethernet é uma tecnologia amplamente difundida, suportada por um leque bastante amplo de equipamentos, e com uma diversidade de protocolos sobre os quais são disponibilizados poderosos serviços de comunicação. Estas características contribuem de forma significativa para facilitar o processo de integração da estrutura.

4.2.4 Suporte a Injecção de Faltas

A injecção de faltas é uma das tarefas que mais condicionalismos impõe ao sistema onde está implementada, e que simultaneamente mais dependente está do desempenho deste, para cumprir com os requisitos impostos pelo processo de injecção. Da mesma forma, a configuração do injector está fortemente dependente dos atributos que se considerem relevantes para o processo de injecção, nomeadamente no que concerne ao que se pretende avaliar, como se processa a injecção, e, os locais onde as faltas são injectadas.

Este panorama não permite muita margem para os injectores acomodarem modificações e assim se adaptarem a variações nas especificações das experiências mesmo que estas sejam ligeiras. Isto torna-se ainda mais notório quando se trata de injecção física de faltas. Estes são portanto condicionalismos que devem ser devidamente ponderados no projecto de injectores, nomeadamente, quando se pretende que a estrutura base possa também ela ser usada para implementar o módulo de injecção - injector (Fig. 4.1). Desta forma, no seu projecto, devem ser tomadas opções quanto ao tipo de funcionalidades suportadas.

De forma a minorar possíveis limitações ou mesmo restrições à utilização do hardware base em funções de injecção, procurou-se aumentar a flexibilidade da sua configuração especificamente para este tipo de aplicação. Este aumento de flexibilidade é suportado num aumento da oferta de soluções para aplicações de injecção. Assim, foi seguida uma abordagem que contempla três opções que endereçam diferentes locais de injecção numa perspectiva de injecção física de faltas:

- Utilização da estrutura base, como unidade central do injector, sendo a parte responsável pela inserção das faltas, implementada num módulo especializado externo que opera de acordo com a unidade base, através de comando enviados pela interface de comunicações genéricas. Assim foi concebido um módulo especializado para efectuar injecção de faltas ao nível do barramento de comunicações.
- Prover na configuração base, hardware capaz de injectar faltas em módulos que implementem a função objecto de avaliação. A

injecção de faltas é efectuada no interior da unidade de processamento do módulo seleccionado, alterando o conteúdo dos registos, da memória e de outras interfaces endereçadas pelo processador.

A técnica utilizada para efectuar este tipo de injecção recorre à instrumentação do circuito que está implementada na unidade de processamento do módulo base (*Scan Chain Fault Injection*). Os recursos adicionais que permitem esta técnica ser uma opção, consistem na utilização de um do controlador IEEE1149.1 SCANSTA101da National [National05]. Este opera como mestre nas comunicações com TAP (*Test Access Port*) do circuito objecto de avaliação, enviando comandos com as faltas para o espaço de endereçamento referido.

- Implementar no próprio hardware funções que permitam injectar faltas no módulo que as comporta. Estando estes módulos integrados numa estrutura cujo objectivo primeiro é a avaliação do comportamento de comunicações através injecção, este hardware foi projectado para emular falhas nos transceiver's (Fig. 4.3). Assim, quando seleccionado o modo de injecção de faltas, e em resposta a comandos do exterior provenientes do injector, é possível provocar faltas nos circuitos de transmissão e/ou de recepção como as que resultam de falhas do transceiver.

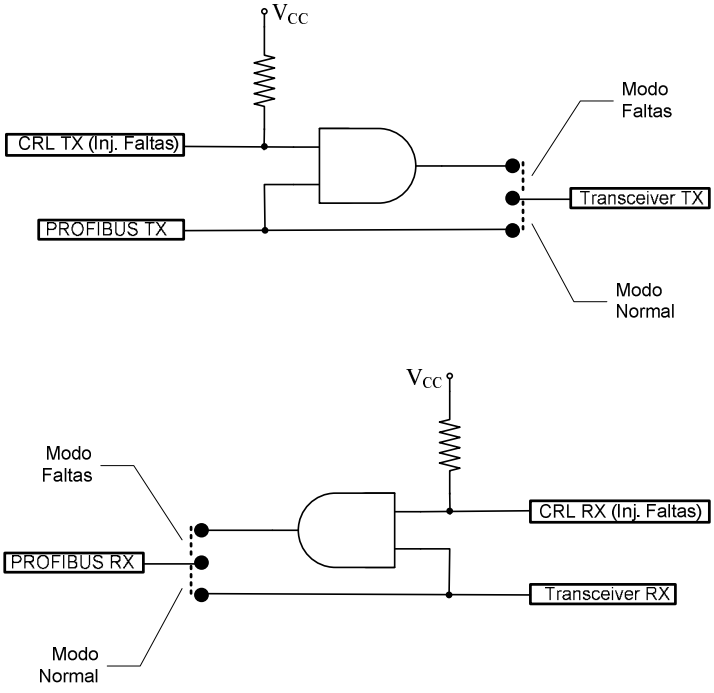

Figura 4.3 - Circuito para emulação de falha de *transceiver*.

O desenvolvimento do ambiente de injecção de faltas envolveu um significativo esforço que foi repartido pela componente de hardware e de software dos módulos da infra-estrutura. Um dos componentes que desempenhou

um papel de relevo, assumindo-se como um dos suportes tecnológicos desta ferramenta, foi o microcontrolador DSTni-LX [DSTni-LX03].

4.2.5 DSTNI-LX

O DSTni-LX-002 é um microcontrolador de 16 bits tendo por base um processador 100% compatível com o 80186 da Intel, estendendo por conseguinte essa mesma compatibilidade ao conjunto de instruções da família 8086. Combina a capacidade do processador com uma considerável quantidade de memória e de hardware de comunicação no próprio circuito. Estas características tornam-no capaz de suportar os requisitos de muitas aplicações embebidas de comunicações, nomeadamente, aquelas que são suportadas por protocolos com ampla utilização em comunicações industriais. Para suportar estes protocolos o DSTni-Lx incluiu na sua arquitectura as seguintes interfaces de comunicações (Fig. 4.4):

- **Portas série assíncronas de elevado débito**: capazes de suportar protocolos que exigem elevadas taxas de transmissão;
- **Controlador *ethernet***: capaz de suportar os protocolos IEEE 802.3, e o ANSI88023, que constituem a norma das redes *ethernet*;
- **Interfaces CAN 2.0B**: capazes de suportar taxas de transmissão até 1 Mbit/s;
- **Controlador para PROFIBUS-DP**: com capacidade para suportar configurações mestre ou escravo, deste tipo de rede.

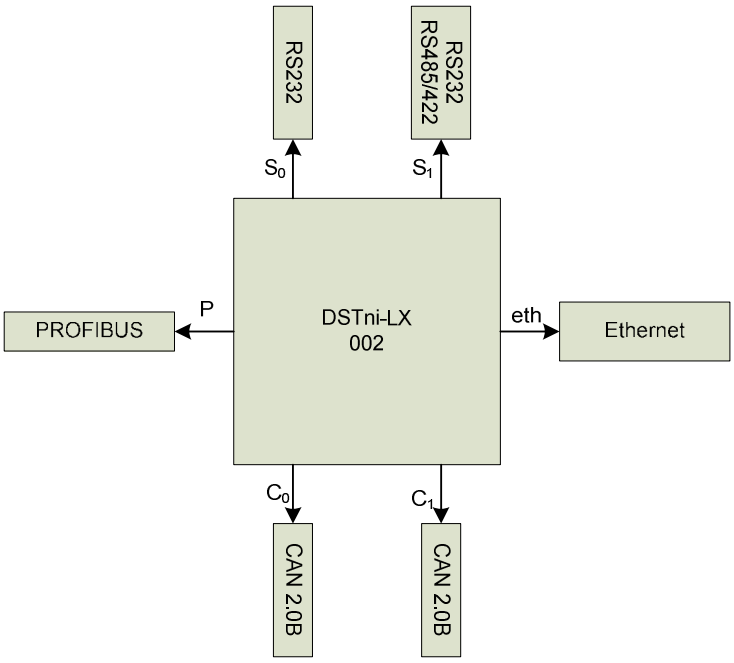

Figura 4.4 - Microcontrolador com suporte de comunicações multi-protocolo.

Enquadrado no ambiente de injecção de faltas proposto, a utilização deste microcontrolador capitaliza um conjunto de vantagens:

- A integração de um alargado número de recursos num único circuito limita a dimensão do hardware nomeadamente ao nível da complexidade das soluções desenvolvidas. Neste sentido, a utilização deste microcontrolador permite apresentar uma configuração compacta para a base de hardware, como ilustra a figura da sua implementação (Fig.4.5);
- O DSTni-LX é um dos microcontroladores reconhecido e indicado pela organização PROFIBUS para o suporte das suas aplicações. Este reconhecimento não pode ser dissociado do facto do microcontrolador integrar o ASPC2 (*Advanced Siemens PROFIBUS Controller 2*) como controlador do PROFIBUS-DP. A sua utilização confere igualmente a vantagem da utilização de um dos ASIC's mais representativo e amplamente difundido em aplicações industriais do PROFIBUS-DP [Siemens05].

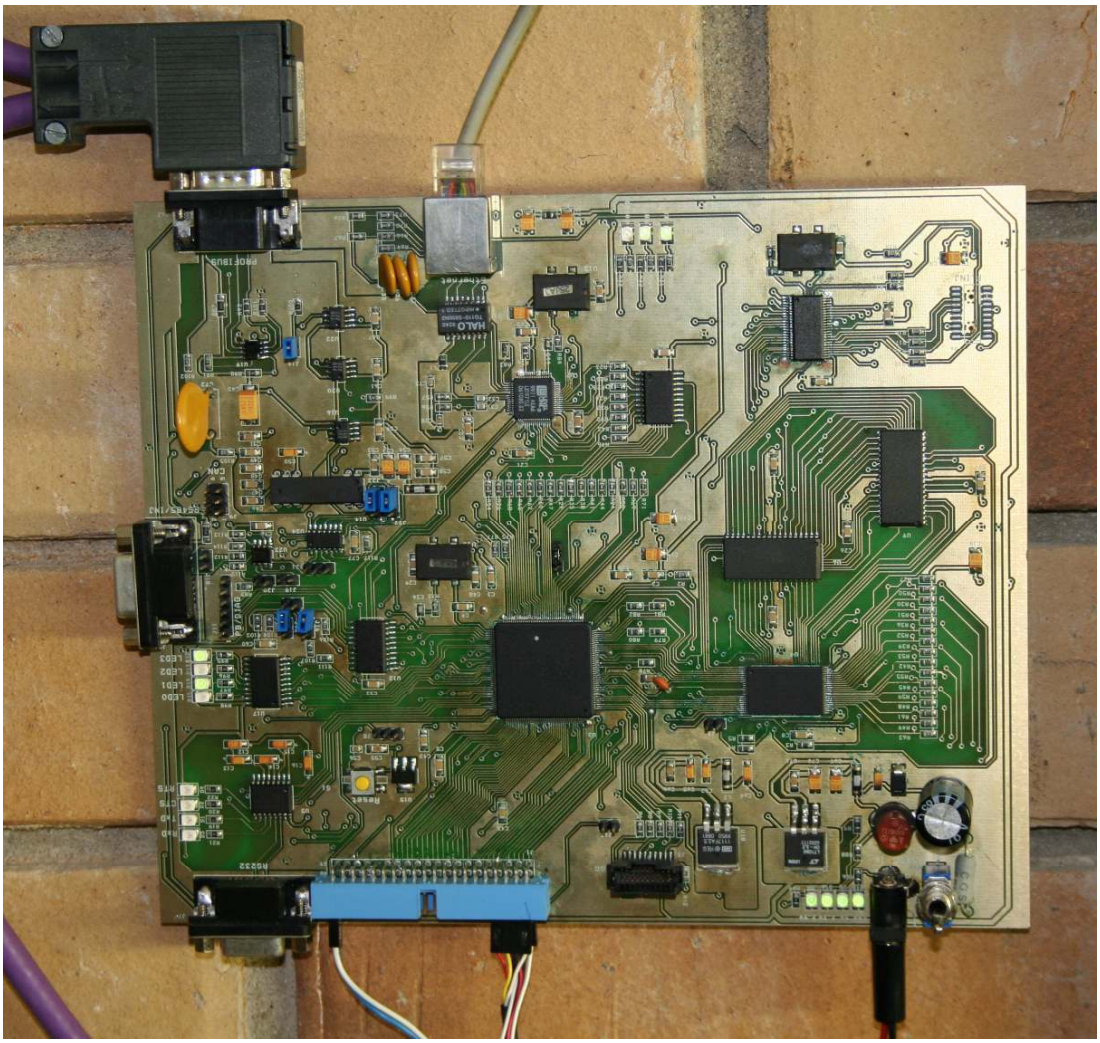

Figura 4.5 - Estrutura base de hardware.

4.2.6 ASPC2

A Siemens tem um historial considerável na oferta de soluções para aplicações em redes PROFIBUS-DP. Uma componente importante desse historial está associada ao desenvolvimento de circuitos de aplicação específica que suportam a implementação dos protocolos de comunicação. Nestas soluções inclui-se um largo espectro de controladores concebidos para satisfazer requisitos impostos por problemas de diferentes escalas.

Neste contexto, existem controladores que são capazes de satisfazer a totalidade dos requisitos da aplicação sem a necessidade de utilização de microcontroladores. Estão englobadas neste grupo, aplicações que fazem uso de estações passivas de baixa complexidade (Fig.4.6). Para aplicações de maior complexidade, existem opções em que os controladores assumem parte significativa das tarefas das comunicações libertando o processador para o desempenho de outras tarefas do sistema.

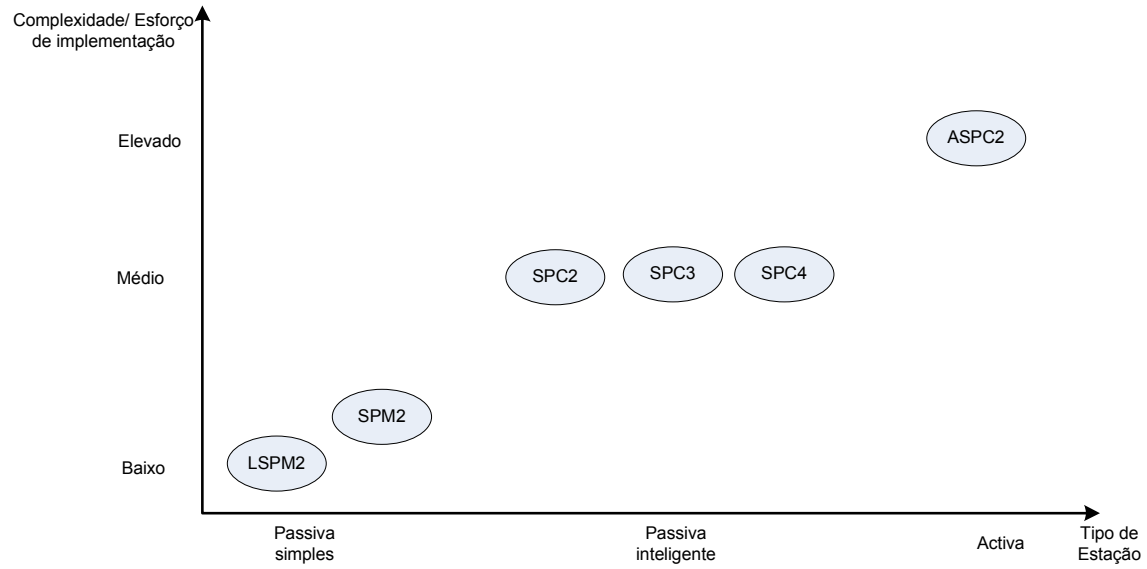

Figura 4.6 - Gama de ASIC's Siemens e sua aplicação.

O ASPC2 representa o nível mais elevado de desenvolvimento desta gama de ASIC's, implementando a camada física e de ligação de dados de acordo com a norma IEC 61158. Actualmente, o seu desenvolvimento atingiu a revisão E, que permite assegurar as funções fornecidas pela extensão DP-V2 do protocolo, para além da DP-V0 e DP-V1, já asseguradas em revisões anteriores. O elevado desempenho evidenciado pelo ASPC2 torna-o especialmente indicado e amplamente utilizado para o suporte das comunicações de PLC's, de controladores de servomecanismos, e também no controlo e monitorização de processos.

O DSTni-LX implementa a última revisão deste ASIC fornecendo assim, uma base de trabalho sólida e validada, que permite o acesso aos parâmetros de configuração e outra informação relevante de grande utilidade para a análise da

operação do protocolo. Esta informação, pode ser obtida na camada de ligação de dados e níveis superiores, permitindo obter uma imagem detalhada dos vários estados de operação.

4.3 Infra-Estrutura de Comunicações

A infra-estrutura de comunicações constitui a plataforma de ensaio onde são realizadas as experiências conducentes à avaliação do comportamento da rede. Experiências essas que fazem parte de dois grupos de ensaio:

- Avaliação da estabilidade do anel lógico;
- Análise da resposta de tempo-real.

No primeiro tipo de experiência intervêm unicamente estações activas. Já no segundo grupo participam todos elementos da rede, ou seja, estações activas e passivas. Torna-se então necessário disponibilizar estações adaptadas às especificidades da operação que vão desempenhar.

4.3.1 Estações Passivas

Nas redes PROFIBUS-DP o mecanismo de troca de informação privilegia a utilização de tramas de acção (*action frames*), num processo de comunicação mestre escravo. Enquadrado neste tipo de funcionamento, os serviços de comunicação são sempre desencadeados e concluídos na estação activa, consequentemente nas experiências de injecção de faltas, o registo dos eventos, que formam a base de trabalho do processo de avaliação da rede, será centralizado nessas estações.

Desta forma as estações passivas não requerem nenhuma configuração específica, podendo a sua função na infra-estrutura de comunicações ser desempenhada por um qualquer equipamento que se encontre disponível comercialmente. Não obstante isto representar um factor de simplificação do esforço de implementação da infra-estrutura, por uma questão de racionalização de recursos a maior parte deste tipo de estação foi implementada com base na estrutura base de hardware. Neste contexto foi desenvolvida uma *stack* de comunicações que implementa este tipo de estação de acordo com a norma da rede [EN96b]. O desenvolvimento consistiu na implementação das máquinas de estados correspondentes à camada de utilizador (Fig. 4.7) sobre os serviços da FDL disponibilizados pelo ASPC2.

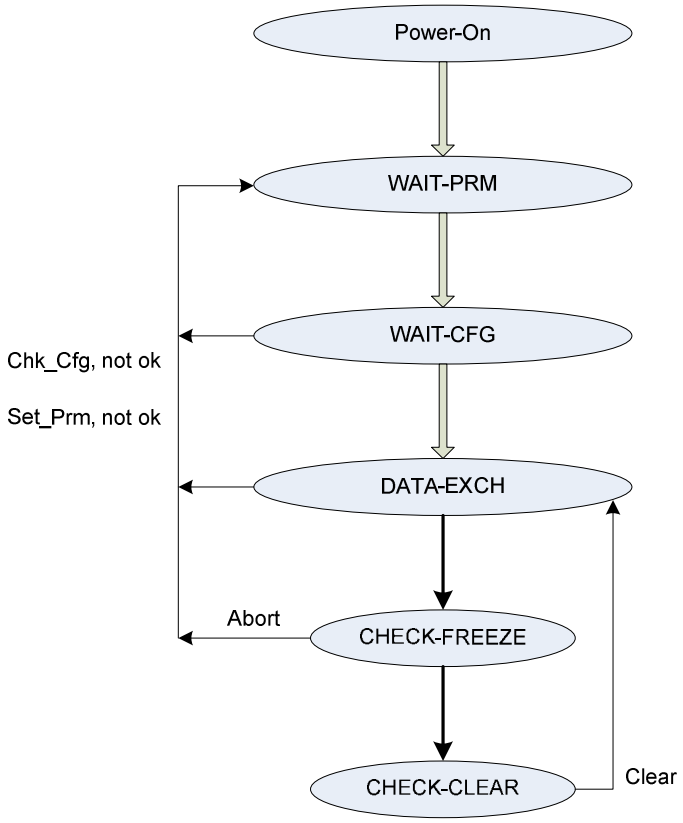

Figura 4.7 - Máquina de estados da camada de utilizador de uma estação passiva.

O necessário processo de validação do nó desenvolvido foi efectuado com recurso a testes intensivos, sendo a sua operação comparada para as mesmas condições com o escravo EM277 da Siemens. Destes testes, obteve-se uma validação conferida por um comportamento das estações semelhante, quer ao nível da operação, quer ao nível do desempenho.

4.3.2 Estações Activas

O desenvolvimento de estações activas PROFIBUS-DP reveste-se de uma considerável complexidade. Do ponto de vista da implementação, parte dessa complexidade encontra-se concentrada na camada 3 do protocolo e deriva do facto de parte substancial da estrutura do PROFIBUS-DP se encontrar concentrada na *DDLM* (*Direct Data Link Mapper*) e sobretudo na interface com o utilizador (*User Interface*) (Fig 4.8).

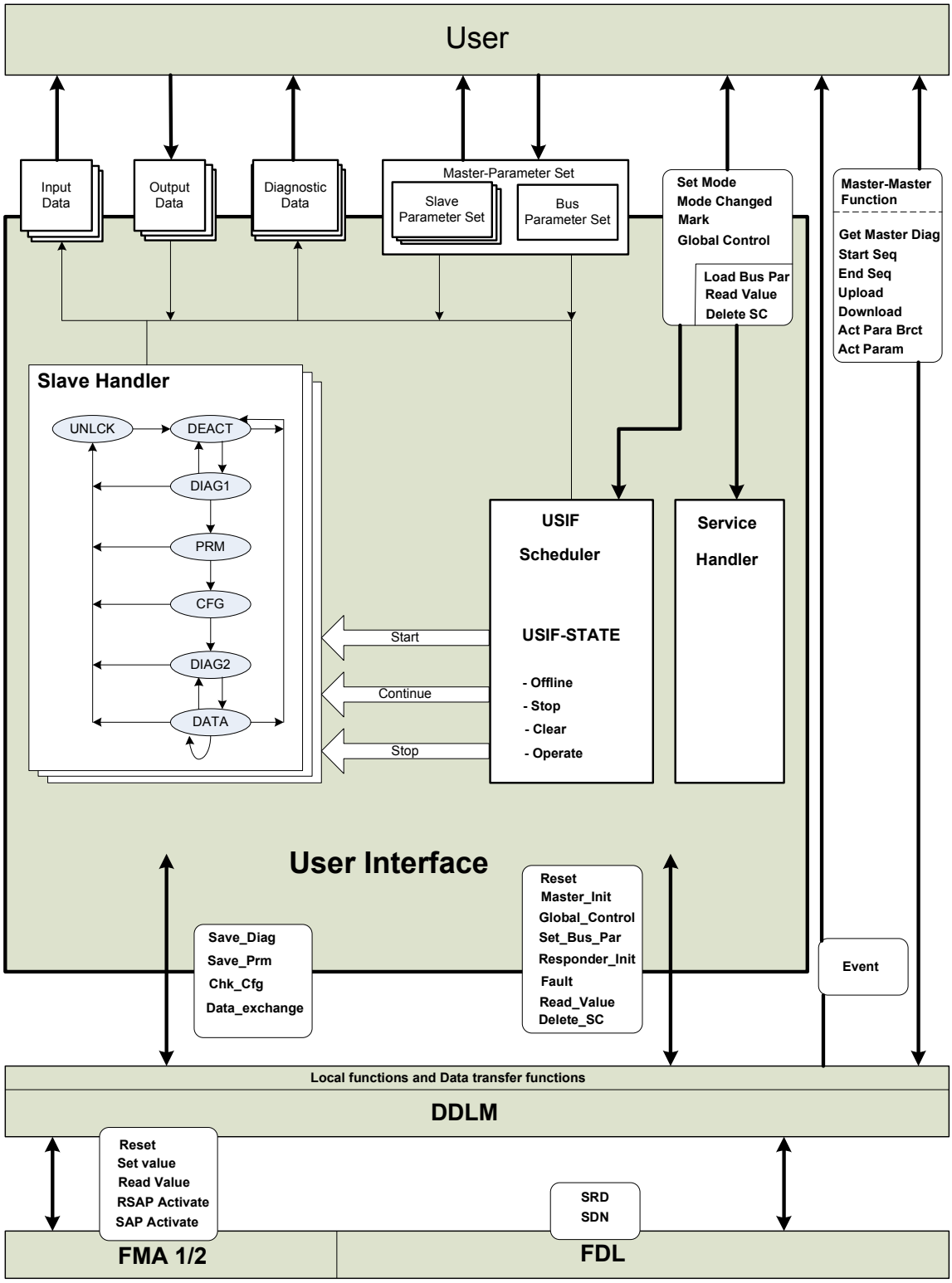

Figura 4.8 - Estrutura de um mestre classe 1.

O DDLM estabelece a ligação entre a Interface com o utilizador e a FDL. A sua função consiste em efectuar um mapeamento dos pedidos com origem no utilizador em serviços da FDL.

A interface com o utilizador (*User Interface*) constitui o núcleo do protocolo, uma vez que nela se encontram as funções que regulam as operações de comunicação. Essas operações são efectuadas em três blocos funcionais:

- ***Slave handler***: que efectua a gestão das comunicações com cada uma das estações passivas que estão na sua esfera de controlo;
- ***Scheduler***: que comanda a execução das *slave hadndlers* de cada estação passiva;
- ***Service handler***: que efectua o controlo de funções realizada na própria estação, nomeadamente de funções de configuração.

O bloco utilizador (*User*) representa a aplicação que faz uso do sistema de comunicação. Tipicamente é constituída por equipamentos de controlo que trocam informação com o subsistema de comunicações numa base temporal cíclica usando para tal uma área de memória partilhada.

4.3.2.1 Implementação

A configuração utilizada na implementação das estações activas da infra-estrutura de comunicações envolveu o desenvolvimento de software da camada 3 do protocolo. Este opera directamente sobre a camada 2, que é suportada pelo hardware do ASPC2. Esta fronteira entre a implementação de hardware e a de software é ela própria uma interface natural, onde se pode analisar a operação da estação activa.

Do ponto de vista da análise das comunicações, é nesta interface que entram em fila de espera para transmissão as mensagens que provêm do utilizador, e onde são disponibilizados pela FDL os resultados desses serviços de comunicação. A realização das medidas para a avaliação do comportamento das comunicações, obtidas a este nível, engloba todas as contribuições de factores e comportamentos da rede que são de difícil modelação. A extrapolação dos resultados para níveis superior do protocolo é um problema de muito menor dimensão. Da mesma forma a este nível é mas fácil correlacionar o comportamento da rede com eventos observados quer ao nível do barramento, quer devido a estados de operação assinalados pela FDL (ASPC2).

Neste contexto a estação foi instrumentada a este nível de forma a poder adquirir todo um conjunto de informação associada:

- A erros detectados pela FDL;
- A estados de operação da FDL;
- A actividade relacionada com os serviços de comunicação, ou seja: envio, recepção e retransmissões de tramas do protocolo.

A implementação da camada 3 comportou algumas alterações na estrutura de módulos da interface com o utilizador (*User Interface*). Estas modificações visaram adequar o seu funcionamento ao ambiente de injecção de faltas. A sua introdução foi condicionada no pressuposto da sua inclusão não induzir nenhum

factor desviante à norma PROBIBUS, nomeadamente ao nível do comportamento da rede de comunicações ou na operação de outros participantes no processo de comunicação.

Neste contexto, a estrutura principal do protocolo foi mantida nomeadamente ao nível de um módulo estruturante como é o *slave handler*. Assim, foi implementado integralmente na sua estrutura, esta máquina de estados, estando a sua implementação de acordo com a norma da rede para comunicações mestre classe 1, ou seja comunicações entre estações activas e passivas (Fig. 4.9).

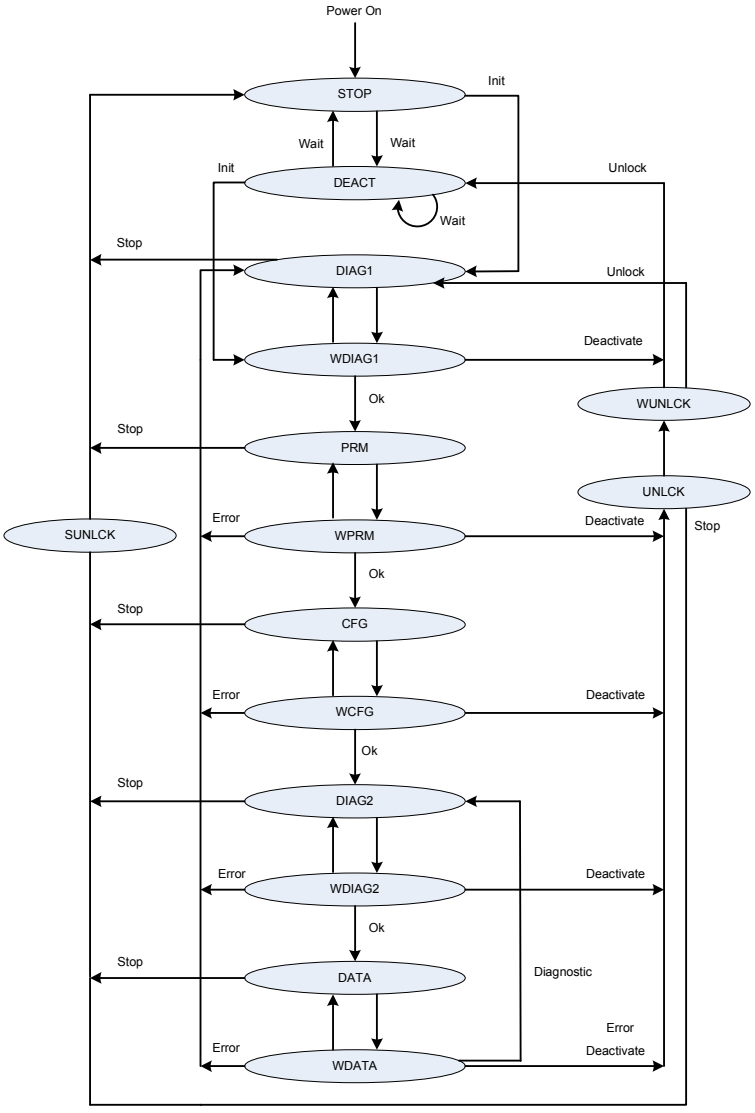

Figura 4.9 - Máquina de estados do módulo *slave handler*.

As alterações incidiram essencialmente ao nível do módulo *scheduler*, que foi substituído por outro que acumula também as funções de *service handler*. Este novo módulo é responsável pelo arranque e paragem das *slave handler* e de estabelecer a ligação das estações activas com a restante infra-estrutura de injecção.

A figura 4.10 representa o diagrama de estado com as tarefas mais relevantes desempenhadas por este módulo.

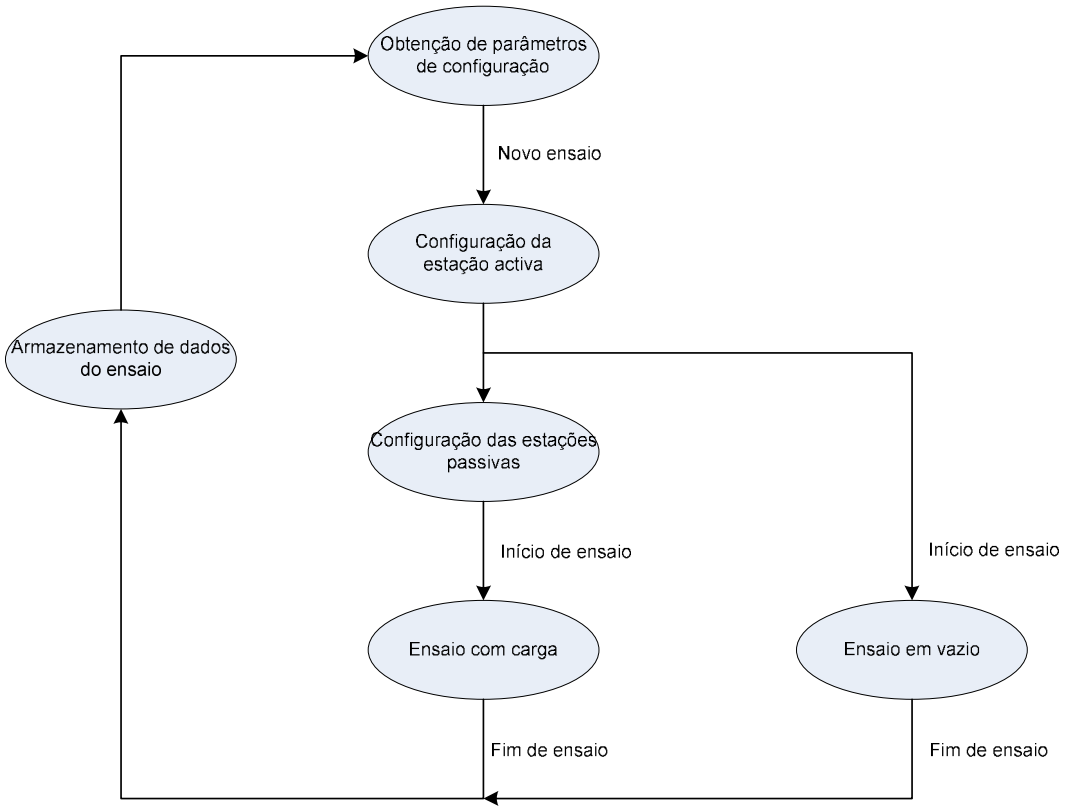

Figura 4.10 - Diagrama de estados das tarefas de coordenação da operação das estações activas no ambiente de injecção de faltas.

Este módulo implementa duas variantes que permitem adaptar a estação quer para ensaios em vazio, quer para ensaios com carga. Estes ensaios são realizados respectivamente como objectivo de avaliar a estabilidade do anel lógico, e avaliar a resposta de tempo-real da rede em cenários de faltas.

O bloco opera em ciclo, estabelecendo um padrão de funcionamento que se adequa à realização de ensaios de pequena duração, como aqueles que são executados nas experiências de injecção. Desta forma, no início de cada ciclo, é estabelecida uma ligação com a unidade de gestão com o intuito de obter os parâmetros que permitam recrear as condições suportam os cenários formulados para os ensaios. Para a infra-estrutura de comunicações os parâmetros de configuração consistem tipicamente em:

- Parâmetros de configuração da FDL;
- Parâmetros de configuração das estações passivas;
- Parâmetros associados às mensagens e seu escalonamento.

Após a recepção dos parâmetros de configuração, e em conformidade com estes, a operação da estação deriva para um modos: ensaio com carga ou ensaio em vazio.

No caso de ensaios com carga a máquina de estados evolui para a parameterização das estações passivas, fazendo evoluir as suas máquinas de estado até ao estado de troca de dados – *Data* (Fig 4.9). A recepção de um sinal

de início de ensaio desencadeia um processo de acerto do relógio de tempo-real, e dá início à execução das comunicações e recolha de informação do comportamento do nó durante esse período.

Expirado o tempo estipulado para o ensaio, os dados recolhidos são enviados para armazenamento na unidade de gestão, dando-se início a um novo ciclo de ensaios.

Da mesma forma que na estação passiva, a operação destas estações foi igualmente objecto de validação. Esse processo consistiu na verificação da operação em rede com outros equipamentos, e a sua interoperabilidade, com estações passivas cujos fabricantes possuem equipamento certificado. Assim, foi verificada a sua correcta operação com estações da Siemens (EM 277) e da Omron PTR1-COM.

4.4 Injector

A injecção de faltas é uma técnica pela qual a operação de um dado sistema pode ser avaliada. Consiste na introdução de faltas de forma controlada durante a sua operação. Este processo permite acelerar a taxa de incidência de eventos anormais comparativamente com o que ocorre durante a operação normal do sistema, possibilitando assim a sua avaliação para um espectro alargado de cenários, que podem ocorrer ao longo do seu ciclo de vida.

De forma a poder obter resultados que sejam relevantes do ponto de vista da análise do funcionamento do sistema objecto de avaliação, o processo de injecção deve garantir o cumprimento de certos requisitos. De entre eles estão aqueles que são impostos pela fundamentação matemática requerida para que os seus resultados tenham representatividade, face à componente dos fenómenos que são estudados. Tipicamente este é um processo estocástico que inerentemente requer um tratamento adequado de acordo com a teoria das probabilidades.

Enquadrando-se com estes requisitos, a operação do injector de faltas está essencialmente orientada para a introdução de erros de natureza aleatória de acordo com uma distribuição de probabilidade que melhor descreva a ocorrência dos eventos que afectam o sistema.

Este processo de injecção tem funções em tudo semelhantes às utilizadas na análise de sistemas discretos, e que são também amplamente aplicadas em técnicas de análise por simulação [Tyszer99] [Banks96]. Um elemento determinante para a operação deste tipo de técnicas prende-se com a geração dos números aleatórios, que suportam as variáveis pelas quais os eventos que ocorrem no sistema são representados.

Estes números devem respeitar importantes propriedades estatísticas, nomeadamente devem possuir distribuição uniforme e ser estatisticamente independentes.

A geração destes números em computador não garante que eles sejam estritamente aleatórios. Contudo, é possível garantir uma boa qualidade em termos das propriedades estatísticas referidas. Em virtude deste facto, estes números são também designados por pseudo aleatórios. Não obstante estes números derivarem de um número inicial designado de semente, que possui um padrão de repetição determinístico, o que contraria as propriedades dos números estritamente aleatórios, esta é uma característica, que em muitas aplicações de simulação não pode ser considerada uma desvantagem. Apresenta antes a vantagem de permitir a reprodução de experiências (repetibilidade) ou a possibilidade de estes serem facilmente portados para avaliação de outros sistemas.

A maior parte das linguagens de programação disponibilizam suporte para a geração de números aleatórios. Para que as propriedades estatísticas referidas se verifiquem é necessário aplicar um conjunto de algoritmos e testes que garantam a qualidade dos números gerados. Este é um tema amplamente debatido na área da simulação da operação de sistemas discretos, e para o qual são propostos vários métodos [Tyser99] [Banks96].

Assim, é possível mesmo em compiladores de 32 bit como os que suportam as linguagens de programação C e C++ gerar números aleatórios de muito boa qualidade.

O suporte à injecção de faltas através da aplicação de técnicas para a geração de números aleatórios no injector apresenta algumas implicações com impacto, designadamente ao nível da sua operação e da sua estrutura:

- A geração em tempo-real destes números é incompatível com o decorrer do processo de injecção de faltas. Desta forma, a sua implementação no injector só pode ser enquadrada num cenário em que sua geração decorre antes de ser iniciar a experiência de injecção.
- A sua inclusão aumenta consideravelmente a complexidade do injector e pressiona significativamente o processo de validação da sua operação.

A implementação de técnicas de injecção de faltas em sistemas reais (físicos) é extremamente exigente ao nível da resposta temporal, estando esta directamente relacionada com as especificidades apresentadas pelas suas funções. Tipicamente a operação do injector está subordinada à realização de todas as tarefa que integram o processo de injecção de uma falta, numa fracção do tempo dispendido na realização da operação do componente do sistema que se pretende afectar. A duração de uma operação de escrita ou de leitura por parte de um microprocessador, ou a duração de um *bit* que circula num barramento de um

sistema de comunicações, e que se pretende alterar, são exemplos que dão uma indicação da escala temporal em que um injector tem de operar.

Este cenário revela quão severos podem ser os requisitos de tempo-real que envolvem estas tarefas. Associados a este cenário, surgem muitas das vezes problemas de controlabilidade do processo de injecção. De uma forma geral, estes ocorrem pelo facto de, não se poder congelar ou alterar a dinâmica do sistema durante a injecção de faltas.

Considerando todos os condicionalismos referidos, foi utilizado no injector uma abordagem, que privilegia a operação deste na componente da coordenação e execução das tarefas de injecção. Todo o processo de criação dos cenários de injecção, incluindo a geração de números aleatórios, é efectuado externamente, através do recurso a software comercial, para a qual não é necessário particular cuidado com a validação.

Inserido nesta estratégia, as tarefas que fazem parte do processo de injecção de faltas, são decomposta em três partes principais:

- **Geração de cenários de injecção**: esta tarefa consiste na geração de números aleatórios de acordo com a distribuição de probabilidade pretendida para o cenário em análise. Na configuração base da injecção de faltas, a cada número aleatório está associado à modificação do valor lógico do estado do barramento, durante um período de tempo correspondente à duração de um *bit* (T_{bit}).

  Contudo para taxas de transmissão mais elevadas, o tempo que o sistema de injecção dispõe para proceder à modificação do nível lógico pretendido é consideravelmente reduzido. Neste cenário o sistema de injecção terá dificuldade em apresentar uma resposta temporal capaz de executar todas as tarefas subjacentes à modificação desse *bit* de forma controlada.

  Assim, a informação resultante da geração dos números aleatórios é posteriormente processada de forma a compatibilizar as grandezas físicas envolvidas com a capacidade de resposta temporal do sistema que as vai executar.

  O pós-processamento efectua uma divisão temporal da experiência em *slots* (unidades de tempo) cujo valor corresponde a $16T_{bit}$. Com base nesta escala temporal é efectuado um mapeamento dos números aleatórios, inserindo-os posteriormente em vectores de 16 posições cuja posição se encontra sincronizada relativamente à dimensão temporal da experiência de injecção de faltas.

- **Despacho**: esta tarefa implementa uma função típica de despacho de um escalonamento. A mudança de escala introduzida na função anterior facilita as operações a este nível. Assim, é efectuado o

despacho dos vectores que foram escalonados em *slots* temporais, para o nível de execução das faltas.

- **Execução**: tarefa que decorre sincronizada com o sistema na escala temporal mais baixa T_{bit}, de forma a tornar efectiva a falta.

As duas últimas tarefas em que foi decomposto o processo de injecção, são implementadas no injector, cuja estrutura apresenta uma arquitectura hierárquica baseada em dois níveis [Carvalho05a]:

- Unidade de controlo;
- Ponta de injecção.

4.4.1 Unidade de Controlo

A unidade de controlo representa o nível superior da arquitectura do injector e desempenha um papel central na operação da infra-estrutura de injecção de faltas. Integrada nesta estrutura assume funções de sincronização e participa no processo de injecção, transmitindo comandos de injecção através do despacho de vectores de faltas. Opera de forma cíclica decorrendo a sua operação de acordo com o diagrama de estados da figura 4.11, no qual estão representadas as suas principais tarefas.

Esta componente do injector é implementada na configuração base de hardware e durante a sua operação estabelece três ligações com as demais estruturas.

- Uma com o nível superior do ambiente de injecção de faltas, do qual obtêm:
  - **Parâmetros de configuração**: que estão relacionados com a parametrização da interface PROFIBUS-DP, assim como da configuração do relógio que comanda a injecção de faltas.
  - **Dados de injecção**: que são constituídos por vectores de faltas e respectivos tempos em que devem ser aplicados;
  - **Informação complementar de gestão**: que permite a coordenação da infra-estrutura de injecção (Nós de comunicação, Injector, Monitor). No processo é verificado se os intervenientes estão prontos para a realização das experiências ou, se as concluíram com sucesso e se os respectivos dados foram armazenados.
- Uma ao nível horizontal que permite a sincronização de tempo-real da estrutura. Consiste no envio de uma mensagem *broadcast* pela rede PROFIBUS-DP, com intuito de proceder ao acerto do relógio do sistema que se encontra implementado em cada um dos

módulos intervenientes, e simultaneamente ordenar o início da experiência de injecção de faltas.

- Uma última estabelecida com a ponta de injecção para o envio de comandos de injecção.

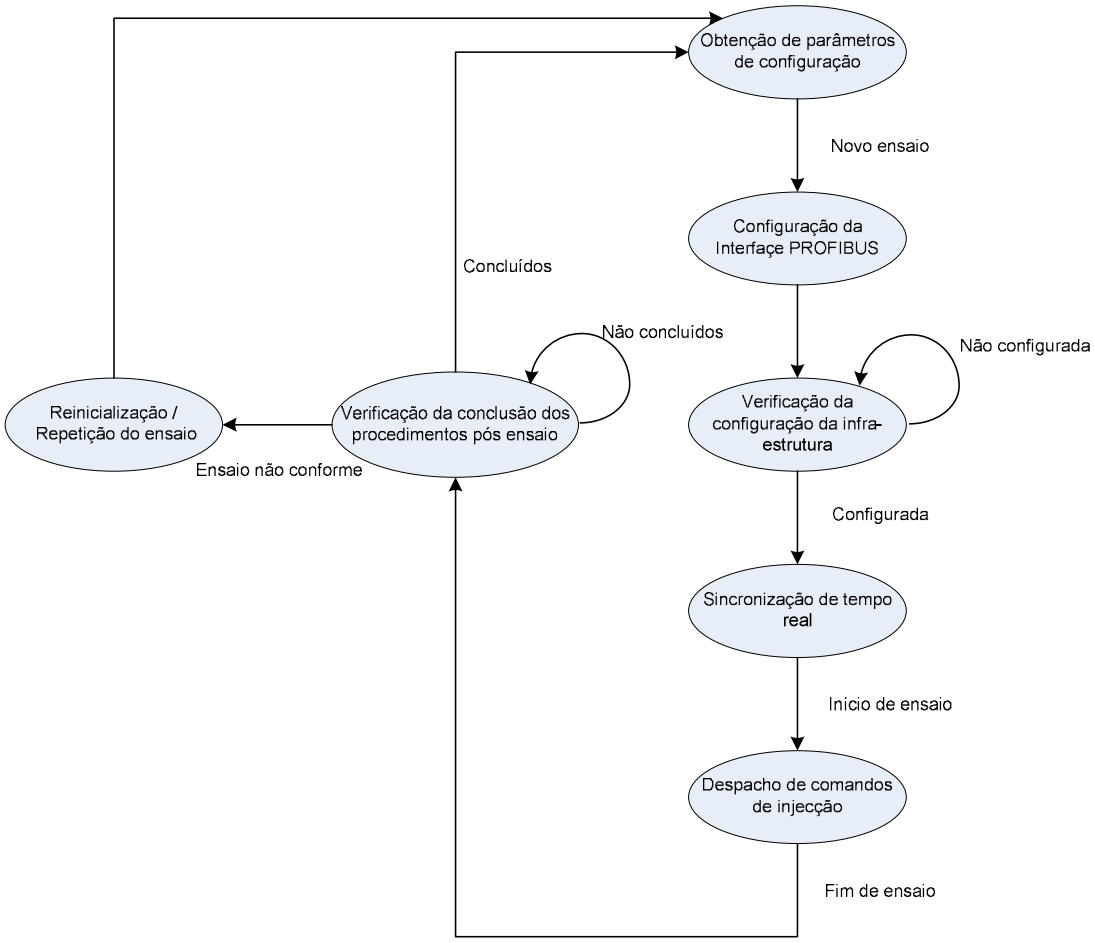

Figura 4.11 - Tarefas executadas pela unidade de controlo do injector.

4.4.2 Ponta de Injecção

A ponta de injecção assegura o nível inferior da arquitectura do injector. Foi especificamente projectada para proceder às etapas finais do processo de injecção, ou seja traduzir os comandos de faltas que são enviados pela unidade de controlo, em erros no sistema de comunicações.

A estrutura da ponta de injecção, assim como, de uma forma geral a dos componentes que se encontram no final da cadeia do processo de injecção, estão fortemente dependentes das características do objecto de estudo e dos atributos que se considerem relevantes para as experiências. Sejam eles relativos ao local de injecção, tipo de faltas, resolução e dinâmica requerida para o processo.

As condições consideradas na avaliação do comportamento do PROFIBUS-DP estão centradas na análise da sua operação, quando esta é afectada por faltas

que resultam de interferências electromagnéticas. Tipicamente estas são de natureza transitória e traduzem-se em erros que incidem nos *transceivers* e de forma mais significativa nos sinais do barramento de comunicações.

Neste contexto a arquitectura da ponta de injecção deve estar adaptada para proceder à injecção de faltas nestes componentes do sistema de comunicações. Da mesma forma, deverão ser considerados quais o efeitos espectáveis das faltas no objecto de avaliação. Neste caso, as faltas que ocorrem nestes componentes têm resultados semelhantes quando analisados na perspectiva da operação protocolo de comunicações. Assim, o processo natural de injecção passa pela introdução de erros nos sinais do barramento de comunicações.

Numa rede PROFIBUS-DP a informação transferida no barramento está codificada em sinais eléctricos com o formato de caracteres de UART. Tipicamente os sinais no barramento incluem (Fig 4.12):

- Pausas na comunicação (*Idle Time*) com duração variável. Estas são utilizadas pela FDL no estabelecimento de estados de operação;
- Sinais de sincronismo para correcta operação da UART (*Start e Stop bits*);
- Dados, mais sinal de protecção da integridade dos dados, encontrando-se estes inseridos entre os sinais de sincronismo da UART.

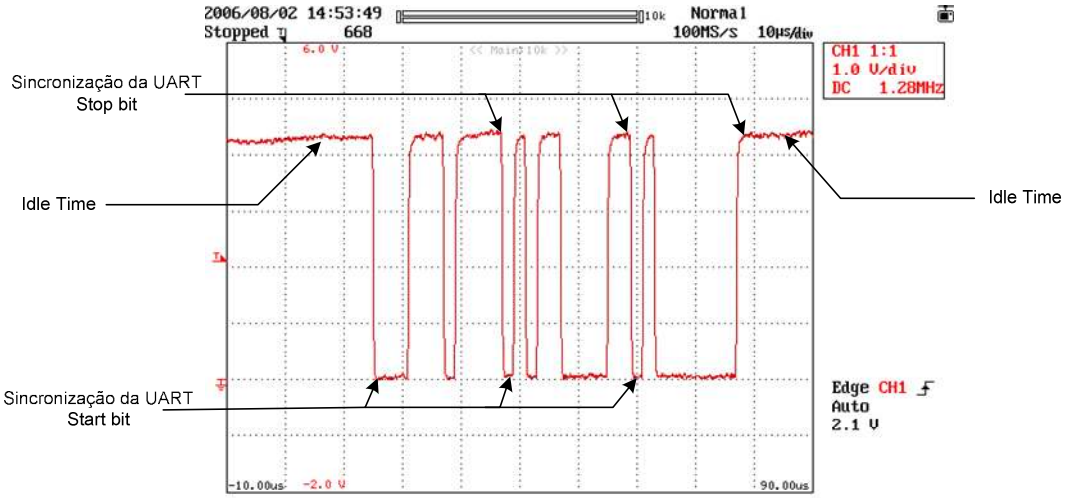

Figura 4.12 - Oscilograma representativo de um telegrama do PROFIBUS-DP constituído por 3 caracteres UART.

A injecção de faltas deve então decorrer de forma que estas sejam detectadas pela UART das estações que integram a rede, e que o resultado da mesma se traduza num erro. Este erro deve respeitar o modelo de inversão de bit (*bit flip*), e pode levar à alteração de qualquer uma das componentes dos sinais representados no oscilograma da figura 4.12.

A UART é um dispositivo cuja função está orientada para assegurar comunicações série de cariz assíncrono. Esta característica das comunicações dificulta o processo de injecção nomeadamente no que refere à sincronização com os sinais do barramento. Uma incorrecta sincronização pode ter como consequência a não injecção da falta, ou que esta seja injectada numa posição diferente da pretendida.

Não obstante as comunicações terem carácter assíncrono, os dados contidos nos caracteres de UART são amostrados de forma síncrona. Nesta operação são usados os bits de sincronismo nomeadamente o bit de arranque (*start bit*), para proceder à sincronização do relógio interno com o sinal do barramento de comunicações. Assim, o valor de cada *bit* é amostrado no centro do sinal que o representa, ou seja após ter decorrido 50% do tempo de persistência do sinal no barramento. Tempo este que varia de acordo com a taxa de transmissão (Fig. 4.13).

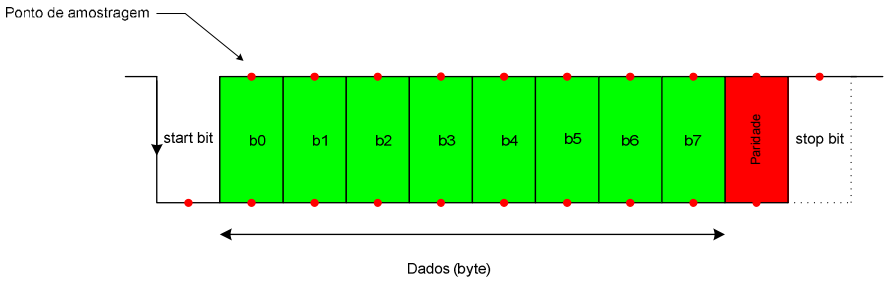

Figura 4.13 - Amostragem de sinais numa UART.

A ponta de injecção deve então integrar no processo de injecção mecanismos que garantam a controlabilidade do processo e que o seu factor de cobertura seja tendencialmente de 100%. Neste contexto deverá ser implementado um sistema de sincronização semelhante ao usado pelas UART's, e obedecer ao seguinte conjunto de procedimentos (Fig4.14):

- Obter o valor do *bit* na primeira fase da sua manifestação no barramento;
- Proceder à injecção forçando o sinal (*forcing*) antes de decorrer o processo de amostragem efectuado pela UART;
- Libertar o barramento na fase final da persistência do *bit*, de forma a garantir que um processo semelhante possa ser efectuado no *bit* seguinte sem que o valor a injectar seja falseado.

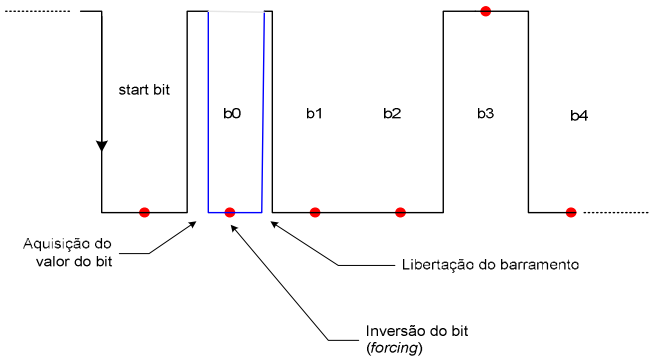

Figura 4.14 - Processo de injecção.

4.4.2.1 Arquitectura

O processo de injecção descrito envolve funções que são críticas na perspectiva da resposta temporal. Assegurar estas funções com base em soluções de software é uma opção de difícil implementação, ou mesmo inviável quando no processo estão envolvidas comunicações com taxas de transmissão elevadas. As especificidades impostas pelos requisitos destas funções tornam mais viável o recurso a técnicas baseadas na aplicação de hardware dedicado.

A solução utilizada na ponta de injecção foi desenvolvida em hardware recorrendo a tecnologia de elevado desempenho, na qual o hardware que implementa determinadas funções críticas apresenta tempos de atraso da ordem de 1.5 a 3ns. Uma descrição funcional da arquitectura encontra-se representada na Figura 4.15.

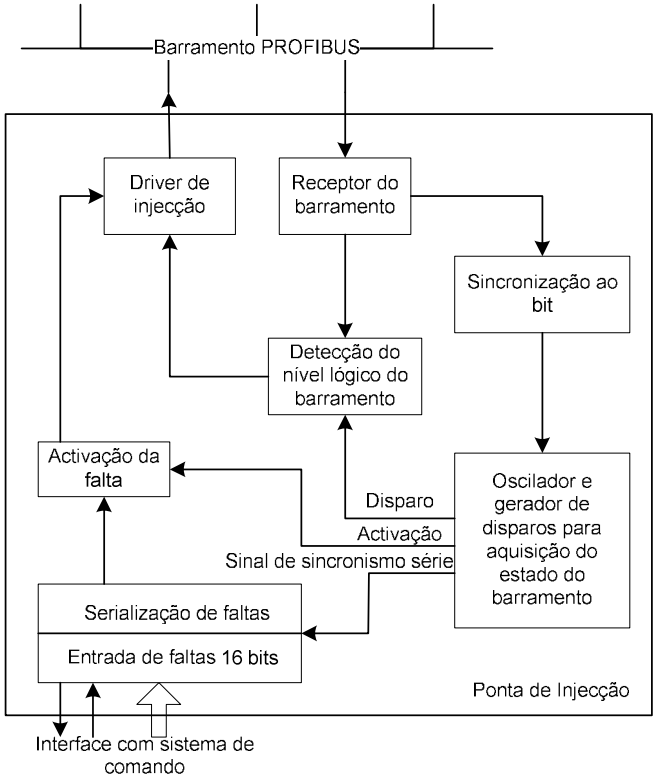

Figura 4.15 - Diagrama funcional da ponta de injecção de faltas.

A ponta de injecção lê permanentemente o barramento de forma a obter informação dos sinais que o percorrem. Esta informação é processada pelo bloco **Sincronização ao bit** que detecta todas as transições descendentes do sinal 1→0, excluindo aquelas que resultam do processo de injecção.

O bloco **Oscilador e gerador de disparos para aquisição do estado do barramento**, gera um sinal de sincronismo com período igual à duração de um *bit* do caracter de UART. Este sinal está na base de toda a coordenação das tarefas que constituem a operação da ponta de injecção. A sua consistência com o estado do sinal do barramento é assegurada através de constante monitorização dos eventos gerados no bloco *Sincronização ao bit*, e em conformidade com estes procede a ajustes que corrigem desvios.

Os oscilogramas das figura 4.16 e 4.17 foram obtidos a partir da ponta de injecção, e representam respectivamente na parte superior o sinal do barramento e na parte inferior o sinal de sincronismo gerado a partir dele.

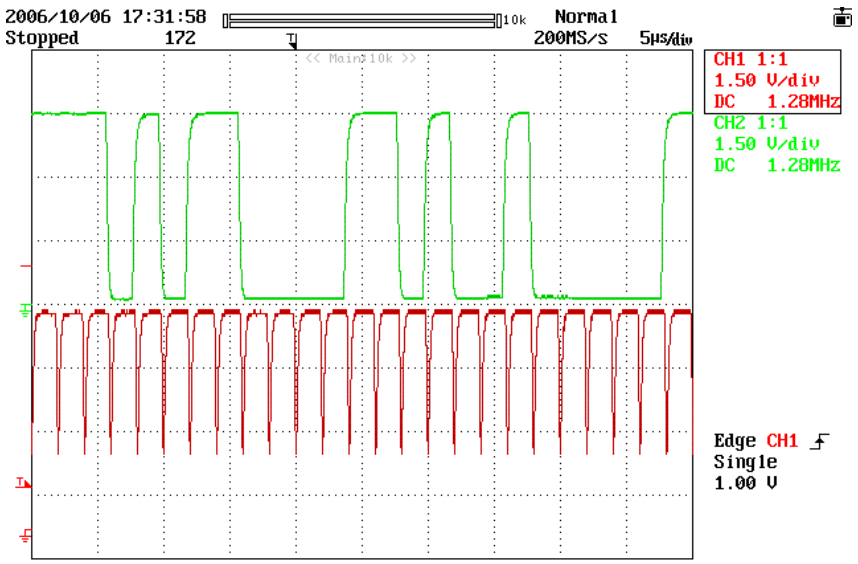

Figura 4.16 - Sinal de sincronismo gerado a partir do sinal do barramento de comunicações.

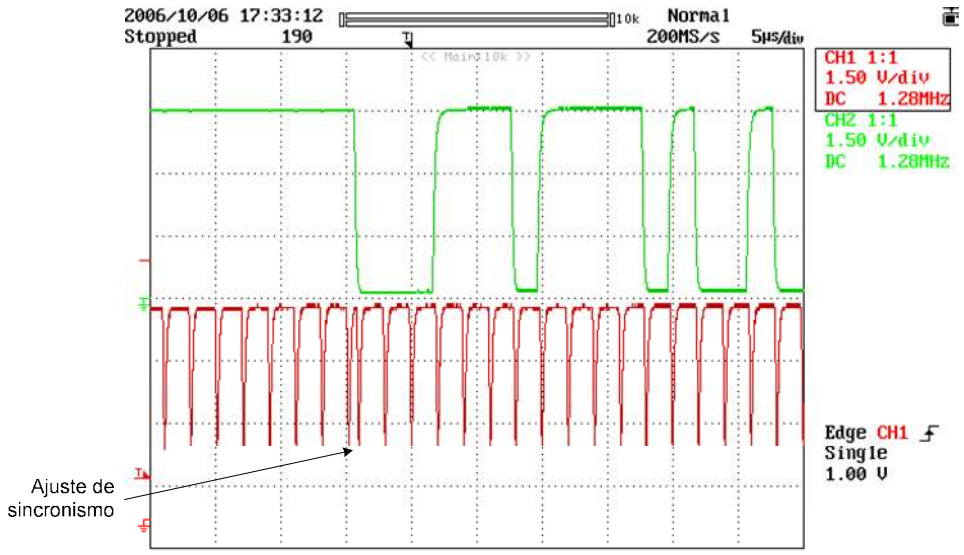

Figura 4.17 - Ajuste do sinal de sincronismo.

Na figura 4.17 está representado o ajuste do sinal de sincronismo após um período de inactividade no barramento. Nestas condições, como pode ser constatado na figura, é espectável a ocorrência de maiores desvios em consequência da ausência de eventos de correcção do sincronismo.

A forma de onda do sinal de sincronismo apresenta uma configuração que reflecte as necessidades de controlo da operação da ponta de injecção e vai de encontro a requisitos específicos de comando das tarefas:

- **Transformação paralelo série dos comando de faltas enviados pela unidade de controlo**. Esta operação é efectuada através da serialização em sincronismo com o **sinal de sincronismo série** (*serial clock*);
- **Controlo do tempo que a ponta de injecção está activa no barramento**. Esta função é comandada pelo sinal **activação** (*enable*) que restringe o período de actividade da ponta de injecção à estritamente necessária à injecção de faltas, inibindo a sua operação nomeadamente no período de aquisição dos sinais do barramento;
- **Controlo da aquisição do nível lógico do sinal do barramento**. Esta tarefa é comandada pelo sinal **disparo** (*trigger*) que dispara o processo que conduz à aquisição do valor lógico correspondente ao *bit*. Este valor será posteriormente invertido no processo de injecção.

O processo é concluído pelo bloco ***Driver* de injecção** que fisicamente impõe um potencial eléctrico que altera o sinal do barramento.

O desempenho da ponta de injecção pode ser observado nos oscilogramas das figuras, 4.18 a 4.20, que representam três momentos de injecção de faltas com resultados distintos. As figuras reportam os instantes que decorrem após a ordem

de injecção representada no sinal da parte inferior do oscilograma e que resulta da serialização de um vector de faltas. Os efeitos das faltas no sinal do barramento são representados na parte superior desse mesmo oscilograma.

As duas primeiras figuras mostram a inversão de *bits* que fazem parte da informação que circula no barramento de comunicação. A última refere-se à injecção de uma falta durante o período de inactividade do barramento. Esta última tem como consequência geração de um sinal que é interpretado como um *start bit* para as UART's.

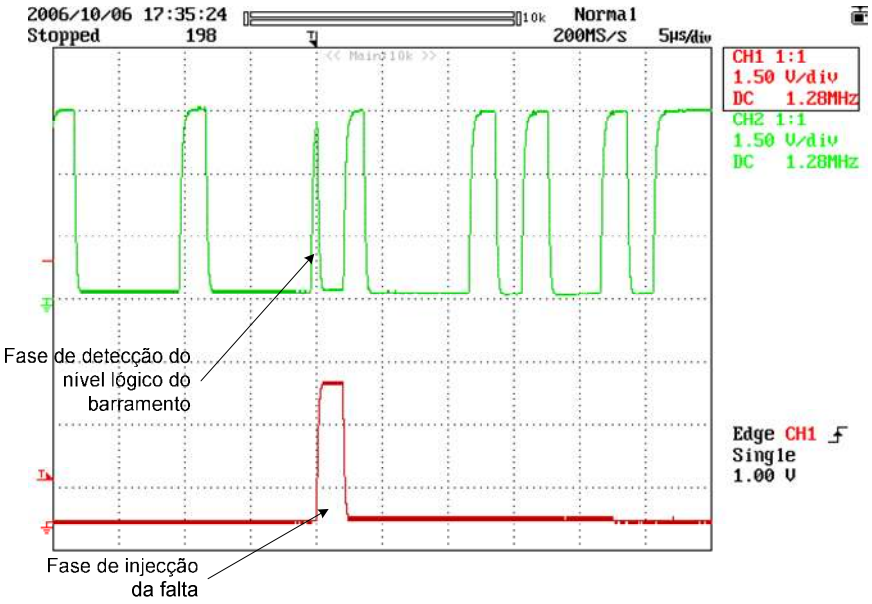

Figura 4.18 - Inversão de um sinal do nível lógico 1 para nível lógico 0.

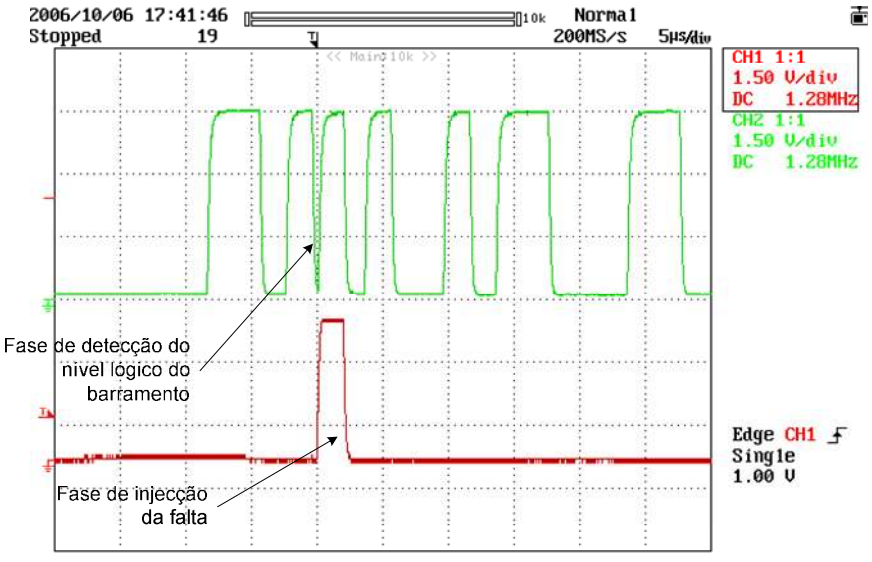

Figura 4.19 - Inversão de um sinal do nível lógico 0 para nível lógico 1.

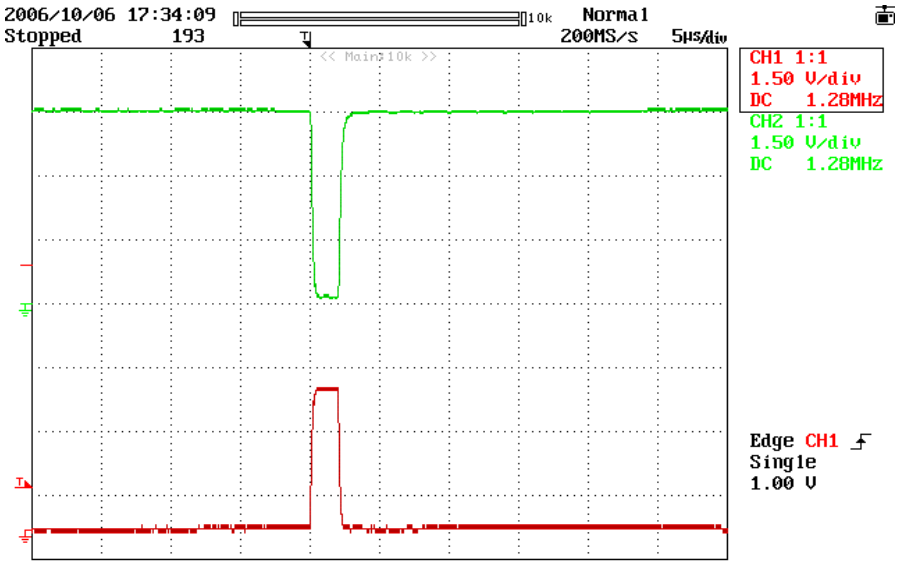

Figura 4.20 - Injecção de falta durante o período de inactividade do barramento

4.5 Monitor

O barramento de comunicações é o local em que incide a maior parte das faltas durante a normal operação do sistema de comunicações. É igualmente o local onde é possível obter uma visão global da manifestação dessas faltas, quer seja ela na forma com são alterados os sinais das comunicações, quer nas alterações ao padrão de comportamento do tráfego consequência dos erros gerados. Por conseguinte este é também um local de eleição para a obtenção de informação relevante para a avaliação do comportamento do sistema de comunicações.

Esta característica torna importante a existência de um observador do estado que registe a actividade do barramento. O monitor é módulo da infra-estrutura de injecção de faltas que assume esta função.

As funções de monitorização da actividade no barramento processam-se através do suporte a comunicações série, disponibilizado pela porta série de alto débito que se encontra incluída na configuração base de hardware. Esta porta é configurada com os mesmos parâmetros da porta série que equipa as estações PROFIBUS-DP. É conectada ao barramento sem contudo efectuar nenhuma outra operação que não seja a recepção (leitura) de todo e qualquer evento que ocorra independentemente de ser assinalado algum erro na sua recepção (ex. erros de paridade, ou ausência de *stop bit* - *framing error*).

Durante as experiências de injecção toda a actividade do barramento é registada. Tipicamente esta actividade é constituída por tramas e por caracteres

gerados pelo processo de injecção. A cada um destes eventos é associado pelo monitor o registo temporal da sua ocorrência.

A actividade de monitorização está integrada nas experiências de injecção de faltas, decorrendo de forma coordenada com a operação dos restantes módulos, num processo que leva ao armazenamento dos dados para posterior análise (Fig 4.21).

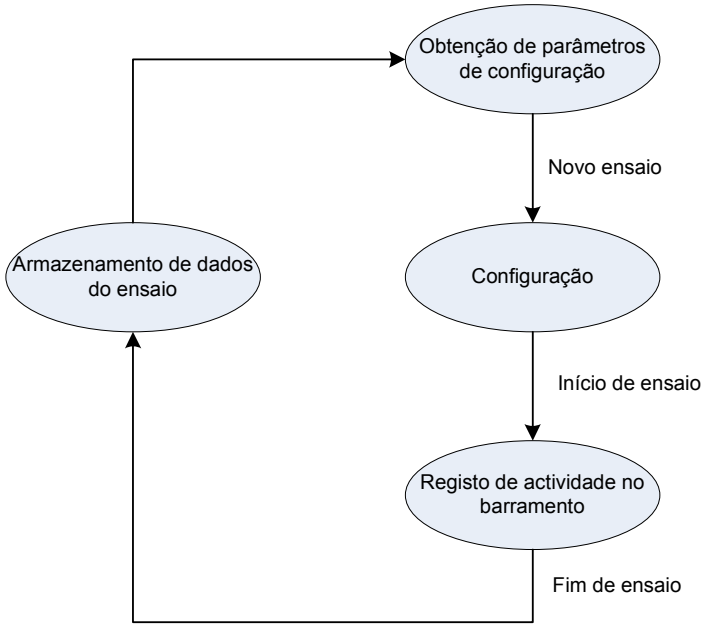

Figura 4.21 - Tarefas do monitor.

A análise dos resultados das experiências enquadrada pelas regras do protocolo, e pela cronologia dos eventos, permite identificar importantes detalhes da operação da rede, nomeadamente: a inserção e remoção de estações do anel lógico, perdas de *token*, ou mesmo identificar causas associadas à remoção ou a perdas de *token*, e de outros estados de operação da FDL.

Numa perspectiva global, esta informação pode ser cruzada com os estados da FDL obtidos em cada estação, e assim obter conhecimento relevante da dinâmica da operação do protocolo, nomeadamente identificar os mecanismos que fazem disparar as várias excepções ao normal funcionamento do protocolo.

A rede PROFIBUS-DP assenta muito da sua organização na troca de tramas de gestão entre estações. A inserção e remoção de estações do anel, assim como o comportamento da FDL decorrem em função do conteúdo dos *token* que circulam na rede. Baseado neste modo de operação é possível desenvolver uma aplicação que efectua a emulação da operação da FDL de cada estação e assim obter uma visão integrada do estado das estações da rede.

A utilização deste emulador permite que o monitor seja igualmente usado fora do contexto da avaliação por injecção de faltas para que foi desenvolvido. Incorporando algumas alterações sobretudo ao nível do software, é possível estender a sua aplicação à avaliação do comportamento de redes PROFIBUS-DP constituídas por nós comerciais (ex. PLC) em ambiente industrial. Neste caso, a

sua utilização é efectuada em redes nas quais existe um historial de anomalias na operação. A sua aplicação permite identificar as causas e avaliar quais os eventos que estão na origem da má operação, ou da diminuição da disponibilidades dos equipamentos, consequência de falhas do sistema de comunicação.

4.6 Gestão do Ambiente de Injecção

O ambiente de injecção de faltas integra uma grande diversidade de entidades que são produtoras e consumidoras de quantidades consideráveis de informação. Fisicamente estas entidades encontram-se distribuídas, possuem dinâmicas temporais muito diversas, e por vezes não existe comunicação directa entre produtor e consumidor.

Apresenta igualmente uma estrutura heterogénea na qual, coexistem ferramentas genéricas disponíveis comercialmente, com outras desenvolvidas para preencher funções específicas no processo de avaliação. As quais por sua vez estão implementadas sobre diferentes plataformas, nomeadamente sobre sistemas operativos Windows, Linux, ou mesmo nos sistemas embebidos que constituem os módulos da infra-estrutura de injecção. A coordenação destas entidades e o estabelecimento de canais de comunicação entre aplicações requer a existência de uma entidade que garanta estes serviços.

A unidade de gestão representa a solução implementada para cumprir estas tarefas. Consiste numa aplicação que corre num computador pessoal, desenvolvida sobre o sistema operativo Linux. Estabelece as comunicações e redirecciona os fluxos de informação existentes no ambiente de injecção, quer seja através do acesso à informação pré-existente em sistemas de ficheiros, quer através de comunicações entre processos suportadas em *sockets*.

A unidade de gestão assume assim um papel central de coordenação de toda a operação do ambiente de injecção, e da qual dependem directa ou indirectamente todas as actividades que culminam nas experiências de injecção de faltas, bem como no processo de análise de resultados.

Na Figura 4.22 é apresenta graficamente uma panorâmica das actividades que decorrem no ambiente de injecção, onde está salientada a função integradora desempenhada pela unidade de gestão. As actividades são apresentadas de acordo com uma estrutura cronológica que representa duas importantes fases de operação:

- Fase de Configuração;
- Fase de Análise.

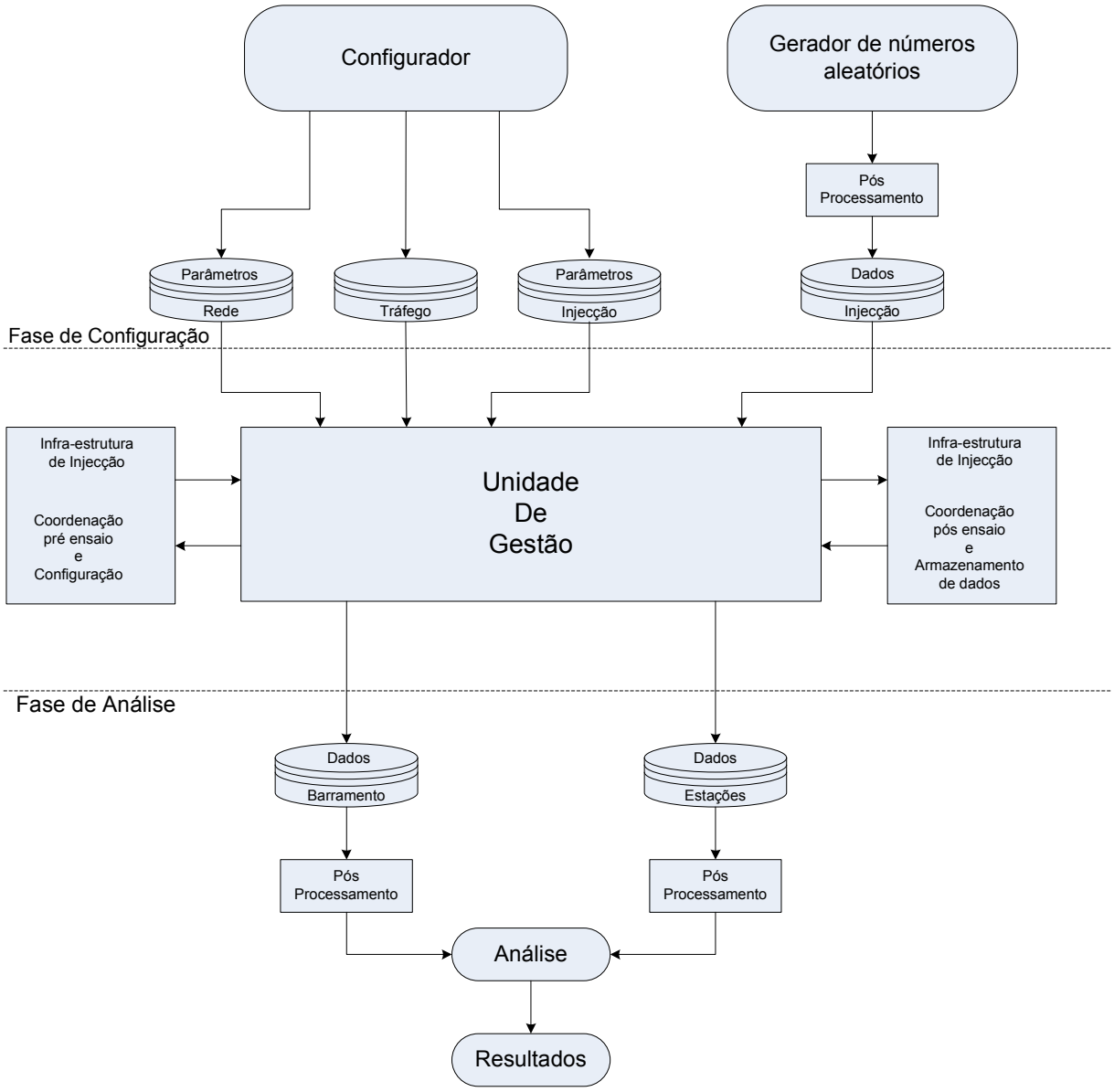

Figura 4.22 - Actividade do gestor do ambiente de injecção de faltas.

4.6.1 Fase de Configuração

As experiências de injecção de faltas são precedidas por um conjunto de operações efectuadas com o objectivo de reunir as condições operacionais fundamentais à sua execução. Nestas operações estão englobadas tarefas relacionadas com o planeamento das experiências e a coordenação da infra-estrutura de injecção.

O planeamento decorre num horizonte temporal muito distinto daquele em que se realizam as experiências de injecção. Por conseguinte não requer a operação conjunta das ferramentas que estão envolvidas no planeamento, com a estrutura operacional, ou seja este é um processo que decorre em *offline*.

Pelo contrário, a coordenação da infra-estrutura de injecção envolve tarefas que estão directamente relacionadas com a configuração e coordenação da estrutura operacional. Este processo antecede o início das experiências de injecção de faltas e portanto requer a interacção da unidade de gestão com a infra-estrutura de injecção, num processo que decorre em modo *online*.

Neste contexto, a fase de configuração pode ser apresentada com base na descrição das operações que se encontram englobadas nos dois blocos seguintes:

- Planeamento operacional;
- Coordenação da infra-estrutura de injecção.

4.6.1.1 Planeamento Operacional

A operação de infra-estrutura de injecção rege-se por um conjunto de informação que a configura definindo um perfil de operação adequado aos cenários considerados relevantes para efeitos da avaliação. Assim, toda a infra-estrutura tem de ser parametrizada.

Esta operação é efectuada com recurso ao **configurador**, uma ferramenta desenvolvida para assistir a configuração da infra-estrutura. Esta ferramenta disponibiliza num ambiente gráfico o acesso a um conjunto significativo dos parâmetros de configuração da infra-estrutura. Nestes incluem-se:

- **Configuração das estações da rede**: nomeadamente ao nível dos endereços taxas de transmissão e dos demais parâmetros da FDL;
- **Tráfego na rede**: através da definição das mensagens de cada estações, serviços de comunicação que as suporta, estações destinatárias, bem como possibilidade de efectuar escalonamento de tráfego síncrono e assíncrono;
- **Parâmetros de injecção**: nos quais estão englobados: a base de tempo do relógio de injecção; o modelo de faltas; o número de experiências, e os respectivos apontadores para os ficheiros de faltas;
- **Parâmetros de monitorização**: relativos à escala temporal usada no processo de registo de eventos do sistema, e apontadores para os locais de armazenamento da informação recolhida.

Esta informação é guardada num ficheiro que funciona como repositório de dados o qual é acedido pela unidade de gestão para proceder à configuração da infra-estrutura de injecção.

A informação que serve de base aos cenários de faltas é produzida no **gerador de números aleatórios**. Esta função é assegurada pelo *Statgraphics*. Trata-se de uma ferramenta de software comercial vocacionada para análise estatística [Statgraphics], que foi integrada no ambiente de injecção para gerar números aleatórios de acordo com a distribuição e a probabilidade pretendida.

De forma a garantir experiências estatisticamente independentes e igualmente com uma dimensão que seja compatível com os recursos do hardware base, nomeadamente ao nível da capacidade de memória, foi desenvolvida uma aplicação que faz uma operação de pós processamento dos números aleatórios gerados. Esta operação consiste no mapeamento dos números aleatórios em vectores, numa escala temporal de acordo com a taxa de transmissão da rede, seguida de escalonamento. Em simultâneo é efectuada uma operação de segmentação que divide esta informação em ficheiros correspondentes a experiências de 3,5s de duração.

4.6.1.2 Coordenação da Infra-Estrutura de Injecção

A coordenação da infra-estrutura de injecção é efectuada pela unidade de gestão, que fica activa nos momentos que antecedem a realização das experiências e num período imediatamente após a sua execução.

A unidade de gestão consiste numa aplicação com a qual os módulos da infra-estrutura de injecção estabelecem comunicação de acordo com o modelo cliente servidor. Durante esta transacção a unidade de gestão procede ao conjunto de operações que dão seguimento aos planos operacionais desenvolvidos durante a fase *offline* do processo de configuração/planeamento. Nessas operações inclui-se:

- **Reconhecimento do hardware**. A unidade de gestão identifica o hardware e as funções que este vai desempenhar. Para manter a consistência da operação da infra-estrutura faz o seguimento destes módulos, mesmo que os parâmetros que os identifiquem se alterem durante horizonte temporal da avaliação;
- **Configuração**. Invoca os ficheiros que resultaram da fase de planeamento das operações, e procede à configuração de cada um dos módulos com a informação correspondente ao ensaio que vai ser efectuado;
- **Suporte à sincronização da infra-estrutura**. Auxilia o injector no processo que visa assegurar que toda a infra-estrutura está configurada. Estão incluídas nestas configurações não só aquelas que são comandadas pela unidade de gestão, mas também as que são do domínio dos módulos, nomeadamente no estabelecimento de comunicação com as estações passivas e a sua configuração;
- **Gestão de informação**: Assegura a gestão do armazenamento da informação, num processo que é desencadeado através do pedido de vários módulos da infra-estrutura;
- **Validação do ensaio**: Verifica se todos os intervenientes no ensaio assinalam a conclusão do ensaio, sem a existência de eventos que levem à sua anulação.

4.6.2 Fase de Análise

De forma semelhante ao que ocorre na fase de configuração, a fase de análise decorre numa etapa distinta da realização dos ensaios (fase pós-ensaio), não necessitando portanto de existir uma ligação entre a estrutura operacional e a de análise de resultados.

O objecto de trabalho utilizado nesta fase é constituído pelos resultados que foram produzidos nos ensaios, e que se traduzem essencialmente em dois tipos de informação:

- Uma relativa aos eventos que ocorrem no barramento, que é obtida pelo monitor;
- Uma outra relativa aos eventos que são assinalados ao nível da FDL de cada estação.

Em ambos os casos a informação reporta todos os eventos que correram no período do ensaio, e por conseguinte contém muita informação que não é relevante do ponto de vista da análise a que se destina. Assim, antes de ser utilizada para o processo de análise é submetida a um pré-processamento que filtra os eventos não relevantes.

Esta tarefa é efectuada através de uma aplicação que foi desenvolvida para este efeito. No caso dos dados do barramento, a aplicação segue o conceito apresentado em §4.5, implementando uma máquina de estados global na qual é reflectido o estado de cada estação no anel lógico. Identifica entre outros eventos: perdas de *token*; saídas e entradas de estações no anel. A sua operação está baseada na emulação da máquina de estados da FDL das estações e fazendo evoluir o estado global do sistema com base na observação dos *tokens* na rede, e na implementação de uma lista de estações activas (LAS).

A informação adquirida nas estações é processada pela mesma aplicação, que efectua uma identificação, e descodificação das excepções e estados da FDL relevantes para a análise.

O resultado do pré-processamento é guardado em ficheiros, onde é incluída igualmente a persistência destes eventos. A execução do estágio final da fase da análise é efectuada com recurso a ferramentas que se encontram já disponíveis. Esta é uma opção que foi assumida no projecto do ambiente de injecção, pelo que a informação do bloco de pré-processamento está formatada para ser facilmente importada por ferramentas especializadas em tratamento estatístico de dados, ou por ferramentas de uso mais, genérico, como é o caso da folha de cálculo Excel.

Nestas ferramentas os dados são tratados e utilizados nos cálculos que permitem obter medidas de avaliação do desempenho na presença de faltas, ou mesmo de aspectos da confiança no funcionamento da rede. Tipicamente as medidas consistem na obtenção da probabilidade de ocorrência dos eventos, valores esperados para as várias componentes temporais associadas aos eventos, e dos parâmetros que descrevem o comportamento de tempo-real da rede, para o cenário de faltas e de cargas considerados nas experiências.

As medidas são obtidas a partir da aplicação de métodos estatísticos que requerem o cumprimento de critérios de representatividade. Desta forma no processo de cálculo são efectuados testes à representatividade dos resultados. Esta depende em grande medida do número de ensaios, podendo por conseguinte haver necessidade de ser formulado um novo planeamento, baseado numa projecção de um número de ensaios suplementares e assim realimentar o ciclo de experiências no ambiente de injecção, num processo inerentemente iterativo.

4.7 Estimação e Qualidade dos Estimadores para Injecção de Faltas

A técnica de injecção de faltas é aplicada fundamentalmente em dois contextos distintos: supressão, e previsão de faltas. Em ambos os casos a aplicação da técnica tem como objectivo obter indicadores relacionados com a forma como o sistema, ou partes que o constituem desempenham a sua função.

Na supressão de faltas os indicadores são essencialmente de natureza qualitativa, as faltas injectadas no sistema têm carácter determinístico e as experiências de injecção decorrem em sincronismo com a sua operação. Esta forma de aplicação da técnica de injecção serve para validar as soluções testadas, sendo igualmente utilizado para remover faltas introduzidas na fase de projecto ou durante a sua implementação.

Pelo contrário, na previsão de faltas os indicadores são essencialmente de cariz quantitativo, e o processo de injecção é inerentemente estocástico.

Uma aplicação típica da técnica de injecção neste contexto está associada à avaliação da confiança no funcionamento de sistemas, nos quais é depositado elevados índices de confiabilidade. Os indicadores associados a essa avaliação estão essencialmente centrados na determinação:

- Do factor de cobertura dos mecanismos de tolerância a faltas implementados pelo sistema;
- Das latências apresentadas na detecção das faltas.

O factor de cobertura representa o grau de eficiência com que os mecanismos de tolerância a faltas:

- Efectuam a detecção de erros;
- Identificam e isolam os componentes afectados;
- Efectuam a reconfiguração do sistema quando necessário.

Em suma a eficiência de cada um dos atributos associados à operação dos mecanismos de tolerância a faltas pode ser quantificada através do factor de cobertura, que representa a probabilidade desse mecanismo efectuar

correctamente a sua operação [Powell95] [Cukier99]. O seu cálculo é efectuado com base na variável aleatória discreta *X* que caracteriza a operação do mecanismo de tolerância a faltas e que assume o valor 0 ou 1 conforme:

$$X = \begin{cases} 1, & \textit{Se a falta é tratada em conformidade} \\ 0, & \textit{Se o mecanismo de tolerância a faltas falha} \end{cases}$$

O valor da variável aleatória *X* reflecte a operação do mecanismo de tolerância a faltas inserido na sua envolvente. Tipicamente a envolvente de um sistema é caracterizada pelo conjunto de faltas que o afecta, e da carga que executa, função da actividade que se desenrola no mesmo.

As faltas *f* que são injectadas no sistema fazem parte de uma amostra da população *F*, que representa a totalidade das faltas que podem ocorrer durante o ciclo de vida do sistema. Portanto, *f* deve ser suficientemente representativo da população *F*, de forma que as observações que resultam da sua aplicação também o sejam.

A actividade que ocorre no sistema é representada pelo conjunto *A*, que deve ser igual à carga real do sistema, ou o mais próximo possível desse cenário de operação. Assim, o factor de cobertura *c* conferido pelos mecanismos de tolerância deve ser definido com base na completa caracterização do espaço de entrada do sistema, ou seja de acordo com o conjunto *G* que representa o produto cartesiano $G = f \times a$.

$$c = prob\{X = 1 | g \in G\} \qquad (4.1)$$

Considerando que, para cada elemento *g* do conjunto *G* é observado *x(g)* na saída o sistema, que representa o valor da variável *X* relativo a esse elemento, e em que *g* ocorre com probabilidade *p(g)*, a expressão do factor de cobertura pode ser rescrita.

$$c = \sum x(g)p(g) \qquad \forall g \in G \qquad (4.2)$$

A expressão assim definida para o cálculo do factor de cobertura apresenta contudo algumas limitações quanto à sua aplicação prática, pois é requerido o conhecimento da função de distribuição de probabilidade *p(g)*, sem a qual não é possível formular qualquer conclusão à cerca dos resultados.

Em termos práticos o factor de cobertura pode ser inferido através da média μ de *X*, desde que os eventos *p(g)* sejam equiprováveis ($p(g) = 1/|G|$).

$$c = \mu = \frac{1}{|G|}\sum x(g) \qquad \forall g \in G \qquad (4.3)$$

Este é um estimador que representa a proporção das ocorrências de *X*. A forma mais precisa de o obter consiste na exercitação do sistema com todos os elemento (*g*) da população *G* e observar os valores *x(g)* de *X* correspondentes a essas experiências, ou seja exercitar o sistema para o conjunto de todas as combinações de faltas e de actividade do sistema (cenários de carga).

Na maior parte dos sistemas este procedimento é impraticável, consequência da dimensão da população que constitui o conjunto de entrada. De forma a tornar o processo exequível, em termos práticos o sistema é submetido a um subconjunto dos elementos que constituem a população. A este processo designa-se de amostragem, devendo os elementos da amostra ser obtidos de forma aleatória geralmente num processo que envolve reposição, para garantir a independência estatística entre os mesmos. A amostra deve igualmente ter dimensão suficiente para garantir a sua representatividade. Considerando $X_i = \{x_1(g_1), x_2(g_2), \cdots, x_n(g_n)\}$ o número de réplicas de cada um dos elementos que constituem a amostra de dimensão *n* da variável aleatória *X*, o valor do factor de cobertura pode ser obtido através do seu valor esperado $\hat{c}$.

$$\hat{c} = \hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i \qquad (4.4)$$

Assim o factor de cobertura é obtido por um estimador que representa a média amostral $\hat{\mu}$ das observações do estado da saída do sistema. Ao contrário do estimador anterior que representa a média populacional μ e por conseguinte apresenta um valor constante, o estimador da média amostral varia com a amostra. Torna-se assim necessário utilizar indicadores que forneçam informação relativa ao rigor (precisão) ou à confiança das estimativas feitas com base nesses estimadores. A estatística fornece ferramentas que permitem caracterizar o rigor do estimador, nomeadamente através no conceito de estimação por intervalo. Neste caso, para cada estimador da média amostral, existe um intervalo de confiança para o qual é possível especificar uma probabilidade de o estimador se encontrar contido nos seus limites, designando-se estes últimos por limites de confiança.

Em sistemas de elevada complexidade pode tornar-se difícil obter uma amostra representativa, que englobe as faltas, sua incidência nas várias componentes do sistema, e que contemple diferentes cenários da sua actividade. Neste enquadramento a partição do espaço amostral através de um processo designado por estratificação, é uma opção que permite reduzir o número de entrada das experiências de injecção de faltas [Cukier99].

A latência como consequência da ocorrência de faltas e subsequente activação de mecanismos que as toleram é a par do factor de cobertura, um parâmetro de grande relevância na avaliação de sistemas tolerantes a faltas. A importância desta avaliação tem repercussões a dois níveis:

- **No projecto dos mecanismos de tolerância faltas**: A este nível a importância advém da necessidade de obter informação da rapidez com que os mecanismos de tolerância a faltas, detectam e efectuam o tratamento das mesmas. O controlo da latência associada à detecção dos erros é importante para assegurar que os erros não se propaguem às restantes partes do sistema, assegurando assim, que a falta seja tolerada no domínio lógico funcional;

- **No desenvolvimento de aplicações de tempo-real**: O atraso na execução das tarefas do sistema, resultante da execução prioritária das acções de recuperação de faltas, tem impacto na validade temporal das funções de um sistema de tempo-real. Desta forma uma caracterização da latência para os mais diversos cenários de faltas e de carga do sistema é relevante para desenvolvimento de aplicações de tempo-real.

Em termos de estimação da latência, a média amostral pode igualmente ser utilizada como estimador. Neste contexto o valor esperado para os tempos de reacção do sistema é obtido a partir das latências apresentadas para cada combinação do conjunto $\{f \times a\}$ que engloba as faltas que o afectam, e a actividade do sistema que resulta da sua carga.

Metodologias para estimação do factor de cobertura em sistemas tolerantes a faltas, avaliados com base na utilização de técnicas de injecção são apresentadas em [Powell95] [Cukier99].

Nesta dissertação o objecto de avaliação não está direccionado para a avaliação da cobertura fornecida pelos mecanismos de tolerância a faltas do PROFIBUS-DP, está antes centrada na:

- Identificação dos mecanismos que são activados aquando da ocorrência de faltas;
- Determinação da probabilidade dos eventos que apresentem maiores impactos no desempenho da rede;
- Avaliação da influência das latências associadas à activação desses mecanismos, em indicadores temporais relevantes para o desempenho de tempo-real da rede.

A inferência dos indicadores referidos cujo processo é intrinsecamente estocástico segue os mesmos formalismos da estatística relativos a obtenção de estimadores, e é efectuado de forma semelhante aos apresentados em [Powell95]. Contudo esta metodologia só refere os princípios para obtenção de estimadores com elevado grau de rigor, não indica a forma como devem ser efectuadas as experiências para que a amostra garanta os pressupostos que estão implícitos à obtenção de estimadores rigorosos. De facto, a avaliação da operação de um sistema por injecção de faltas é um processo complexo que muitas das vezes não é facilmente enquadrado nos problemas clássicos analisados na estatística.

Neste contexto, é utilizada uma metodologia não só para a obtenção dos estimadores, mas também para o dimensionamento, configuração e execução das experiências. Esta metodologia segue uma abordagem que é frequentemente utilizada na simulação de sistemas de eventos discretos [Tyszer99] [Banks96].

4.7.1 Metodologia Implementada no Ambiente de Injecção de Faltas

A avaliação por simulação do funcionamento de um sistema, consiste na estimulação das suas entradas com um conjunto de variáveis aleatórias, e na observação na saída do modelo que o representa das variáveis das grandezas de relevo para a avaliação. O ambiente de injecção apresentado nesta dissertação implementa o mesmo princípio, substituindo contudo o modelo por um sistema real, evitando assim os problemas inerentes à necessidade da sua validação e também de o calibrar, através do fornecimento de parâmetros de funcionamento que correspondam à real operação do sistema.

A utilização na entrada de variáveis aleatórias (números aleatórios) implica que as variáveis de saída do sistema sejam igualmente aleatórias. Assim, o resultado na saída, correspondente a um determinado valor na entrada, representa uma amostra da população que é constituída por todas as observações possíveis. Um aumento da dimensão da experiência, ou seja a exercitação da entrada por um maior número de eventos, aumentará na mesma dimensão o número de observação que constituem a amostra. Contudo apesar dos números aleatórios poderem representar a ocorrência de eventos estatísticamente independentes na entrada do sistema, isso não garante que as variáveis de saída sejam igualmente independentes. Em muitas situações a observação X_{i+1} pode em certo grau, ser influenciada pela observação X_i que a precedeu. Desta forma, a amostra pode ser constituída por elementos entre os quais se verifique a existência de autocorrelação.

Isto resulta numa redução da independência estatística entre elementos da amostra, contrariando assim, um dos pré-requisitos que devem ser considerados na obtenção do estimador, potenciando o risco deste se transformar num estimador enviesado.

A fase em que se processam as observações tem igualmente influência no resultado do estimador. Na execução de uma experiência o sistema passa por duas fases:

- **Fase transitória**, em que se verificam variações significativas na resposta do sistema;
- **Fase estacionária**, onde as saídas do sistema apresentam uma maior estabilidade (uniformidade).

Neste contexto, se o objecto de avaliação está centrado no regime estacionário a componente transitória pode assumir um peso importante e influenciar de forma significativa o resultado das experiências. Assim, no sentido de obter estimadores tão precisos quanto possível, estas questões devem ser correctamente endereçadas.

A diminuição do enviesamento do estimador devido ao efeito de autocorrelação entre elementos da mesma amostra, pode ser alcançado pela repetição da experiência com diferentes valores de entrada, num número de vezes

suficientemente elevado. Neste método, a autocorrelação continua a existir entre elementos da amostra de uma mesma experiência, mas as amostras são independentes quando consideradas entre diferentes experiências. Assim, a média amostral obtida a partir dos contributos das várias amostras pode ser considerado um estimador não enviesado.

A minimização da influência da componente transitória na análise de regime estacionário pode ser obtida com base em duas abordagens:

- Eliminar as observações que correspondem à fase de arranque da experiência – fase transitória;
- Assumir que, aquando do arranque do sistema o seu comportamento se aproxima do regime estacionário.

A aplicação da primeira estratégia tem como principal dificuldade a identificação de quais as observações que podem ser ignoradas. A dificuldade deriva do facto de muitas das vezes o ponto a partir do qual o sistema alcançou a fase estacionária não ser de fácil identificação.

Na aplicação da segunda estratégia, o principal obstáculo surge quando não existe conhecimento suficiente acerca da dinâmica do sistema.

Uma forma de inferir qual a fase em que se encontra o sistema consiste na análise da diferença entre as estimativas obtidas a partir de amostras sucessivas. No caso da diferença assumir um valor pequeno poder-se-á de forma aproximada considerar que o sistema se encontra em regime estacionário. Um método mais rigoroso para esta avaliação é baseado no teorema do limite central e recorre à utilização da média e do desvio padrão para estimar a componente transitória com auxílio de uma representação gráfica [Tyszer99].

O método implementado na recolha de informação obtida através da realização das experiências de injecção de faltas seguiu, a primeira das abordagens descritas anteriormente. Em termos operacionais, a infra-estrutura de comunicações é configurada e posta a funcionar somente com as estações que são objecto de avaliação. É assegurado uma margem de tempo para que o anel lógico estabilize o seu tempo de ciclo, nomeadamente no que respeita à carga que está a processar, e dos eventos de remoção do anel dos módulos que asseguram a operação do injector e do monitor (Fig.4.23). Estes módulos só fazem uso da rede PROFIBUS-DP durante a fase de sincronização de tempo-real (configuração da infra-estrutura), encontrando-se as interfaces PROFIBUS-DP desligadas do anel enquanto decorrem as experiências de injecção de faltas.

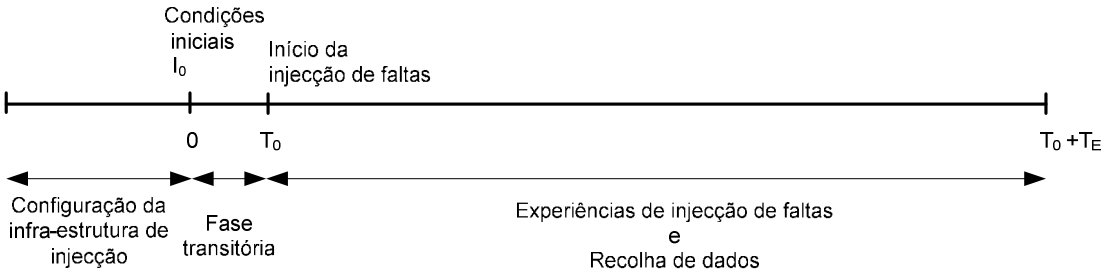

Figura 4.23 - Inicialização e recolha de informação das experiências de injecção.

Apesar da duração da experiência ser um parâmetro relevante para o dimensionamento e estabelecimento das correctas condições de avaliação, ela própria não resulta directamente de uma imposição que possa ser atribuída à natureza do problema. No entanto o seu dimensionamento deve levar em consideração aspectos como [Banks96]:

- A dimensão da experiência não deve ser demasiado pequena, sob o risco de os desvios impostos pelas condições iniciais poderem assumir um peso significativo no resultado da observação. O aumento da dimensão da experiência atenua de forma significativa a influência das condições iniciais no resultado final;
- Custos resultantes do aumento da dimensão das experiências, nomeadamente no que se refere às condições requeridas para suportar tais experiências.

O dimensionamento das experiências efectuadas no ambiente de injecção teve em consideração estes dois aspectos. Assim, foi utilizado um tempo de experiência T_E=3.5s. Considerando a dinâmica das comunicações na rede, este tempo assegura uma boa representatividade das experiências nomeadamente quanto à actividade no barramento e ao conjunto de faltas que afecta o tráfego desse mesmo barramento. Simultaneamente, esta dimensão está adequada às capacidades de armazenamento de informação disponibilizados pelos módulos que suportam as estações PROFIBUS-DP.

Existem vários métodos para obter estimadores relativos à operação de um sistema em regime estacionário [Tyszer99] [Banks96].

- **Método das replicações independentes**: Consiste na realização de um número de experiências estatisticamente independentes, que têm em comum a mesma duração e as mesmas condições iniciais, mas executadas de acordo com diferentes números aleatórios;
- **Método da Média dos Lotes (*Batch Means*)**: A estimação é efectuada com base na informação obtida numa única experiência de longa duração, da qual as observações são divididas em blocos adjacentes não sobrepostos;
- **Método Regenerativo**: Consiste na divisão da experiência num conjunto de sequências de experiências de dimensões mais

> reduzidas, que têm início em pontos particulares da experiência, cujo estado é independente de todo o historial do sistema. Estes pontos são designados de pontos de regeneração e são característicos de determinados sistemas denominados de regenerativos.

Uma comparação entre o método das replicações independentes e o método da média dos lotes, revela que este último é mais eficiente ao nível do tratamento da informação que resulta da componente transitória da experiência. No método da média dos lotes é rejeitada unicamente informação referente a uma única experiência, pelo contrário no método das replicações independentes isso ocorre para as *n* réplicas da experiência. Em contraste, a execução de uma única experiência no método da média dos lotes potencia o efeito de autocorrelação entre elementos da amostra obtida. Uma forma de reduzir este efeito requer uma escolha criteriosa da dimensão dos blocos, para que ocorra um aumento da independência das amostras que resultam dessa separação. Neste processo, é exigido um controlo adicional que permita monitorar a autocorrelação através da medida da covariância [Alexopoulos04] [Tyszer99]. Ao contrário, no método das replicações independentes as amostras das experiências são estatisticamente independentes entre si.

Analisando estes dois métodos na perspectiva da aplicação ao ambiente de injecção, o método da média dos lotes assenta a sua operação na execução de uma experiência de longa duração, o que coloca dificuldades ao nível da implementação, dada a forma como é processado o armazenamento da informação nos módulos do ambiente de injecção. Por seu lado, o método das replicações independentes permite ajustar a dimensão das experiências aos recursos que são disponibilizados. Acresce que devido à forma como o sistema de injecção executa as experiências (Fig. 4.23) a injecção de faltas e recolha de informação relevante só ocorre após o sistema se encontrar na fase estacionária, e que o tempo gasto para atingir esse estado é negligível relativamente ao tempo dispendido nas restantes componentes de configuração da infra-estrutura.

O método regenerativo produz resultados precisos. Contudo, apresenta dificuldades de implementação, nomeadamente no que respeita à identificação dos pontos de regeneração, e mesmo quando estes são identificados, nem sempre são alcançados devido a combinações de eventos que podem ocorrer no sistema [Tyszer99].

Tendo em consideração estes condicionalismos, a abordagem utilizada no ambiente de injecção de faltas está de acordo com o método das replicações independentes.

4.7.2 Método das Replicações Independentes

Assumindo que o conjunto $\{X_1, X_2, \cdots, X_n\}$ representa uma sequência de valores da variável de saída X de interesse para a análise do comportamento em

regime estacionário de um determinado atributo de um sistema, e que essa sequência converge para a função de distribuição da variável:

$$\lim_{n\to\infty} \Pr ob\{X \le x\} = F(x) \qquad (4.5)$$

Considere-se igualmente que a fase transitória da operação do sistema está identificada e que é estimado o ponto $\hat{n}_0$ a partir do qual a média dos elementos da variável tem valor aproximado à média μ. Assim, cada uma das *M* replicações independentes de experiências de dimensão *N*, especificadas no método das replicações independentes, têm uma média amostral de acordo com.

$$\hat{\mu}_m = \frac{1}{N-\hat{n}_0} \sum_{n=\hat{n}_0+1}^{N} X_{mn} \qquad (4.6)$$

Consequência da independência entre as *M* experiências (replicações), o resultado de cada uma das suas médias amostrais $\hat{\mu}_m$ é igualmente independente, pelo que o seu valor esperado $E[\hat{\mu}_m] \approx \mu$ pode ser considerado de forma aproximada um estimador não enviesado de μ.

$$\hat{\mu} = \frac{1}{M} \sum_{m=1}^{M} \hat{\mu}_m \qquad (4.7)$$

O intervalo de confiança associado a este estimador pontual pode ser determinado com base na variância amostral.

$$s^2(\hat{\mu}_m) = \hat{\sigma}^2(\hat{\mu}_m) = \frac{1}{M-1} \sum_{m=1}^{M} (\hat{\mu}_m - \hat{\mu})^2 = \frac{1}{M-1} \sum_{m=1}^{M} \left(\hat{\mu}_m^2 - \frac{M}{M-1} \hat{\mu}^2 \right) \qquad (4.8)$$

Os fundamentos da estatística estabelecem para amostras com distribuição de probabilidade não necessariamente normal, com média μ e variância σ^2, a variável padronizada associada a $\hat{\mu}$ é dada por:

$$Z = \frac{\hat{\mu} - \mu}{\frac{\sigma}{\sqrt{n}}} \qquad (4.9)$$

Representando $s^2(\hat{\mu}_m)$ da expressão (4.8) um estimador não enviesado da variância (σ^2) de $\hat{\mu}_m$, então o factor $s^2(\hat{\mu}_m)/M$ é uma estimativa da variância de $\hat{\mu}$. Assim, considerando o desvio padrão *s* como a raiz quadrada da variância amostral, a variável padronizada (4.9) pode ser determinada com base numa distribuição *t Student* de *M-1* graus de liberdade.

$$t_\mu = \frac{\hat{\mu} - \mu}{\frac{s(\hat{\mu}_m)}{\sqrt{M}}} \qquad (4.10)$$

É então possível definir desigualdades que de acordo com uma determinada probabilidade delimitam o valor que a variável padronizada (4.10) pode assumir.

$$\Pr ob\left\{ t_{M-1}\left(1-\frac{\alpha}{2}\right) \le \frac{\hat{\mu}_m - \mu}{\frac{s(\hat{\mu}_m)}{\sqrt{M}}} \le t_{M-1}\left(1-\frac{\alpha}{2}\right) \right\} = 1-\alpha \qquad (4.11)$$

A distribuição *t Student* é uma distribuição simétrica centrada em torno de zero, pelo que as inequações da expressão (4.11) podem ser reescritas em ordem a μ e apresentar o resultado na forma de um intervalo de confiança (4.12) que incluirá com probabilidade $1-\alpha$ o valor esperado de μ. Em que α representa em média a proporção das vezes que em que o intervalo de confiança não contém o parâmetro que se pretende estimar.

$$\mu \in \left[\hat{\mu} - t_{M-1}\left(1-\frac{\alpha}{2}\right)\frac{s(\hat{\mu}_m)}{\sqrt{M}},\ \hat{\mu} + t_{M-1}\left(1-\frac{\alpha}{2}\right)\frac{s(\hat{\mu}_m)}{\sqrt{M}} \right] \qquad (4.12)$$

Por vezes a análise está centrada na probabilidade de um determinado atributo associado a uma variável aleatória pertencer a um intervalo especificado. Inferir essa probabilidade pode ser o objecto de interesse, como é o caso nesta dissertação da determinação probabilidade de ocorrência de certos eventos quando o PROFIBUS-DP é afectado por faltas.

A sua determinação pode ser efectuada através de um processo semelhante ao anterior [Tyszer99].

Seja *I* o intervalo de interesse e que:

$$p_n = \Pr ob\{X_n \in I\} \quad e \quad p = \lim_{n\to\infty} p_n \qquad (4.13)$$

Para cada uma das *M* réplicas da experiência, obtém-se uma estimativa da proporção da ocorrência do evento de interesse.

$$\hat{p}_m \frac{x_m}{N - \hat{n}_0} \qquad (4.14)$$

Em que x_m é o valor da variável aleatória X_{mn} pertencente ao intervalo *I*, excluindo os eventos observados durante a fase transitória, $n > \hat{n}_0$. O valor esperado $\hat{p}$ das *M* réplicas independentes é aproximadamente igual à média *p* ($E[\hat{p}_m] \approx p$).

$$\hat{p} = \frac{1}{M}\sum_{m=1}^{M} \hat{p}_m \ ; \qquad (4.15)$$

e $s^2(\hat{p}_m)$ um estimador da variância de $\hat{p}_m$.

$$s^2(\hat{p}_m) = \frac{1}{M-1}\sum_{m=1}^{M} (\hat{p}_m - \hat{p})^2 \qquad (4.16)$$

De forma semelhante a variável padronizada (4.17) tem uma distribuição que pode ser aproximada a *t Student* com *M-1* graus de liberdade, e um intervalo de confiança definido pela expressão (4.18).

$$t = \frac{\hat{p} - p}{\frac{s(\hat{p}_m)}{\sqrt{M}}} \qquad (4.17)$$

$$p \in \left[\hat{p} - t_{M-1}\left(1 - \frac{\alpha}{2}\right)\frac{s(\hat{p}_m)}{\sqrt{M}},\ \hat{p} + t_{M-1}\left(1 - \frac{\alpha}{2}\right)\frac{s(\hat{p}_m)}{\sqrt{M}} \right] \qquad (4.18)$$

4.7.2.1 Dimensionamento do Número de Réplicas

O método das replicações independentes permite estabelecer um conjunto de procedimentos que devem ser cumpridos na execução das experiências que ocorrem no ambiente de injecção de faltas. As expressões representadas estipulam a forma como os dados das experiências são tratados durante a fase de análise para obter estimadores e respectivos intervalos de confiança. Não obstante a grande importância de toda a fundamentação anterior, o processo fica incompleto sem a identificação da forma como deve ser dimensionado o número de réplicas (experiências).

O número de experiências influência a precisão do estimador e por conseguinte está subordinado à precisão que neste se pretende depositar. A influência do número de experiências reflecte-se essencialmente nos seguintes parâmetros:

- **Na precisão do intervalo de confiança**, que varia na razão inversa das amostras;
- **No grau de confiança do intervalo**, ou seja na probabilidade do intervalo de confiança incluir o parâmetro estimado.

Na maior parte das vezes não existe à priori qualquer informação que permita efectuar uma estimativa do número de experiências a realizar. Nestas circunstâncias o seu dimensionamento não é um problema de resolução directa, obrigando ao recurso de uma experiência piloto que forneça informação acerca do estimador. Suportado nesta informação é possível projectar experiências complementares num processo iterativo até se obter o grau de precisão pretendido.

Uma forma de implementar o processo de controlo das iterações consiste em considerar um intervalo de confiança cuja largura seja uma fracção do valor do estimador objecto de análise.

Assumindo Δ como a largura do intervalo de confiança, dada pela diferença dos seus limites de confiança, e ε uma fracção do estimador pontual $\hat{\vartheta}$ da quantidade ϑ, em que Δ e $\hat{\vartheta}$ representam o valor obtido na experiência piloto, ou como resultado de experiências complementares, então pode-se considerar como critério de paragem a verificação da seguinte inequação:

$$\Delta \leq \varepsilon \hat{\vartheta} \qquad (4.20)$$

No caso do resultado da inequação não satisfazer a condição (4.20) procede-se ao dimensionamento de um número M_{x+1} de experiências complementares ao somatório das M_x anteriores. O dimensionamento é efectuado com recurso à expressão (4.21), sendo executado as vezes necessárias até que se obtenha a precisão desejada.

$$M_{x+1} = \left\lceil \left(\frac{\Delta}{\varepsilon \hat{\vartheta}} \right)^2 \sum_x M_x \right\rceil - \sum_x M_x \qquad (4.21)$$

4.8 Síntese

Com o objectivo de fornecer uma ferramenta de suporte à avaliação do funcionamento do PROFIBUS-DP em cenários excepção, foi desenvolvida uma infra-estrutura de injecção de faltas de acordo com os seguintes pressupostos:

(i) Apresentar uma arquitectura modular;

(ii) Seguir uma estratégia de recolha de dados orientada à construção de um repositório de informação;

(iii) Ser capaz de implementar uma metodologia de realização de experiências de acordo com fundamentação matemática, que assegure o rigor e representatividade estatística das amostras obtidas nas experiências de injecção de faltas.

O desenvolvimento modular da infra-estrutura visou a sua adaptação aos requisitos impostos pelas tarefas de injecção de faltas e da recolha de dados num ambiente distribuído. Da mesma forma a estrutura modular permitiu agrupar módulos segundo uma organização hierárquica, dividindo assim a tarefa de injecção em várias subfunções. Esta subdivisão do problema de injecção acomodou muita da heterogeneidade dos requisitos subjacente a esta função, tendo sido possível especificar e implementar módulos do controlo de injecção que se revelaram bastante eficientes ao nível da controlabilidade e reprodutibilidade das experiências, assim como, na capacidade de injecção de faltas com elevada resolução temporal.

Num contexto de avaliação que tem como objecto a avaliação modos de operação que não são conhecidos à priori, e da verificação do seu impacto em vários indicadores de comportamento da rede, a definição de uma estratégia para a recolha e processamento da informação é relevante.

Na concepção da infra-estrutura de injecção de faltas, optou-se por não seguir uma filosofia assente na obtenção directa dos estimadores que avaliam o funcionamento da rede. Esta opção é justificada pelo facto do tipo de avaliação que se pretende efectuar requer um padrão experimental que evolui com o

conhecimento obtido. Neste processo numa primeira fase são identificados os modos de operação e com base nestes são especificados estimadores quer para a determinação da probabilidade de ocorrência quer para caracterização da influência destes no comportamento da rede.

Acresce que o processo de avaliação é inerentemente estocástico pelo que está sujeito a um conjunto de regras no dimensionamento das experiências para salvaguarda da representatividade dos resultados obtidos, o que se traduz na necessidade de efectuar um elevado número de experiências. A realização de experiência para a obtenção de um estimador específico, apresenta inconvenientes importantes designadamente por poder implicar a repetição de experiências já realizadas. Este é um processo extremamente dispendioso da perspectiva temporal. Igualmente esta abordagem dificulta a correlação de informação nomeadamente para despiste de falsas hipóteses, muitas das vezes necessária neste tipo de experiências.

A estratégia implementada consistiu assim na constituição de um repositório de informação que pode ser acedido em qualquer altura e sobre o qual podem ser efectuadas as mais diversas análises. Esta abordagem não requer repetições de experiências das quais já foram obtidos os dados reduzindo assim de forma muito significativa o tempo da componente experimental. O inconveniente principal desta abordagem está relacionado com uma necessidade elevada de recursos para armazenamento de informação.

Em último, a estrutura da infra-estrutura teve que garantir os mecanismos de suporte essenciais à realização de experiências de acordo com as condições impostas pela fundamentação matemática que garante a validade dos resultados obtidos. Neste contexto, a infra-estrutura incluiu recursos quer ao nível da capacidade de armazenamento quer ao nível da automatização de procedimentos para repetição de pequenas experiências de injecção de faltas que facilmente se adaptem a uma execução que esteja de acordo com o método das replicações independentes.

Esta página foi intencionalmente deixada em branco

Capítulo 5

PROFIBUS-DP: Avaliação do Desempenho em Cenários de Faltas

5.1 Introdução

A operação das redes de comunicações, e mais concretamente o seu desempenho quando afectadas por faltas, está de uma forma geral intrinsecamente associado às características do protocolo e em particular das camadas inferiores que o compõem.

Os níveis inferiores dos protocolos estão incumbidos de entre outras tarefas, de tratar erros de comunicações e de resolver conflitos que ocorram ao nível do meio físico. Neste enquadramento, uma parte substancial da componente temporal da latência e do *jitter* introduzido nas comunicações ocorre precisamente devido à necessidade de execução dos mecanismos responsáveis por assegurarem as referidas tarefas. Quando se menciona os níveis inferiores refere-se concretamente à camada de ligação de dados.

No PROFIBUS-DP muitas das funções referidas são efectuadas precisamente nessa camada que toma a designação *Fieldbus Data Link* – FDL. Acresce que a forma como as estações estão organizadas na rede, torna sensível a execução destas tarefas. Sobretudo, se a forma como foram implementadas seja susceptível de apresentar padrões de operação que resultem na activação combinada de vários mecanismos em consequência de diferentes modos de faltas. Nestas condições o tempo que estes mecanismos possam estar activos pode traduzir-se em elevadas latências afectando assim de forma significativa a resposta de tempo-real da rede.

Do exposto fica patente a necessidade de compreender os mecanismos que regem a operação da FDL e de avaliar o seu comportamento para os mais

diversos cenários de excepção que possam ocorrer derivado a faltas com incidência no barramento.

5.2 Camada de Ligação de Dados

Os princípios organizacionais e funcionais do PROFIBUS-DP foram já alvo de uma abordagem em §2.3. Nesta secção o protocolo é revisitado no sentido de aprofundar aspectos que são relevantes para a compreensão e suporte à avaliação da operação do PROFIBUS-DP na presença de faltas.

A camada de ligação de dados (FDL) assegura um conjunto de serviços de comunicação que suportam toda a transferência de informação do utilizador sem que este interfira nos serviços base de comunicação, nem se tenha que preocupar com aspectos relacionados com a gestão da rede ou do controlo de erros. Neste contexto, a descrição da FDL pode ser centrada nos três aspectos seguintes:

- Controlo de acesso ao meio;
- Suporte à transmissão de dados;
- Integridade da informação;

5.2.1 Controlo de Acesso ao Meio

O controlo de acesso ao meio regula toda a actividade da estação na rede, bem como a forma como esta se processa. No PROFIBUS-DP é implementado entre estações activas, um método de controlo baseado na descentralização da gestão do acesso ao meio, regendo-se as comunicações que são estabelecidas com as estações passivas de acordo com uma estratégia centralizada baseada numa configuração mestre escravo.

De acordo com a norma da rede [EN96a], a execução destes métodos são regidos segundo a máquina de estados da figura 5.1. Cada estado operacional da estação é definido para vários cenários de operação, englobando estados relativos a eventos de inserção ou de remoção de estações do anel, da mesma forma que prevê acções para tolerar a ocorrência de faltas.

A descrição que se segue da máquina de estados da FDL, refere os estados operacionais focando essencialmente os cenários que são mais representativos de uma operação em regime estacionário, ao qual vagamente são referidas excepções. Isto resulta de uma opção em favor de uma descrição mais fluida. Uma descrição de todas as funcionalidades implementadas pelo controlo de acesso ao meio encontra-se na norma da rede [EN96a].

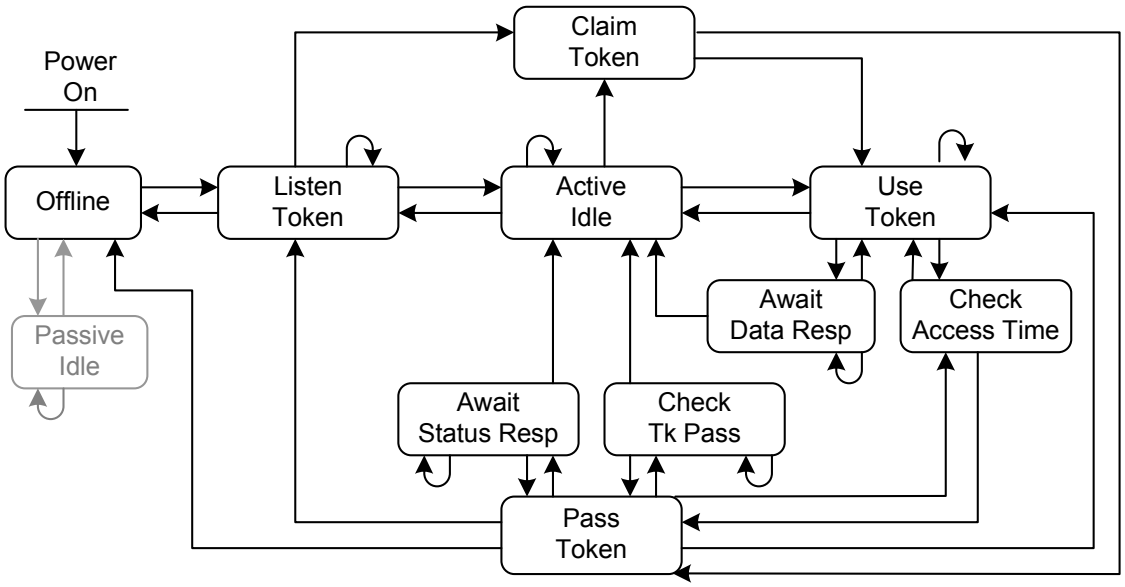

Figura 5.1 - Máquina de estados do controlo de acesso ao meio de uma estação activa PROFIBUS-DP.

Offline

Estado operacional no qual a estação ainda não está do ponto de vista lógico conectada à rede. A FDL estabelece o seu estado de operação em *offline* logo após a ocorrência de um dos seguintes eventos: Arranque da estação, ou ocorrência de um erro severo. Permanece neste estado até à completa parametrização da FDL.

Listen Token

Após ter sido completamente parametrizado a FDL deve evoluir para o estado *Listen Token*. Neste estado a FDL deve monitorizar o tráfego do barramento de forma a identificar a estações que se encontram a operar no mesmo, e assim, construir a lista das estações activas (LAS). Este processo é efectuado com base no conteúdo do *token* enviado por cada uma dessas estações. A lista fica completa após ter sido observado dois ciclos idênticos e de forma consecutiva. A partir desse momento a estação está pronta a entrar no anel, devendo para tal aguardar a recepção de uma trama *Request FDL Status* endereçada pela estação que a precede no anel.

Active Idle

Neste estado, a FDL inserida no anel lógico aguarda a recepção de tramas que lhe sejam endereçadas, sem contudo por sua iniciativa poder exercer actividade no anel. Quando lhe são endereçadas tramas com dados (tramas de acção) deve proceder em conformidade com o que lhe é enviado. Quando lhe é endereçada pela estação que o precede um *token* válido, a FDL ganha acesso ao barramento para poder exercer a sua actividade.

Claim Token

Se num dos estados anteriores (*Listen Token* e *Active Idle*) a FDL não identificar qualquer actividade no barramento durante um limite máximo estipulado (*timeout*), então deve evoluir para o estado *Claim Token*. Neste estado, deve gerar um *token* e proceder à re-inicialização ou à inicialização do anel lógico. No primeiro caso, a LAS existente até ao momento da detecção do *timeout* permanece válida, e a estação pode de imediato evoluir para o estado *Use Token*. O segundo caso ocorre quando o timeout se verifica na estação com menor endereço, e neste cenário há necessidade de gerar uma nova LAS.

Use Token

Após a recepção de um *token* ou da ocorrência de uma re-inicialização a FDL estabelece operação no estado *Use Token*. Neste estado operacional a estação pode estabelecer o ciclo de mensagens com as estações passivas da sua esfera de controlo. Deve para tal respeitar as regras do protocolo relativamente ao tempo e prioridades das mensagens, sendo que o protocolo garante pelo menos o envio de uma mensagem de alta prioridade.

Await Data Response

Na posse de um *token*, sempre que procede ao envio de uma trama de acção (*action frame*), a FDL deve aguardar o tempo de recepção (*slot time*), suficiente para permitir a recepção da resposta da estação destinatária. Após concluída a transacção, podendo nesta estarem incluídas retransmissões devido a erros ou ausência de resposta da estação endereçada, a FDL deve retomar o estado *Use Token*.

Check Access Time

O limite máximo de actividade que uma estação que detém o *token* pode exercer no barramento é regulada pelo *Token Holding Time*. Assim, antes de se proceder ao envio de uma nova trama deve ser verificado se ainda dispõe desse tempo. Caso este tenha expirado a FDL deve iniciar os procedimentos de passagem do *token*.

Pass Token

No estado *Pass Token* a FDL tenta passar o *token* à estação que o sucede no anel lógico. No caso de se tratar da única estação activa no anel, a FDL deve enviar um *token* endereçado à própria estação.

Check Token Pass

Após o envio do *token* a FDL deve evoluir para o estado *Check Token Pass* onde deve confirmar a passagem *token*. Neste processo aguarda um *slot time*, monitorizando o barramento à espera que a estação para a qual foi endereçado o *token* inicie a actividade. A ocorrência de actividade dentro do período definido pelo *slot time* faz suspender a actividade no barramento da FDL e evoluir para o estado *Active Idle*. A inexistência de qualquer actividade no barramento durante o período estipulado provocará o início de nova tentativa de passagem do *token* no estado *Pass Token*.

Await Status Response

A FDL estabelece operação neste estado quando não conhece a estação que lhe sucede no anel. Este cenário pode ocorrer devido à necessidade de proceder inicialização do anel, ou durante pesquisa de estações na gama de endereços abrangida pela GAP. Neste processo, a FDL aguarda pelas confirmações às tramas *Request FDL Status* durante um *slot time*. Caso receba qualquer outra trama que não seja resposta ao serviço enviado a FDL, entra no estado *Active Idle*.

O controlo de acesso ao meio implementa um conjunto de componentes cuja função é importante para estabelecer o estado de operação. Serviços de comunicação de gestão da rede, geração de uma lista das estações activas no anel lógico, e temporizadores, são componentes que são utilizados de forma integrada pelo controlo de acesso ao meio da FDL na definição dos vários estados de operação.

5.1.1.1 Serviços de Comunicação de Gestão

Os serviços de comunicação de gestão têm a particularidade de não contribuírem directamente para o processo de transferência de dados que constitui a essência da comunicação entre estações, mas antes estão orientados para funções que permitem coordenar a operação das estações na rede. Estes serviços são constituídos por duas tramas.

Token

O *token* é uma trama que tem como principal função transferir a autorização de acesso ao meio (barramento). Esta função envolve uma transacção entre duas estações, o detentor do *token* e a estação candidata à sua recepção. Desta forma, a estrutura da trama reflecte isso mesmo e nesse contexto é constituída unicamente por três *bytes* (caracteres de UART) (Fig. 5.2):

- Identificador da trama –SD *Start Delimiter*;
- Endereço da estação de destino – DA *Destination Address*;
- Endereço da estação de origem – SA *Source Address*.

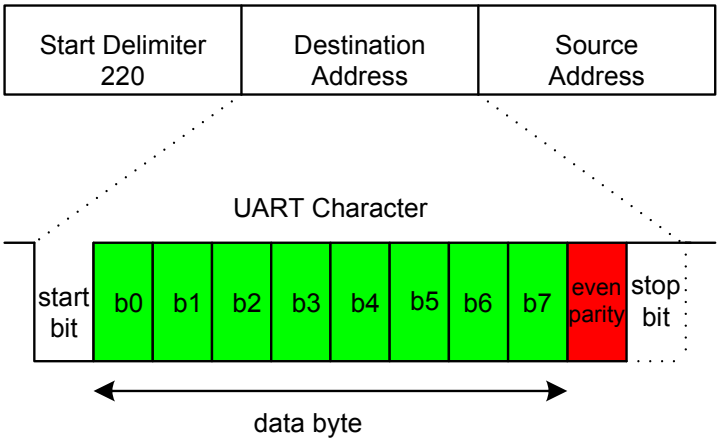

Figura 5.2 - Estrutura do *token*.

Request FDL Status

A trama *Request FDL Status* é utilizada pela FDL para obter o estado operacional das demais estações (Fig. 5.3). A sua utilização pode ser efectuada em dois contextos distintos:

- **A pedido do utilizador**, para mapear o estados de todas as estações que fazem parte da rede, incluindo estações passivas numa lista designada de *Live List*;
- **Por iniciativa da FDL**, e sempre que tenha expirado o tempo para actualização da GAP-List, uma gama de endereços em que cada estação activa é responsável por gerar eventos de actualização com o objectivo de pesquisar estações e proceder à sua inserção no anel lógico. Neste caso, a trama é enviada no espaço de endereçamento compreendido entre o endereço da estação detentora do *token* e o endereço da estação que a sucede na lista das estações activas.

  Sempre que uma estação recebe esta trama responde com o seu estado operacional codificado no campo *Function Code* (FC). Assim, a sua resposta representa um dos seguintes estados:

  - ***Not Ready***: ocorre quando uma estação não está pronta para entrar no anel lógico, nomeadamente por ainda não ter construído a sua lista de estações activas;
  - ***Ready to Enter Logical Ring***: A estação aguarda inserção no anel lógico;
  - ***Station in Logical Ring***: Se a estação se considera membro do anel lógico;

- ***Slave Station***: quando o endereço inquirido é ocupado por uma estação passiva.

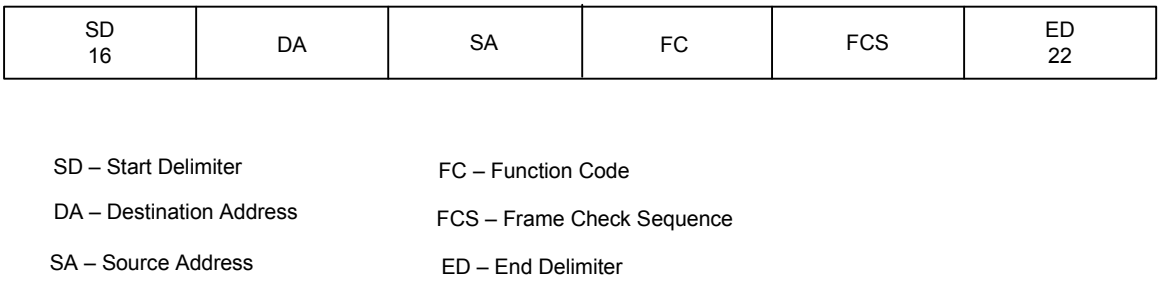

Figura 5.3 - Estrutura da trama que suporta o serviço *Request FDL Status*.

5.1.1.2 Lista de Estações Activas

A eficiente execução das acções de controlo está de uma forma genérica condicionada por uma correcta percepção da envolvente onde o sistema de controlo actua. Essa percepção tem de ser consistente e por conseguinte muitas das vezes não pode cingir-se unicamente à imagem que obtém instantaneamente da actividade do sistema, uma vez que esta pode estar afectada por erros ou interferências que se traduzam em factores desviantes ao seu desejável funcionamento.

Em determinados cenários, isto requer o recurso a um capital de informação que actue como memória do sistema e assim permita suportar e validar tomadas de decisão nas acções de controlo.

No PROFIBUS-DP a LAS fornece uma imagem da constituição global do sistema no que se refere às estações que concorrem a um mesmo recurso – acesso ao barramento. Assim, cada novo evento observado no sistema pode ser avaliado de acordo com as regras do protocolo e essa avaliação pode ser sustentada por informação que a contextualize com o estado do sistema.

De uma forma geral, grande parte das operações que envolvem a inserção de estações no anel lógico, passagem de *token*, ou na definição de estado de operação da FDL está directamente condicionada à memória do sistema que é fornecida pela LAS. Neste contexto, a LAS é um dos componentes que assume particular relevância no suporte à operação do protocolo.

5.1.1.3 Temporizadores

Parte significativa das operações efectuadas pela FDL é executada num contexto temporal restrito. Uma componente dessas operações decorre de forma síncrona com eventos observados no barramento de comunicações, ou dentro de intervalos temporais bem definidos. De forma a monitorizá-los a FDL implementa um conjunto de temporizadores.

Todos esses temporizadores têm por resolução a duração da transmissão de um *bit* T_{bit}, ou seja o inverso da taxa de transmissão. Em função do seu valor são derivados os tempos das tarefas efectuadas na FDL, e que de forma extensiva se repercutem nos tempos observados ao nível do utilizador.

A configuração base dos temporizadores que monitorizam a actividade do barramento e que servem de regulação para a operação da FDL inclui:

- ***Idle Timer:*** Temporizador para verificar a sincronização do hardware dos receptores. Assegura um tempo de barramento no estado *idle* suficiente para a sincronização do hardware dos receptores;
- ***Syn Interval Timer:*** Temporizador para monitorização do meio de transmissão. No caso em que no hardware do receptor não ocorra nenhuma sincronização dentro do tempo contado no *syn timer*, este evento é interpretado como uma falta de natureza permanente no canal de recepção;
- ***Slot Timer*****:** Temporizador que estipula o tempo máximo no qual deve ser recebida a resposta aos serviços de comunicação, ou para que se verifique actividade no barramento após a transmissão de um *token*. Caso o tempo expire antes que um dos eventos ocorra, procede-se à repetição do serviço;
- ***Timeout Timer****:* Temporizador para monitorizar a actividade das estações activas no barramento. Este temporizador é múltiplo do *slot time*.

A operação da rede está assim subordinada à verificação das condições temporais impostas por estes temporizadores, o que tipifica a rede de comunicação num perfil de comunicação síncrono, caracterizado por um padrão de funcionamento no qual o barramento apresenta sempre máxima utilização, independentemente da carga gerada na camada do utilizador e suas características temporais.

5.1.2 Suporte à Transmissão de Dados

O PROFIBUS-DP utiliza dois tipos de serviço para transmissão de dados:

- Serviços confirmados – *Send and Request Data* (SRD);
- Serviços não confirmados – *Send Data with No acknowledge* (SDN).

Nos serviços confirmados os dados são transferidos em ciclos entre estação activa e passiva, ou seja a estação activa transfere dados de saída para a estação passiva e esta se possuir entradas responde transferido os dados de entrada. Quando a estação passiva representa um dispositivo de saída simples, confirma o serviço através de uma resposta curta codificada num *byte*.

A estação activa contacta com todas as estações passivas da sua esfera de influência, decorrendo o processo numa estação de cada vez e enquanto existir tempo de posse de *token* (*hold token*). Caso o tempo de *token* expire antes de concluída a ronda pelas estações passivas, a estação activa concluirá o processo durante a recepção de *token* subsequentes. A frequência do contacto com as estações passivas – *polling*, é ditada pela geração de pedidos de comunicação na camada do utilizador.

Nos serviços não confirmados a estação activa utiliza um processo de difusão (*broadcast*) para fazer chegar a informação a todas os seus destinatários. Neste caso não existe lugar a qualquer confirmação por parte dos receptores do serviço.

O PROFIBUS-DP disponibiliza dois tipos de tramas para suportar os serviços de comunicação. Uma, com comprimento fixo do campo de dados (8 bytes). Uma outra, onde esse comprimento é variável (Fig. 5.4).

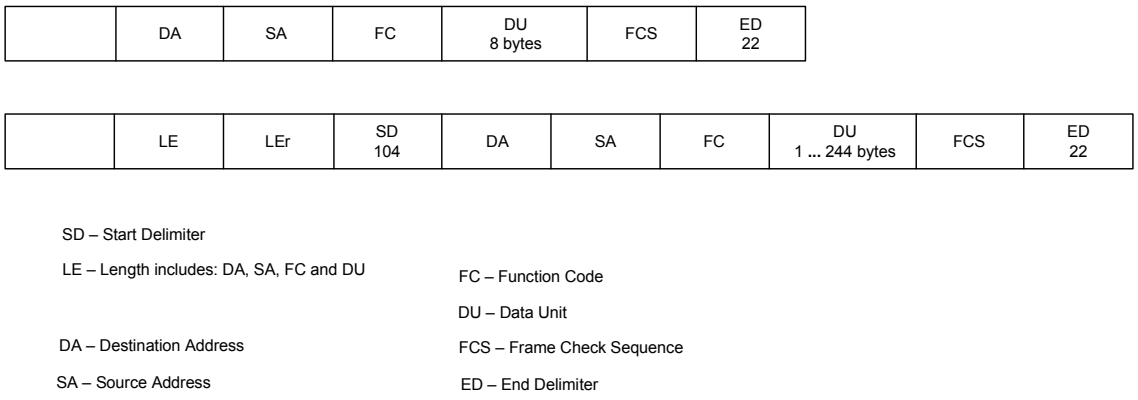

Figura 5.4 - Estrutura das tramas que suportam os serviços de transmissão de dados.

5.1.3 Integridade da Informação

Uma das causas mais frequentes da ocorrência de erros nas comunicações sucede no canal de comunicações, e resulta da degradação da qualidade dos sinais onde está codificada a informação. Por vezes, as fontes desta degradação assumem intensidade suficiente para tornar o processo de descodificação impossível, ou até mesmo de, pontualmente, conseguir alterar o seu significado.

A FDL do PROFIBUS-DP utiliza dois métodos para detectar erros deste tipo:

- Um primeiro baseado num mecanismo de integridade do *byte*, que é implementado com recurso ao bit de paridade da UART;
- Um segundo baseado num código que atesta a integridade da trama – *Frame Check Sequence* (FCS).

O método do *bit* da paridade consiste na atribuição do valor 0 ou 1 a esse *bit*, de forma a perfazer com os restantes um número par ou ímpar. No PROFIBUS-DP as UART's são configuradas para fornecerem paridade par. Assim a UART emissora deve atribuir um valor ao *bit* de paridade que transforme a soma da

cadeia de bits num número par. Durante a recepção, a não verificação do número par é assinalada como um erro de paridade.

Este método contudo oferece uma cobertura na detecção de erros limitada, que não abrange basicamente os dois cenários de faltas seguintes:

- Combinações de alterações em número par de *bits* do bloco de dados excluindo o *bit* de paridade (Fig. 5.5);
- Combinações de alterações em número ímpar de *bits*, em que um deles é o *bit* de paridade.

No *token*, o bit de paridade constitui o único mecanismo de detecção de erros implementado na trama. Neste contexto, o protocolo de acesso ao meio deve ser capaz de acomodar alguma inconsistência que possa ser transmitida por via de tramas com erros, cuja integridade foi dada como correcta pelo mecanismo de paridade.

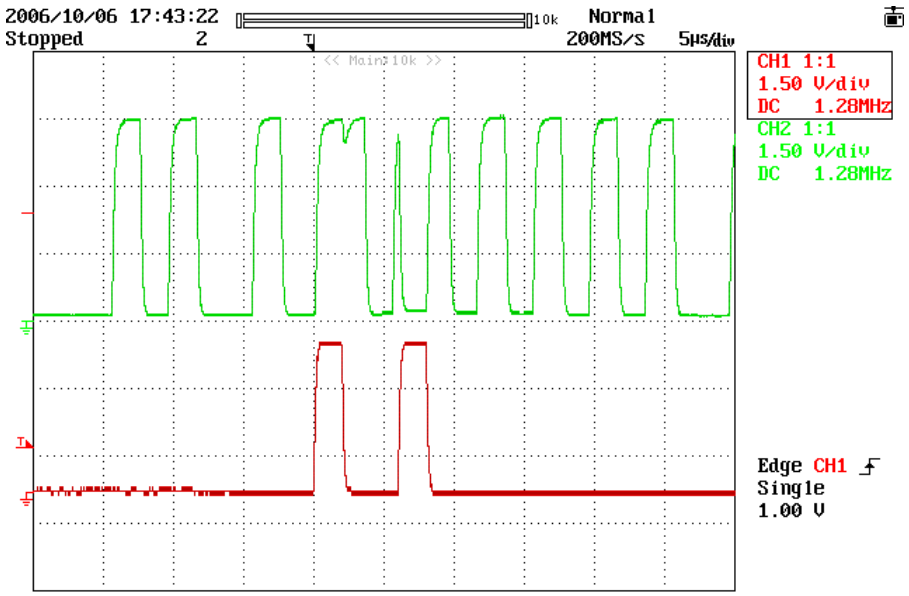

Figura 5.5 - Erro não detectado pelo mecanismo de paridade.

Nas restantes tramas do protocolo, para além do mecanismo de paridade é adicionado um *byte* de controlo (*Frame Check Sequence* - FCS). O conteúdo desse *byte* é calculado de acordo com códigos de *Hamming*, um tipo de código utilizado para encriptação, ou na detecção de erros de comunicação [Wells98]. O PROFIBUS-DP determina o FCS com base em códigos *Hamming* com distância 4. O processo é aplicado a todos os bytes das tramas com excepção dos campos com os códigos que as identificam e as terminam.

No domínio das comunicações a escolha dos códigos de detecção de erros envolve a tomada de um conjunto de decisões nas quais estão considerados aspectos como [Koopman04]:

- A distância de *Hamming* a utilizar;

- Qual a dimensão das mensagens que irão ser protegidas;
- A dimensão que os códigos assumem nas mensagens.

Para a tomada de decisão concorrem por vezes factores que são antagónicos no que diz respeito às soluções, o que em muitas das vezes obriga a uma solução que privilegia alguns aspectos do problema em detrimento de outros.

No PROFIBUS-DP a escolha recaiu sobre um código de detecção de erros que ocupa unicamente 1 byte na trama, e que provavelmente resultou de uma decisão deste tipo. Naturalmente, esta solução não oferece o mesmo grau de cobertura de outros códigos CRC que são utilizados em outras redes, um factor que pode assumir maior importância em aplicações mais específicas. Encontram-se nesta classe de aplicações aquelas que apresentam requisitos de segurança que estejam directamente relacionados com a integridade da informação.

Uma avaliação da eficiência dos mecanismos de detecção de erros do PROFIBUS-DP é apresentado em [Willig99a]. A avaliação foi efectuada para condições específicas nas quais os eventos (erros) são equiprováveis, com probabilidade constante e que produzem uma inversão do valor da informação. A probabilidade de detecção de erros é quantificada e são assinaladas algumas insuficiências dos mecanismos, assim como, é efectuada uma comparação com resultados obtidos a partir da utilização de CRC de 32bits. Não obstante esta comparação confrontar dois códigos muito diversos, nomeadamente no comprimento que apresentariam na trama, ou seja, 1 byte no PROFIBUS-DP em relação aos 4 bytes que representaria a outra solução, a comparação evidencia uma relativa baixa taxa de cobertura do método de detecção de erros implementado pelo FCS do PROFIBUS-DP. A análise apresenta uma probabilidade de não detecção da inversão de 4 bits da ordem de $9 \cdot 10^{-10}$.

Apesar destes aspectos menos positivos que são apontados aos mecanismos de detecção de erros dos serviços de comunicação do PROFIBUS-DP, quando analisados na perspectiva da integridade da aplicação estes não são por si factores limitativos para o bom funcionamento das aplicações. De facto, se a aplicação apresentar restrições importantes quanto à integridade da informação estes podem ser assegurados pelo perfil de aplicação PROFIsafe, que torna o processo totalmente independente dos mecanismos implementados na FDL garantindo níveis de integridade SIL3 [Stripf05].

Contudo, esta aparente fragilidade do mecanismo de detecção de erros numa trama que assume particular importância para a gestão do anel lógico, como é o *token* e todo o padrão síncrono de operação da FDL, baseado em *timers* e na troca e monitorização destas tramas, suscita outras questões. Questões essas, que não estão relacionadas com a integridade da informação transferida entre estações, mas antes com questões de cariz temporal que derivam de uma redução de desempenho. Estas por sua vez podem afectar a disponibilidade de sistemas com elevados requisitos de segurança suportados pelo PROFIsafe, ou tornar pontualmente a rede incapaz de garantir em determinados cenários resposta de tempo-real.

Neste contexto torna-se pertinente:

- Avaliar quais os efeitos na operação da FDL de erros que ocorram no barramento, identificando e caracterizando modos de operação associados às faltas;
- Avaliar a susceptibilidade do PROFIBUS-DP para apresentar determinados modos de operação na presença de faltas, e qual o contributo destes no aumento da latência das comunicações;
- Avaliar de uma forma global os efeitos dos erros na operação de tempo-real da rede de comunicações.

5.3 Caso de Estudo

A operação do PROFIBUS-DP é pautada por duas componentes: uma de gestão e manutenção da rede, e uma outra que assegura a comunicação de dados. Quando a avaliação da operação é centrada nas características temporais da rede, essa avaliação está genericamente orientada para os efeitos da resposta dos serviços de comunicações no nível da aplicação. Contudo uma avaliação da rede fora do contexto do contributo da componente de gestão, pode não representar correctamente a dinâmica das comunicações sobretudo em cenários de faltas.

De facto, os serviços de comunicação são unicamente atómicos ao nível da FDL, uma vez que após terem sido desencadeados não existe outra actividade intercalar que não seja eventuais repetições do serviço. Quando os serviços de comunicação são desencadeados pela aplicação, esta atomicidade deixa de ser garantida, sendo frequente a existência intercalada de serviços de gestão. Assim, a componente de gestão assume também ela um papel importante no desempenho temporal da rede. Neste contexto, a sua caracterização é fundamental para a percepção dos factores que alteram o desempenho da rede assim, como, o peso relativo que estes assumem na sua operação.

Nas secções que se seguem é efectuada uma avaliação à operação da rede englobando as várias vertentes da sua operação. Numa primeira fase a avaliação é orientada à identificação de modos de operação da FDL que resultam da ocorrência de faltas. Numa segunda fase é efectuada uma avaliação global do desempenho na presença de faltas do PROFIBUS-DP.

5.3.1 Condições Gerais de Avaliação

A avaliação da rede, como é proposta nesta dissertação, é suportada por um conjunto de ensaios que têm por base cenários nos quais a operação da rede é perturbada por faltas. Estas perturbações, em conjugação com outros parâmetros

associados ao funcionamento da rede (ex. carga) formam o conjunto das variáveis de entrada do sistema.

Aquando do desenvolvimento da infra-estrutura de injecção de faltas, foram previstos mecanismos que permitem assegurar a injecção de diversos modelos de faltas. Nesse processo, foi dado especial ênfase às faltas de natureza transitória incidindo de forma particular no modelo de inversão de bit (*bit-flip*). Contudo, este modelo por si só não assegura as condições necessárias a uma representação capaz, que descreva quer numa perspectiva funcional, quer numa perspectiva formal o mecanismo de perturbação da rede. O modelo que descreve a ocorrência das faltas no ambiente de operação da rede é um componente fundamental para completar essa representação.

Estes como outros eventos têm interferência na evolução do estado de um sistema. A forma como estes ocorrem tem frequentemente um cariz intrinsecamente estocástico, pelo que a sua representação é essencialmente suportada em variáveis com uma determinada função de probabilidade ou função de densidade de probabilidade.

Neste contexto, o comportamento do objecto de avaliação, ou parte específica deste, deve ser analisado de forma a se proceder à selecção de uma função com um padrão de distribuição que melhor represente a componente por ela modelada. Tipicamente existem duas grandes classes de distribuições:

- **Distribuições contínuas**: nas quais as variáveis aleatórias assumem um qualquer valor dentro de um intervalo;
- **Distribuições discretas**: nas quais as variáveis aleatórias assumem unicamente valores inteiros.

Dentro de uma mesma classe existe uma grande diversidade de distribuições que têm como elementos diferenciadores as formas apresentadas pelas suas funções, assim, como dos parâmetros da distribuição (ex. parâmetro de localização – média e parâmetro de dispersão – variância). Outras particularidades apresentadas pela função de distribuição revelam-se de grande importância em determinados cenários. Uma dessas particularidades está associada à propriedade que caracteriza a ausência de memória entre os elementos da função de distribuição. Uma função que possui esta propriedade é designada por distribuição sem memória (*memoryless*), e permite associar uma probabilidade à ocorrência de eventos futuros sem que seja necessário conhecer o que ocorreu no passado, ou seja pelo facto de um evento ainda não ter ocorrido isso não tem qualquer significado nem influência no tempo que resta para a sua ocorrência.

$$\Pr\left(X > s+k \,\middle|\, X > k\right) = \Pr\left(X > s\right) \qquad \Rightarrow s,k > 0 \tag{5.1}$$

Existe unicamente duas distribuições que apresentam esta propriedade:

- A distribuição exponencial no domínio contínuo;
- A distribuição geométrica no domínio discreto.

A distribuição exponencial é caracterizada pela função densidade de probabilidade:

$$f(x)=\begin{cases}0, & se\ x<0\\ \lambda\cdot e^{-\lambda\cdot x}, & se\ x\geq 0\end{cases} \tag{5.2}$$

E respectivos parâmetros de localização e de dispersão.

$$\mu=\frac{1}{\lambda} \tag{5.3}$$

$$\sigma^2=\frac{1}{\lambda^2} \tag{5.4}$$

A distribuição exponencial é usada em diversas áreas de aplicação nomeadamente na modelação do comportamento de sistemas quando os eventos que se pretendem modelar apresentam um padrão acentuadamente aleatório.

A fiabilidade é uma das áreas onde a distribuição exponencial, assim como as distribuições gama e Weilbull, das quais a primeira é um caso particular são usadas para modelar falhas de sistemas. A incidência dos eventos ou o tempo de vida de um componente é representada por pelo parâmetro λ, que no último caso representa a taxa de avarias do componente.

A distribuição geométrica é caracterizada pela função de probabilidade:

$$f(x)=\begin{cases}p\cdot(1-p)^{x-1}, & \Rightarrow x\in\aleph\\ 0, & caso\,contrário\end{cases} \tag{5.5}$$

E respectivos parâmetros de localização e de dispersão.

$$\mu=\frac{1}{p} \tag{5.6}$$

$$\sigma^2=\frac{(1-p)}{p^2} \tag{5.7}$$

A distribuição geométrica relaciona o número de experiências de Bernoulli que são necessárias realizar para que um processo mude de estado. Na função (5.5), *x* é um elemento da variável aleatória *X* que representa o número de experiências ao fim do qual se obtém sucesso (ou mudança de estado), considerando a probabilidade de sucesso *p*.

O processo de avaliação da operação do protocolo requer a utilização de um modelo que descreva a alteração dos sinais do barramento, resultado de interferências de natureza electromagnética. Este é um processo físico que pode ter origem diversa, resultado designadamente da operação de equipamento industrial à qual está associada a comutação de cargas envolvendo elevados gradientes de corrente eléctrica, ou a processos naturais como são as descargas atmosféricas. A ocorrência destes eventos tem uma natureza marcadamente aleatória, e um padrão da interferência complexo e de difícil modelação matemática.

O grau das interferências nos equipamentos que se encontram próximos da fonte emissora depende da eficiência das protecções implementadas no equipamento (ex. blindagem, isolamentos), e da intensidade do fenómeno. O cenário mais comum após ultrapassadas as protecções traduz-se em alterações momentâneas nos sinais eléctricos dos circuitos afectados. Nas redes de comunicação isto significa geralmente a destruição da informação codificada nos sinais que suportam a transmissão da informação. A destruição da informação resulta da inversão de bits das tramas, pelo que a caracterização destes eventos para a avaliação proposta na dissertação não requer uma distribuição contínua que permita representar a probabilidade de ser atingido o limiar da intensidade da interferência que gera o evento assim como, da sua ocorrência temporal. Antes é requerida uma distribuição que represente a probabilidade da ocorrência da inversão de bits provocada pela interferência, sendo este um processo para o qual uma distribuição discreta melhor se adapta.

Em síntese, quando as interferências assumem uma intensidade suficiente para ultrapassar as protecções conferidas pelos mecanismos físicos de protecção do canal de comunicação, de que é exemplo a blindagem do cabo de comunicações, essas interferências podem traduzir-se em erros que resultam na destruição da informação contida numa ou mais *slots* de informação T_{bit}. Este é um fenómeno acentuadamente aleatório que pode ser aproximado a um processo de Bernoulli, correspondendo neste caso o número de experiências ao número de T_{bit} que decorrem até se produzir uma falha da blindagem e a consequente destruição da informação.

Tendo por base esta associação, a probabilidade *p* da distribuição geométrica relaciona a taxa de erros que incide nas comunicações (*Bit Error Rate* - BER). Desta forma, uma variável aleatória que represente uma amostra dos erros no barramento apresentará um valor esperado que tende para a média da distribuição.

$$E\left(\frac{1}{BER}\right) \approx \mu = \frac{1}{p} \qquad (5.8)$$

5.3.2 Avaliação Preliminar

A caracterização da operação da FDL em cenários de faltas requer uma identificação prévia dos eventos que levam a FDL para estados que tenham um maior impacto na estabilidade do anel lógico, assim como, na diminuição da resposta temporal da rede.

Uma forma de simplificar o processo de caracterização consiste na execução de um conjunto de experiências nas quais a rede é configurada especificamente para avaliar os efeitos das faltas nos mecanismos implementados na FDL [Carvalho05a].

5.3.2.1 Condições Específicas da Avaliação

Para suportar esta avaliação a infra-estrutura de comunicações foi configurada com um conjunto de nove estações activas. Os seus endereços foram atribuídos de acordo com uma distribuição uniforme, estabelecendo assim GAP's de dimensão variável.

$$Estações = \{9,20,25,32,35,38,51,69,83\} \qquad (5.9)$$

As estações foram configuradas para operar em vazio, pelo que nestas circunstâncias existem unicamente serviços de comunicação na rede com origem na FDL e com finalidade de estabelecer e manter o anel lógico a funcionar. Os parâmetros da FDL permanecem constantes durante a totalidade das experiências (Tabela 5.1).

Parâmetro	Valor
Taxa de transmissão (*bit rate)*	500kbit/s $\Rightarrow$ T_{bit}=2μs
T_{TR} – *Traget Rotation Time*	20ms – 10000 T_{bit}
T_{ID1} – *Idle Time 1*	37 T_{bit}
T_{ID2} – *Idle Time 2*	100 T_{bit}
T_{SL} – *Slot Time*	200 T_{bit}
T_{RDY} – *Ready Time*	11 T_{bit}
HAS – *High Station Address*	126
T_{GUD} – *GAP Update Time*	60000 T_{bit} (G=6)

Tabela 5.1 - Parâmetros da FDL.

Os restantes parâmetros de configuração que são específicos do ASPC2 foram configurados para valores de máxima tolerância às perturbações a que a rede é submetida.

De acordo com a metodologia indicada em §5.3.1, as faltas são modeladas por uma distribuição geométrica. Atendendo a que não é possível estabelecer um cenário suficientemente abrangente das perturbações que afectam a operação da rede, unicamente com base num conjunto de variáveis aleatórias, obtidas a partir de um valor fixo de probabilidade, procedeu-se a uma avaliação parcelar a qual pressupõe a repetição das experiências, para diferentes taxas de erros (*Bit Error Rate* - BER).

De igual modo o fenómeno da interferência apresenta um padrão que não pode ser unicamente descrito pela ocorrência dos erros. A sua duração (*Bit Error Length* - BEL) é igualmente importante. O BEL modela a quantidade de *slots* de informação (T_{bits}) que são destruídos pelos erros. As experiências contemplam três configurações de BEL: 1, 2 e 4 T_{bit} contíguos.

Esta última componente assume maior relevância em função dos dois últimos tipos de erros não serem alvo de uma correcta cobertura pelo mecanismo que suporta a integridade da informação do token. Assim, uma verificação de quais as consequências na operação da FDL que resultam da ocorrência de erros que não são detectados pelo mecanismo de paridade, representa um bom contributo para a caracterização da operação da rede nestas condições.

5.3.2.2 Caracterização de Modos de Operação

A identificação dos modos de operação do PROFIBUS-DP mais relevantes em cenários de faltas foi efectuada através de uma avaliação prévia de cariz qualitativa ao comportamento da rede. Os modos de operação identificados resultaram da aplicação das regras do protocolo à informação recolhida pela infra-estrutura de injecção de faltas. Esta informação suporta a avaliação, respectivamente através do fornecimento de uma imagem do estado do barramento, e do estado da FDL de cada uma das estações.

A imagem do barramento fornece uma indicação global do tráfego na rede, com incidência no tipo de tramas, sua sequência e manifestações dos erros nos sinais do barramento. Mais especificamente, quais as suas consequências no valor dos bytes que compõem as tramas. Esta última informação é mais detalhada na trama de *token,* onde é possível verificar em que byte ocorreu o erro e qual o seu resultado na informação do *token*.

O estado de cada FDL é reportado pelos registos implementados no ASIC ASPC2. Com base nos registos do ASIC é possível identificar para cada estação, o seu estado de operação e erros na operação [Siemens05].

Estados operacionais:

- ***Offline***: estado específico de não operacionalidade (Fig. 5.1);
- ***Listen Token***: estado de monitorização da actividade do barramento (Fig. 5.1);
- ***Hold Token***: engloba os estados operacionais da FDL em que a estação detém o *token*;
- ***Not Hold Token***: engloba estados operacionais nos quais a estação não detém o *token*.

Excepções à normal operação:

- ***Timeout***: Ausência de actividade no barramento por tempo superior ao especificado no temporizador de *timeout*;
- ***LAS Useless***: O limite máximo permitido de *token's* inválidos por cada 256 rotações de *token* foi alcançado;
- ***TS-ADR error***: Estação com endereço já existente no barramento;
- ***Pass-Token error***: Erro grave na transmissão do token;

- ***HSA error***: Detecção de estação com endereço fora do espaço de endereçamento;
- ***Response error***: Detecção de erro na resposta a serviços de comunicação.

Desta avaliação prévia identificaram-se sete excepções à normal operação do protocolo, que se destacam pelo maior significado que podem assumir os seus impactos na operação na rede.

I. Erro Fatal

O PROFIBUS-DP implementa um mecanismo para verificar a integridade dos componentes do canal de comunicação, nomeadamente do seu *transceiver*. O mecanismo é baseado na comparação do *token* transmitido. Neste processo, a estação que detém o *token* deve proceder à monitorização do *token* em simultâneo com a sua transmissão. Caso esta não receba o seu próprio *token*, então a FDL deve assumir um erro grave no seu canal de transmissão ou de recepção [EN96a]. Neste contexto, a FDL deve cessar a sua actividade no barramento e transitar para o estado *offline*.

A norma da rede fornece unicamente uma referência funcional do mecanismo não especificando como deve ser efectuada a comparação. No ASPC2 o mecanismo é implementado da seguinte forma: considera-se a existência de um erro grave no canal de comunicações se na transmissão do *token* a FDL não receber o identificador da trama (SD), ou o receber com erros em duas transmissões consecutivas do *token* [Siemens05].

Este mecanismo está especificado para a detecção de erros permanentes. Contudo, o processo de comparação não o torna imune a reportar falsos erros que derivem de faltas transitórias.

Numa perspectiva operacional, um erro deste tipo para além de provocar a transição da FDL para o estado *Offline*, provoca igualmente a perda do *token*.

II. Erro no Token

Durante a passagem de *token*, se em duas tentativas consecutivas se verificar a existência de erros, desde que ambos não sejam do tipo descrito anteriormente a estação abandona o anel lógico e entra no estado *Listen Token*. O *token* é perdido durante o processo, no entanto, a estação mantém o conteúdo da LAS. Desta forma, após ter sido gerado um novo *token* a estação pode entrar de novo no anel lógico, após este lhe ter sido passado pela estação que a precede, ou ser ela própria a assumir a tarefa de re-inicialização do anel, sem necessitar de estabelecer uma nova LAS.

III. Erro Durante o Slot Time

Na transferência do *token* a FDL da estação ainda detentora do *token* deixa o barramento em estado *idle* durante um tempo especificado pelo parâmetro T_{SL}. Este tempo serve para assegurar que a estação destinatária do *token* o possa receber e iniciar a sua actividade no barramento. Só após monitorizar actividade dessa estação na rede, a estação até então detentora do *token*, assume o sucesso na transferência do *token*.

Se durante o período *idle time* a FDL da estação detentora do *token* monitorizar uma outra qualquer actividade no barramento, assume que uma outra estação detém o *token* e retira-se para o estado *Active Idle*.

A ocorrência de uma falta durante o período *idle time*, produz uma inversão do estado lógico do sinal do barramento que é interpretado como o *start bit* pelas UART's (Fig. 4.19 da §4.42), com a consequente geração de um caracter de UART. Este caracter é interpretado pela estação que monitoriza o barramento como um *byte* de uma trama de outra estação, transitando de imediato para o estado *Active Idle.*

Se a estação destinatária não reconheceu como válido o *token* enviado, o que em cenários de faltas tem grande probabilidade de ocorrer devido à possibilidade do *token* que foi previamente enviado ter sido afectado, então nestas condições, produz-se uma perda de *token*.

IV. Inicialização do Anel

A inicialização do anel é um processo que tem de ser encetado sempre que o anel esteja inactivo e a saída desse estado se faça por uma estação que não tenha a LAS construída.

A indução de eventos de inicialização do anel provocada pela ocorrência de faltas, será sempre muito penalizadora para o desempenho global da rede. De facto, a indução deste evento provoca o colapso do anel lógico, uma vez que a estação que inicia a operação no anel sem LAS construída, enviará um *token* endereçado a ela própria que terá como consequência a auto exclusão do anel de todas as estações que dele faziam parte. Em [Willig01] [Willig99b] [Willig99c] este processo é designado por *Ring Jacking*. A ocorrência deste evento já foi abordada em §2.4.1, tendo sido referido que não obstante o evento ser possível, este não ocorre pela razão descrita nesse trabalho.

Os resultados das experiências de injecção de faltas revelaram que o evento de inicialização com o correspondente colapso do anel lógico ocorre como o culminar de uma cadeia de eventos.

Numa primeira fase, a estação com menor endereço da rede entra no estado não operacional *Offline*. A infra-estrutura retarda o lançamento da estação na rede cerca de 50ms. Esta é uma opção tomada com o objectivo de manter um grau de automatização da execução dos ensaios. Na realidade uma estação que

entre neste estado deverá ser arrancada pelo operador depois de avaliada, as condições de operação.

Se durante o processo de arranque, ou numa fase em que a estação ainda não tenha terminado de construir a sua LAS, ocorrer uma perda de *token*, será esta estação a recuperar da situação, o que implicará uma inicialização do anel lógico. Na prática, este evento será muito pouco provável e só ocorrerá se durante o arranque da estação com o menor endereço na rede, ocorrer uma perda de *token*.

V. Erro no Endereço de Estação

No estado *Active Idle* sempre que uma estação monitoriza no barramento dois *tokens* sucessivos com o campo *Source Address* (SA) igual ao seu endereço, a FDL da estação deve abandonar o anel lógico e transitar para o estado *Listen Token* [EN96a]. Na revisão E do ASPC2, que integra o ASIC DSTni-LX002 usado nas experiências de injecção de faltas, este evento é igualmente observado. Contudo, é detectado logo na sua primeira ocorrência.

Nas experiências que suportaram esta avaliação este evento foi observado no conjunto de estação com endereço:

$$Estações_{TS-ADR\,Error} = \{32, 35, 38, 51, 83\} \qquad (5.10)$$

Estas estações têm como afinidade o facto dos seus endereços serem permutáveis entre algumas estações quando ocorrem erros de comprimento dois - 2 BEL (erros não detectados pelo mecanismo de paridade).

VI. Inconsistência na Lista das Estações Activas

A LAS é um dos elementos de suporte da operação da rede PROFIBUS-DP. De forma a manter a lista actualizada, as estações monitorizam constantemente os *token* que circulam no barramento. Esses *token's* podem ser afectados por erros, alguns dos quais não são detectados pelo mecanismo de protecção da trama, pelo que manter a integridade da LAS é fundamental para permitir a estabilidade do anel lógico.

Possíveis casos que levem a uma inconsistência da LAS e acções para sua resolução não são referidos na norma. Esta tarefa deve ser assumida através de solução do domínio da implementação.

No ASPC2 existe um registo que permite parametrizar um contador do número de *tokens* não válidos por cada 256 rotações do *token*. Quando é alcançado o limite do contador, a FDL transita para o estado *Listen Token* para restabelecer a LAS. Este contador pode servir para detectar e limitar os efeitos que resultem da inconsistência da LAS.

Nas experiências verificaram-se diversos eventos que indiciam a existência de um estado de inconsistência que é assinalado pela excepção *LAS-Useless*.

Os eventos observados que estão na origem do estado descrito pertencem a um conjunto de eventos dos quais alguns se tornam activos, e que têm um padrão

cujo mecanismo pode ser desta forma descrito. Considere-se o conjunto das estações representadas na LAS:

$$\{LAS_i, LAS_{i+1}, LAS_{i+2}, LAS_{i+3}, \cdots, LAS_{i+n}\} \quad (5.11)$$

Durante o processo de transferência do *token* da estação LAS_{i+2} para a estação LAS_{i+3}, ocorre um erro não detectado pelo mecanismo de paridade que transforma a transacção em:

$$LAS_{i+1} \rightarrow LAS_{i+3} \quad (5.12)$$

Este erro só pode ser detectado pela estação LAS_{i+2} que repete o processo de transferência do *token*.

$$LAS_{i+2} \rightarrow LAS_{i+3} \quad (5.13)$$

Após emissão deste *token*, o anel lógico entra parcialmente em colapso, permanecendo unicamente as estações LAS_{i+1} e LAS_{i+2} a operar. As restantes não aceitam o *token* endereçados por estas estações e respondem às tramas *Request FDL Status* com o estado *Master it is in the logical ring*. Este comportamento mantém-se até ser atingindo o limite de *token's* inválidos parametrizado no contador, a partir do qual as estações restabelecem a LAS no estado *Listen Token*.

Este mecanismo é em tudo semelhante ao utilizado para detectar erros do tipo *TS-Address Error*. Contudo, uma análise aos dados recolhidos mostra a existência de uma diferença essencial que condiciona o tipo de evento detectado. Essa diferença está relacionada com o valor que o campo SA assume em resultado do erro não detectado e da sua posição na LAS.

- Quando o erro no campo SA, altera o endereço original para um que é pertença da estação que a antecede na LAS, a dinâmica do processo leva à ocorrência de um estado de inconsistência da LAS das estações que monitorizam o barramento, exceptuando as estações com endereço LAS_{i+1} e LAS_{i+2};
- Quando o erro no campo SA altera o endereço original para um que é pertença de uma estação que não está contígua (na LAS) à estação afectada, a dinâmica do processo produz um erro *TS- Address Error,* na estação que monitoriza o *token* e que possui esse endereço.

VII. Remoção por Salto da Estação

Uma rede de comunicações é um sistema dinâmico, que pode ver a sua estrutura alterada ao longo do tempo, quer através da adição, quer através da remoção de nós de comunicação. No PROFIBUS-DP uma estação é removida do anel quando a sua entrada na LAS é apagada. A saída do anel está geralmente associada a uma alteração funcional da operação da estação, em regra devido ao desligar do equipamento, ou como resultado de falhas do equipamento que não lhe permitam continuar o processo de comunicação. No essencial da operação esta estação deixará aceitar os *tokens* que lhe são endereçados.

Neste contexto, o mecanismo de remoção de estações do anel está baseado na limitação do número de tentativas de passagem do token sem sucesso. A norma da rede especifica o envio de um *token*. No caso de ausência de actividade, no barramento por parte da estação endereçada, o processo é repetido até duas vezes. A partir desse limite a passagem do *token* é tentada para a estação seguinte. As estações que são saltadas por este processo são removidas da LAS pelas estações que monitorizam o barramento.

Este processo pode numa primeira instância indiciar uma possível fonte de remoção de estações do anel, consequência de erros que não são detectados pelo mecanismo de paridade implementado no *token – Remoção por Salto da Estação*. Em [Willig01], este evento é apontado como um dos factores de instabilidade do anel, e pode ser descrito da seguinte forma:

Se uma estação com endereço LAS_{i+1} no processo de passagem do token para a estação LAS_{i+2}, sofrer um erro não detectado no token, e que este novo token endereça uma estação LAS_{i+3}, a estação LAS_{i+2} considera-se ultrapassada e abandona de imediato o anel lógico. Se no processo que medeia a construção de uma nova LAS no estado Listen Token são esgotados o número de tentativas de passagem do token, e este é efectivamente passado para outra estação, então esta é removida das LAS das restantes estações.

Contudo, este modo de operação não é enquadrável com o especificado na norma [EN96a], uma vez que esta define que a passagem de *token* só se torna efectiva após a estação endereçada estabelecer actividade no barramento. Actividade essa que é verificada através da monitorização de um identificador (*SD*) de trama sem erros, pelo que não é possível uma estação ser removida do anel pelo mecanismo descrito.

Assim, a remoção de estações do anel pelo evento *Remoção por Salto da Estação* só ocorre em condições muito específicas, em que o limite de tentativa de passagem do token é alcançado sem que este tenha sido efectivamente passado e quando não exista perda de *token* devido à detecção pela estação emissora de erros em dois *tokens* consecutivos.

Uma dessas situações pode ocorrer para o seguinte cenário:

1. A estação recebe um *token* com um erro não detectado, que o transforma num *token* endereçado por uma estação que não a antecede na LAS;
2. O *token* é repetido, mas desta vez sem erros. Devido à diferença de *tokens* válidos a sua aceitação requer a confirmação através de um segundo *token* igual;
3. O terceiro *token* é afectado por um erro de qualquer tipo.

Uma outra situação em que o limite de passagens de *token* é alcançado, decorre de condicionalismos físicos do processo de interferência e da estrutura igualmente física do canal de comunicação.

No primeiro caso, a intensidade, o local onde ocorre a interferência e os parâmetros físicos dos componentes do canal de comunicações, podem originar erros no *token* cujo efeito é parcialmente sentido na rede, ou seja, podem ser detectados somente por algumas estações. Neste caso a estação a que se destina o *token* pode rejeitar os *tokens* e a estação que os transmite não detectar os erros e atingir o limite de tentativas que a levem a saltar a estação.

Num segundo cenário, a interferência afecta globalmente toda rede. Contudo, a estrutura do *transceiver* faz com que no processo de teste à sua estrutura (*loop-back*), o sinal do andar de saída se imponha localmente à interferência. Nesta situação, o limite de tentativas pode igualmente ser atingido.

Na infra-estrutura de injecção de faltas, a ponta injectora está equipada com componentes activos capazes de imporem um sinal que na grande maioria das situações provoca erros que afectam todas as estações da rede. Contudo, o barramento é um sistema físico com parâmetros distribuídos que inclui resistências, indutâncias e capacidades, susceptíveis de provocar atenuações ou outras alterações do sinal da ponta de injecção, que numa última instância não consiga impor um sinal suficientemente forte para produzir o erro na zona onde é efectuado o *loop-back*.

Embora marginal, quando comparado com a eficiência da ponta de injecção, este efeito teve a virtude de mostrar um comportamento susceptível de ocorrer nas redes reais.

5.3.2.3 Perfis de Perturbação da Rede

Uma análise aos eventos identificados revela que a sua influência na rede não é igual. Contudo, estes revelam padrões que podem ser facilmente enquadrados em duas classes [Carvalho05a]:

- ***Interrupção do Serviço do Sistema* (*System Outage*)**: perturbações que provocam uma interrupção temporária da operação de todas as estações da rede. Este padrão de operação é resultado da perda de *token*, que se traduz na impossibilidade do anel operar até que um novo *token* seja gerado. Esta classe agrupa os eventos: *Erro Fatal*; *Erro no Token* e *Erro Durante o Slot Time*.

  O evento *Inicialização do Anel* não pode ser enquadrado nos de eventos que contribuem para a *Interrupção do Serviço do Sistema*. Este é um evento que decorre de condições muito específicas, e sempre depois de se ter verificado uma perda de *token* em consequência de um dos três eventos referidos.

- ***Interrupção do Serviço da Estação* (*Station Outage*)**: perturbação que se caracteriza igualmente por uma inibição temporária das estações efectuarem os seus serviços de comunicação na rede. Contudo, nesta classe, os efeitos estão confinados a um número restrito de estações. Este padrão de operação resulta da saída do anel lógico das estações que são afectadas pelos eventos: Erro no

Endereço de Estação; *Inconsistência na Lista das Estações Activas* e *Remoção por Salto da Estação*.

5.3.2.4 Mecanismos de Recuperação

Qualquer uma das classes de eventos descritas provoca indisponibilidade temporal das estações para efectuarem os seus serviços de comunicação (*Outage Time*). A perturbação persiste até que as acções encetadas pelo protocolo reponham as condições de operação anteriores. O PROFIBUS-DP recupera desses estados através de dois mecanismos.

I. Timeout

O mecanismo de *timeout* é implementado por um temporizador que monitoriza a actividade das estações no barramento, assim como, o tempo que o barramento está no estado *idle*. O temporizador é iniciado após o arranque da interface de comunicações no estado *Listen Token*, ou após a recepção do último *bit* de uma trama. O mesmo temporizador é parado quando recebe o primeiro *bit* da trama seguinte [EN96a].

Quando o barramento permanece no estado *idle* por um período superior ao valor de *timeout*, o mecanismo assinala o evento e o barramento é considerado inactivo pela estação onde este ocorreu. Como resultado a estação assume o controlo do barramento gerando um novo *token*. O tempo de *timeout* é especificado através de:

$$T_{TO} = 6 \cdot T_{SL} + 2 \cdot n \cdot T_{SL} \qquad (5.14)$$

Em que n representa o endereço da estação e T_{SL}, o *Slot Time*. Recorrendo a esta expressão e considerando que a estação com o menor endereço se encontra num estado operacional é possível determinar o valor mínimo para que a rede recupere de uma perda de *token*.

Contudo, determinados cenários de faltas são susceptíveis de induzir um comportamento não desejável na operação do temporizador. De facto, em ambiente onde as comunicações possam ser perturbadas por interferências, os tempos de recuperação tendem a prolongar-se, e em casos extremos assumir valores muito significativos como demonstram os resultados das experiências efectuadas para a avaliação da operação da rede.

Contribui para este comportamento, o facto das faltas que ocorrem durante o período em que o barramento se encontra no estado *idle* provocarem alterações no sinal que é interpretado nas UART's como sendo um *start bit*, gerando desta forma um caracter na UART. Este comportamento associado a um padrão de repetição causado por múltiplas faltas, provoca um igual número de reinicializações do temporizador, com consequências temporais relevantes ao nível do desempenho da rede.

II. Inserção de Estações

No PROFIBUS-DP as estações que formam o anel são responsáveis por assegurar que estações activas que não são membros do anel possam ser admitidas no mesmo. A verificação da existência de estações prontas para serem admitidas no anel é efectuada periodicamente e com frequência definida pelo *GAP Update Time* T_{GUD}.

Este processo é efectuado em cada estação, através do envio de uma trama *Request FDL Status* para cada um dos espaços de endereçamento da GAP. O serviço de comunicação é efectuado para um único endereço por cada recepção do *token*, e somente, se após efectuados os serviços prioritários a estação possuir ainda *Token Holding Time*. O T_{GUD} é definido através da expressão:

$$T_{GUD} = G \cdot T_{TR}, \qquad 1 \le G \le 100 \qquad (5.15)$$

Desta forma, o tempo para inserir uma estação no anel depende dos seguintes parâmetros:

- **Do comprimento da GAP**;
- **Do T_{RR}**.

A ocorrência de falta durante o processo fará aumentar o tempo de inserção devido a qualquer uma das seguintes razões:

- Repetição de mensagens que provocam um aumento do T_{RR} que no limite pode ter como consequência a não existência de *Token Holding Time*, para efectuar o serviço *Request FDL Status*;
- Saída de estações do anel e consequente aumento do comprimento da GAP.

5.3.2.5 Estimadores para Avaliação dos Modos de Excepção

A avaliação da influência na operação do PROFIBUS-DP causada pelos eventos descritos nas secções anteriores passa numa primeira instância por aferir da susceptibilidade do protocolo a tais eventos. Tendo presente este objectivo foram, definidos estimadores para determinar o valor esperado da sua probabilidade. Os estimadores incluem parte do conjunto de eventos observados e_{obs} e de agrupamentos de eventos que definam não só a susceptibilidade do PROFIBUS-DP aos eventos bases mas também caracterizem a susceptibilidade a padrões de perturbação da rede.

$$\begin{aligned} e_{obs} = \{&Erro\ Fatal, Erro\ no Token, Erro\ Durante\ o\ SlotTime, Inicialização\ do\ Anel, \\ &Erro\ no\ Endereço\ de\ Estação, Inconsistência\ da\ LAS, \text{Re}moção\ por\ Salto\ da\ Estação\} \end{aligned} \qquad (5.16)$$

Neste contexto, foram definidos estimadores base $\hat{p}_i$ relativos ao conjunto de eventos *i*, e estimado o seu valor a partir dos dados amostrados nas experiências de injecção de faltas.

$$\begin{aligned} i = \{&Interrupção\ do\ Serviço\ do\ Sistema, Erro\ Fatal, Inicialização\ do\ Anel, \\ &Erro\ no\ Endereço\ de\ Estação, Inconsistência\ da\ LAS, \text{Re}moção\ por\ Salto\ da\ Estação\} \end{aligned} \qquad (5.17)$$

Assim, considerando *n* o número de experiências independentes, E_{ij} o número de eventos observados do tipo *i* para cada experiência *j* e TK_j o número total de *tokens* transmitidos na rede durante essa mesma experiência. Assumindo o início da experiência em regime estacionário §4.7.2 o estimador para probabilidade do evento *i* - $\hat{p}_i$ é representado pela expressão:

$$\hat{p}_i = \frac{1}{n}\sum_{j=1}^{n}\frac{E_{ij}}{TK_j} \tag{5.18}$$

Os outros estimadores que fazem parte dos eventos observados (e_{obs}) foram obtidos indirectamente sendo o seu valor derivado a partir dos estimadores base. Fazem parte destes, os eventos *Erro no Token* e *Erro Durante o Slot Time*, que podem ser agrupados num único evento[2] e estimar o seu valor a partir da probabilidade dos estimadores *Interrupção do Serviço do Sistema* e *Erro Fatal*.

De forma semelhante, o estimador *Interrupção do Serviço da Estação* que caracteriza a susceptibilidade do PROFIBUS-DP a um padrão de perturbações que afecta uma ou um número limitado de estações pode ser derivado a partir dos estimadores que quantificam os eventos *Erro no Endereço de Estação, Inconsistência na Lista das Estações Activas e Remoção por Salto da Estação*.

A metodologia seguida para a obtenção das amostras foi descrita em §4.7.2, e está de acordo com o método das replicações independentes. Os estimadores foram definidos para um intervalo de confiança de 95% e uma largura do intervalo de confiança inferior a 5% do valor estimado. A precisão do intervalo de confiança foi especificada para o evento mais frequente, e consequentemente aquele que previsivelmente maior relevância apresenta para o desempenho da rede. No caso dos eventos com um valor de probabilidade baixo relativamente ao evento mais frequente, esta especificação foi relaxada podendo a largura do intervalo alcançar cerca de 20% do valor estimado.

5.3.2.6 Análise da Frequência de Modos de Excepções

A susceptibilidade do PROFIBUS-DP a faltas que induzem padrões de operação com relevância no desempenho da rede é caracterizada nas secções que se seguem. A sua apresentação é efectuada com recurso a gráficos nos quais estão representadas tendências do comportamento do protocolo a esses mesmos em cenários.

A caracterização dessa susceptibilidade é efectuada através da representação da probabilidade da ocorrência dos eventos em função do BER e do BEL, que surgem no sistema como manifestações das faltas que estão na base dos dois perfis de perturbação da rede. Perda de *token* à qual está associada a *Interrupção*

[2] Os efeitos destes eventos na rede são iguais, e o evento *Erro no Token* representa um caso particular do *Erro Durante o Slot Time*, em que o erro não ocorre durante o *Slot Time*, mas após, este ter expirado traduzindo-se o seu efeito na corrupção do *token* sem contudo gerar um erro severo – *Erro Fatal*

do Serviço do Sistema, e saída de estações do anel que contribuem para a *Interrupção do Serviço da Estação*.

I. Susceptibilidade à Perda de Token

A perda de *token* leva inevitavelmente a um período durante o qual todo o sistema fica inibido de comunicar. A probabilidade do sistema entrar neste estado está representada na figura 5.6. Contribuem para este estado as faltas que levam à manifestação de qualquer um dos eventos: *Erro Fatal*, *Erro no Token* e *Erro Durante o Slot Time*.

Quando a análise da figura é efectuada na perspectiva da sensibilidade do PROFIBUS-DP ao cumprimento das faltas (BEL), as variações verificadas na probabilidade de ocorrência da *Interrupção do Serviço do Sistema* são marginais. De facto, constata-se um pequeno aumento do valor da probabilidade da interrupção do Serviço do Sistema nas experiências em que a configuração das faltas é 4 BEL e 2 BEL.

A justificação para esta diferença advém da existência de uma fonte de perda de *token* suplementar às que foram descritas em §5.3.2.2. Isto ocorre quando se verifica uma inconsistência na LAS em algumas estações. Neste caso, uma estação que não conheça o seu sucessor entra no estado *Await Status Response* e inicia procedimentos de manutenção da GAP. Se durante o processo receber uma trama que não seja identificada como resposta ao serviço *Request FDL Status*, a FDL da estação entra em *Active Idle* ocorrendo uma perda de *token* [EN96a].

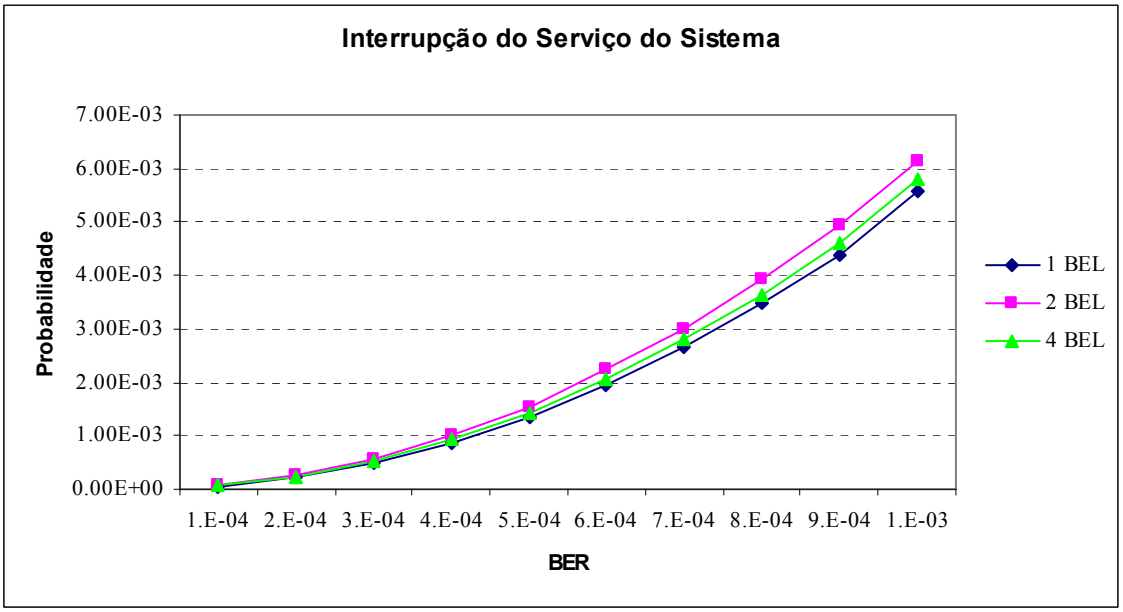

Figura 5.6 - *Interrupção do Serviço do Sistema.*

Não obstante a existência destas pequenas variações, quando avaliada na perspectiva da definição de um padrão de comportamento da perda de *token* relativamente à configuração das faltas, pode-se considerar sem grande perda de rigor que a *Interrupção do Serviço do Sistema* é independente da mesma.

Em contraste a *Interrupção do Serviço do Sistema* exibe uma grande sensibilidade ao BER, tendo um padrão de variação de acordo com uma lei exponencial. Este comportamento está fortemente correlacionado com o valor do parâmetro *Slot Time* (T_{SL}). O valor esperado para a ocorrência de erros é especificado pela expressão (5.6) e estabelecida a sua relação com o BER em (5.8). De acordo com estas, é facilmente constatável, que um aumento do BER se traduz num menor valor esperado para ocorrência de faltas. Assim, para valores de BER pequenos o intervalo entre faltas tende a ser maior que *Slot Time*. O aumento de BER tenderá a produzir erros com um intervalo inferior ou igual à duração do *token*, mais o tempo que o barramento está no estado idle. Nestas condições aumenta a probabilidade de ocorrências dos eventos *Erro Durante o Slot Time*, *Erro no Token* e *Erro Fatal*.

Neste contexto, é espectável que a contribuição para a *Interrupção do Serviço do Sistema* tenha uma maior predominância do evento *Erro Durante o Slot Time*, seguido com um valor substancialmente inferior pelo *Erro no Token*, e por último do *Erro Fatal* representado na figura 5.7.

A independência estatística dos eventos de perda de *token* permite obter o valor da probabilidade agregada dos dois primeiros eventos com base na subtracção da componente do evento *Erro Fatal* na *Interrupção do Serviço do Sistema*.

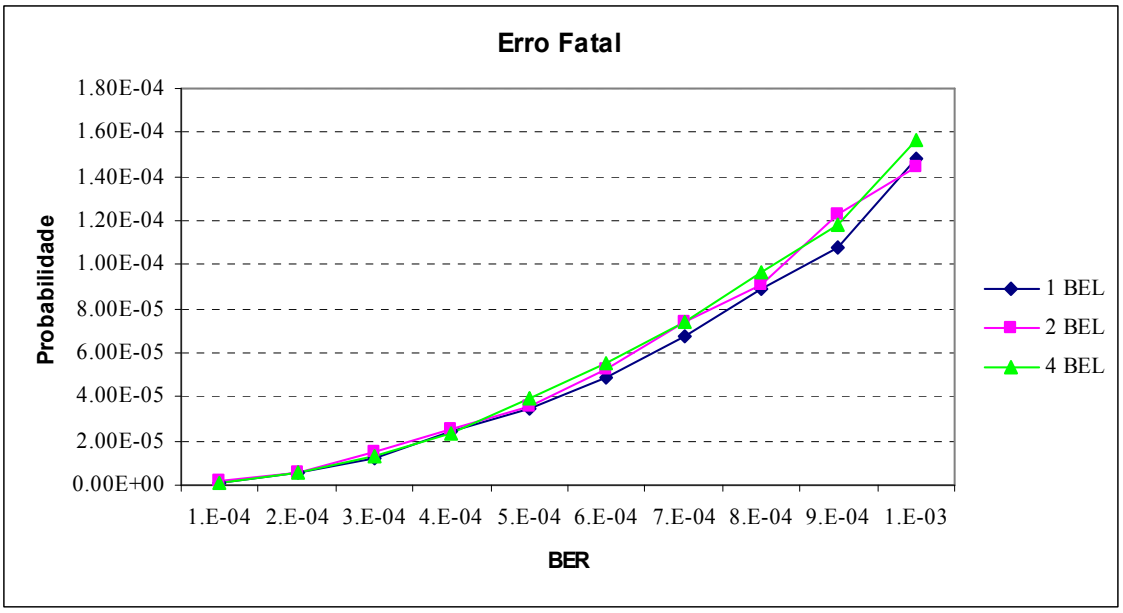

Figura 5.7 - *Erro Fatal.*

Fora do contexto da *Interrupção do Serviço do Sistema* é apresentado o comportamento do evento *Inicialização do Anel*, em função do BEL e do BER (Fig. 5.8). Na figura constata-se a baixa probabilidade deste evento. Acresce que o valor apresentado para BER's mais elevados está sobre avaliado. Isto decorre da reinicialização do mecanismo *timeout* por faltas que produzem com alguma frequência tempos de recuperação superiores a 60ms. Neste caso, o anel é

iniciado pela estação que recupera do estado *Offline* induzindo um evento *Inicialização do Anel*.

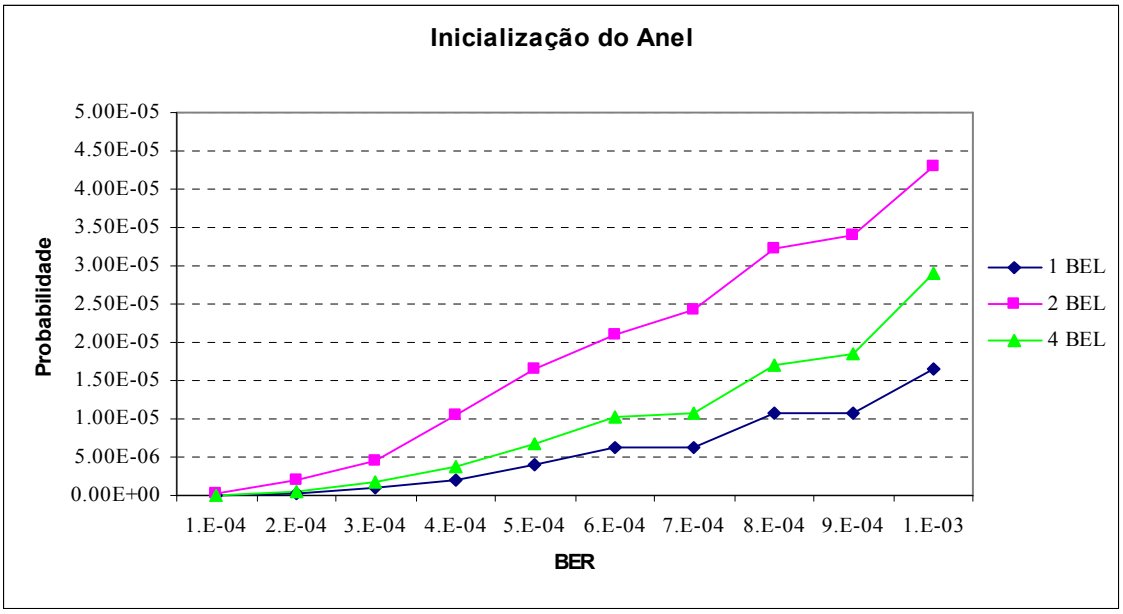

Figura 5.8 - ***Inicialização do Anel.***

II. Susceptibilidade à Saída de Estações do Anel Lógico

A remoção forçada de estações do anel lógico é uma outra manifestação da perturbação da operação da rede PROFIBUS-DP ocasionada por faltas transitórias. A susceptibilidade do PROFIBUS-DP a este tipo de perturbação – *Interrupção do Serviço da Estação* está ilustrada na figura 5.9.

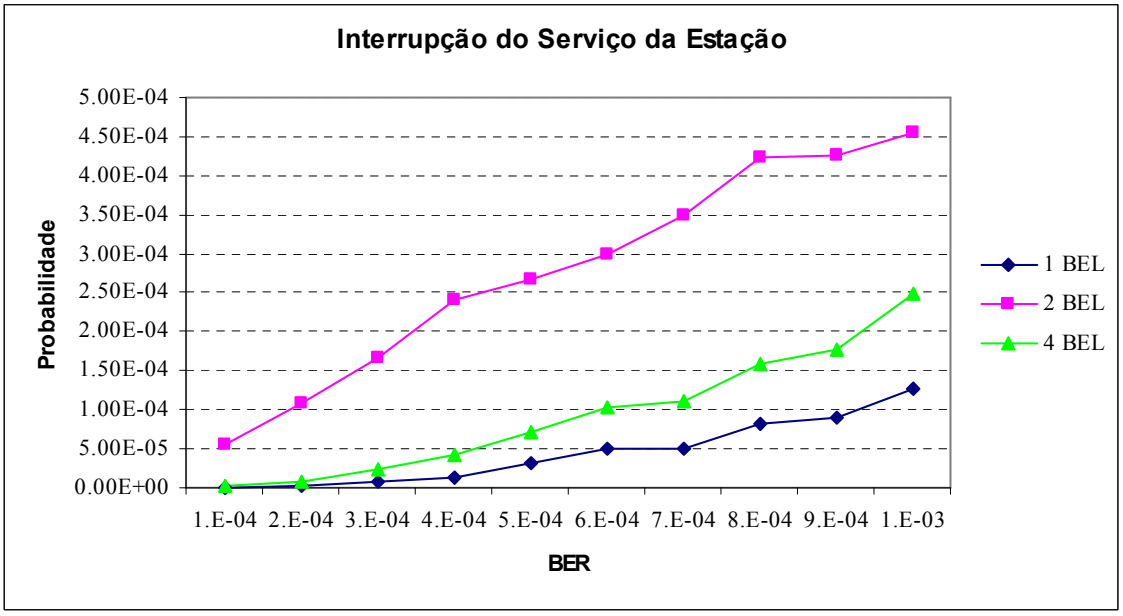

Figura 5.9 - ***Interrupção do Serviço da Estação.***

Na figura é evidenciada uma elevada sensibilidade da *Interrupção do Serviço da Estação* à configuração das faltas, particularmente àquelas que produzem com

mais frequência erros que não são detectados pelo mecanismo de paridade. O caso mais significativo verifica-se para a configuração 2 BEL. Isto resulta das especificidades dos eventos que contribuem para a *Interrupção do Serviço da Estação* que os tornam mais prováveis a erros provocados por essa configuração.

A *Interrupção do Serviço da Estação* é igualmente sensível ao BER, contudo não de forma tão acentuada como a verificada para a *Interrupção do Serviço do Sistema*. Como já foi referido o aumento do BER torna mais provável a ocorrência de erros em *tokens* consecutivos o que também potencia a ocorrência de remoção de estações, nomeadamente pelo evento *Remoção por Salto da Estação* e *Inconsistências na Lista das Estações Activas*. Desta forma, a *Interrupção do Serviço da Estação* depende também de T_{SL}.

As figuras 5.10 a 5.13 representam o padrão de susceptibilidade do PROFIBUS-DP a eventos que contribuem para a *Interrupção do Serviço da Estação*.

O evento *Erro no Endereço de Estação* está representado na figura 5.10, e apresenta a particularidade de só registar manifestações deste erro para configurações de faltas 2 BEL. O facto de não apresentar ocorrências para as outras configurações não significa uma impossibilidade da ocorrência desses eventos mas simplesmente que são muito pouco prováveis. De facto, este evento verifica-se no ASPC2 em condições muito específicas, em que o campo SA de um *token* é alterado para um valor de uma estação que se encontra no anel, mas cuja sua posição não é contígua na LAS, e sem que o erro seja detectado pelo mecanismo de paridade. Na norma, este é ainda mais improvável em função da necessidade do erro ter de ocorrer duas vezes consecutivas, para que se produza um evento do tipo *Erro no Endereço de Estação*.

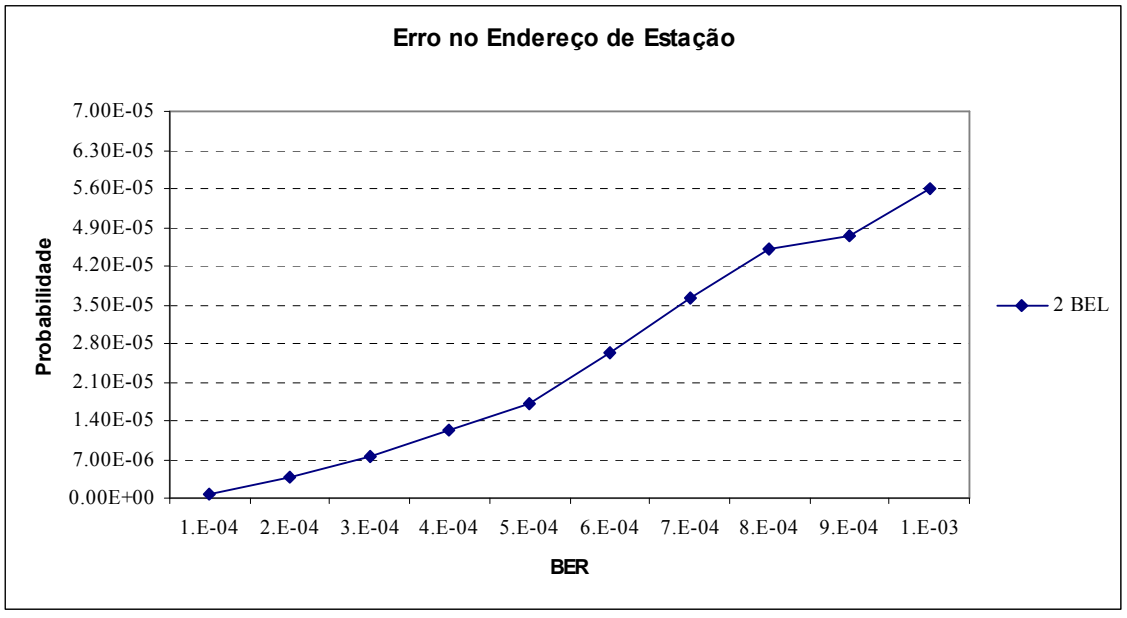

Figura 5.10 - Erro no Endereço de Estação.

A figura 5.11 representa a probabilidade da ocorrência de inconsistências na LAS de estações que se encontram no anel.

Na figura verifica-se que o valor da probabilidade obtido para o cenário 2 BEL se destaca pela sua importância em relação ao cenário 4 BEL e 1 BEL. Este é um evento que partilha um certo grau de semelhança com o evento *Erro no Endereço de Estação*, sobretudo na forma como é detectado (§5.322 – III). Da mesma forma do verificado no caso anterior, as características específicas que estão na origem deste evento são igualmente mais facilmente reunidas quando se geram erros do tipo 2 BEL.

Esta é de resto uma característica comum aos eventos que contribuem para a *Interrupção do Serviço da Estação* e que está na origem do desvio da curva que representa o cenário 2 BEL relativamente aos demais dois cenários da avaliação da *Interrupção do Serviço da Estação.*

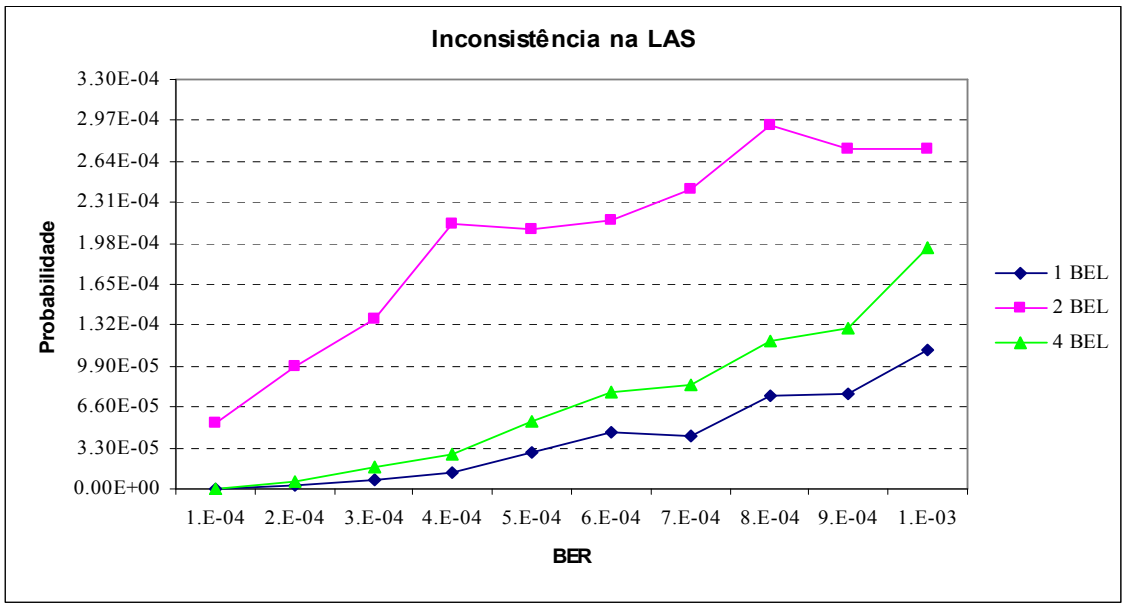

Figura 5.11 - Inconsistência na Lista das Estações Activas.

A probabilidade apresentada na figura 5.11 foi calculada tendo por base todos os eventos de inconsistência da LAS registado nas experiências. Uma característica deste evento está associada ao facto de ocorrer simultaneamente num grupo de estação. Assim o seu valor representa para as condições de ensaio a probabilidade de uma estação ser afectada por uma inconsistência da LAS. Uma outra representação desta probabilidade pode ser obtida formulando-a para o evento agregado, ou seja define a probabilidade do macro evento que conduz a uma inconsistência da LAS na rede num dado instante. A probabilidade do evento agregado da inconsistência da LAS está representada na figura 5.12.

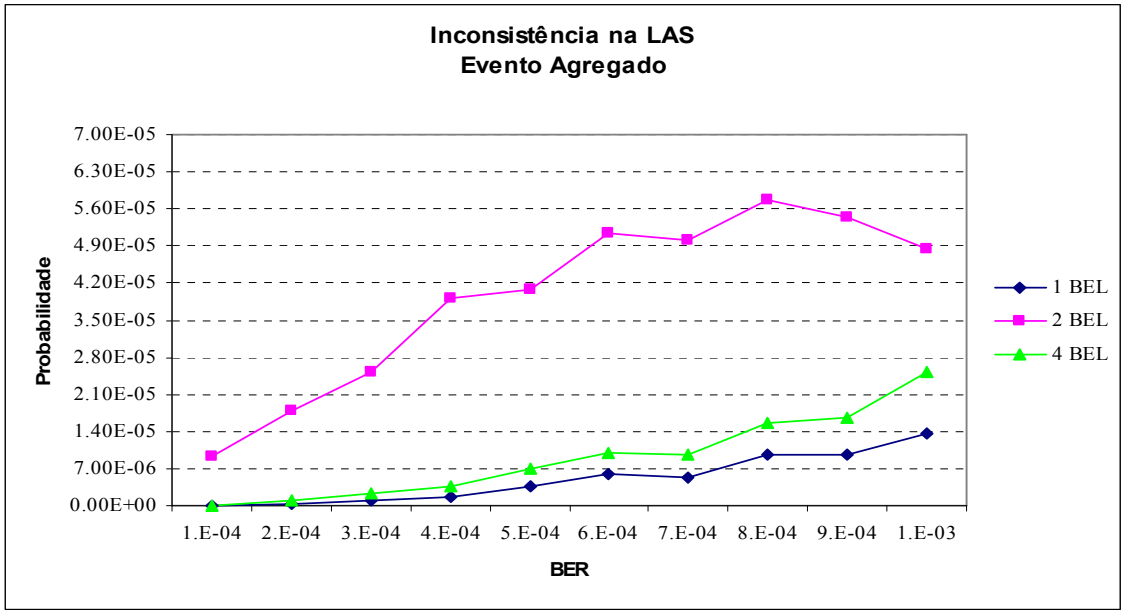

Figura 5.12 - Inconsistência da LAS agregada.

Na figura observa-se uma considerável redução da probabilidade da inconsistência da LAS relativamente à apresentada na figura 5.11. Esta diferença é reveladora do número considerável de estações que são afectadas pelo evento que gera a inconsistência da LAS.

O *Remoção por Salto da Estação* é um evento no qual os erros não detectados pelo mecanismo de paridade assumem igualmente uma importância considerável. Mais uma vez a curva que descreve a probabilidade para a situação 2 BEL destaca-se das de mais. Assim, como neste caso a curva 4 BEL destaca-se da 1 BEL (Fig 5.13). A justificação para este comportamento está igualmente relacionada com a maior facilidade de gerar endereços válidos não detectados pelo mecanismo de paridade que a configuração 1 BEL.

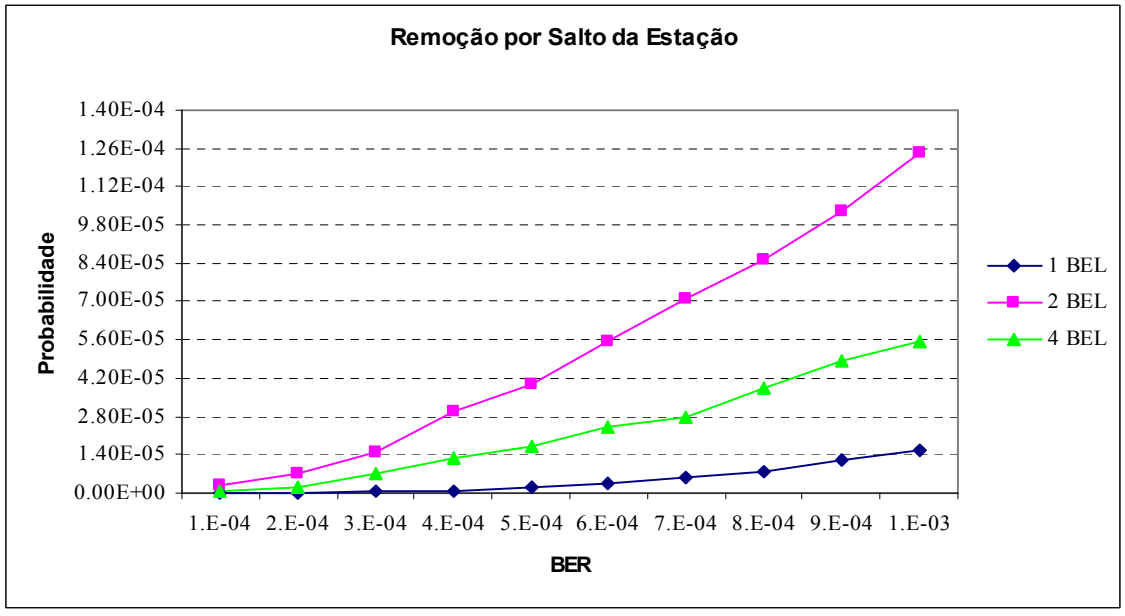

Figura 5.13 - Remoção por Salto da Estação.

Quando uma estação recebe um token de uma estação que não está registada como a estação que a antecede na LAS, o token só é aceite quando a estação receber dois token iguais consecutivos [EN96a]. Após ter recebido um endereço válido cujo erro não é detectado quer pelo mecanismo de paridade, quer pelo mecanismo de *loop back* pelas razões descritas em §5.322 - IV, a estação que detém o token tem de enviar igualmente dois *tokens* válidos para que a estação o aceite. Neste enquadramento a aumenta a probabilidade de ocorrerem erros que façam a estação ser saltada por ter sido ultrapassado número de tentativas para a passagem do *token*.

III. Comparação da Importância dos Contributos

A importância que se atribui a um evento depende em parte das premissas consideradas relevantes para a avaliação. Neste contexto, um evento que afecta a operação de um sistema pode ser considerado severo pelos efeitos, que de forma isolada produz nesse mesmo sistema, ou pela frequência com que este tende a manifestar-se.

De acordo com esta perspectiva a importância de um evento pode ser relativizada. Assim, um evento severo em termo de consequências para o sistema, pode não representar um problema grave se a sua probabilidade não for significativa. Da mesma forma, um evento que não seja considerado demasiadamente severo, pode revelar-se numa fonte significativa de degradação do desempenho do sistema, se a sua frequência assumir valores não desprezáveis.

O mesmo ocorre relativamente às classes *Interrupção do Serviço do Sistema* e *Interrupção do Serviço da Estação*. Estas têm uma importância relativa no desempenho no sistema de comunicações, assim como os seus eventos têm contributos diferentes para a degradação de desempenho do mesmo.

Quando no contexto da avaliação se considera como elemento mais importante o grau de perturbação provocado pelos eventos no sistema, inequivocamente a *Interrupção do Serviço do Sistema* é o principal candidato, pois representa um estado em que todo o sistema é incapaz de comunicar. Em contraste, a *Interrupção do Serviço da Estação* exibe sempre um determinado grau de confinação da perturbação, ou seja, os seus efeitos não são observados em todas as estações.

Não obstante, esta avaliação fornecer uma informação que permite valorizar o evento em si, não permite avaliá-lo numa perspectiva mais global, enquadrado com os demais eventos de acordo com as suas frequências relativas. Neste sentido procedeu-se a uma comparação dos contributos das duas classes de eventos em função da totalidade das ocorrências.

Na figura 5.14 é fornecida uma comparação do peso relativo da ocorrência dos eventos que originam *Interrupções do Serviço do Sistema* e *Interrupções do Serviço da Estação*. Os resultados mostram que da perspectiva da probabilidade dos eventos existe uma bipolarização nos cenários 1 BEL e 4 BEL. A *Interrupção do Serviço do Sistema* apresenta resultados que variam entre os

94,5% e os 99,6% da totalidade das interrupções de serviço observadas. Em contraste, a *Interrupção do Serviço da Estação* exibe um valor residual que varia entre os 0,4% a 5.5% das ocorrências.

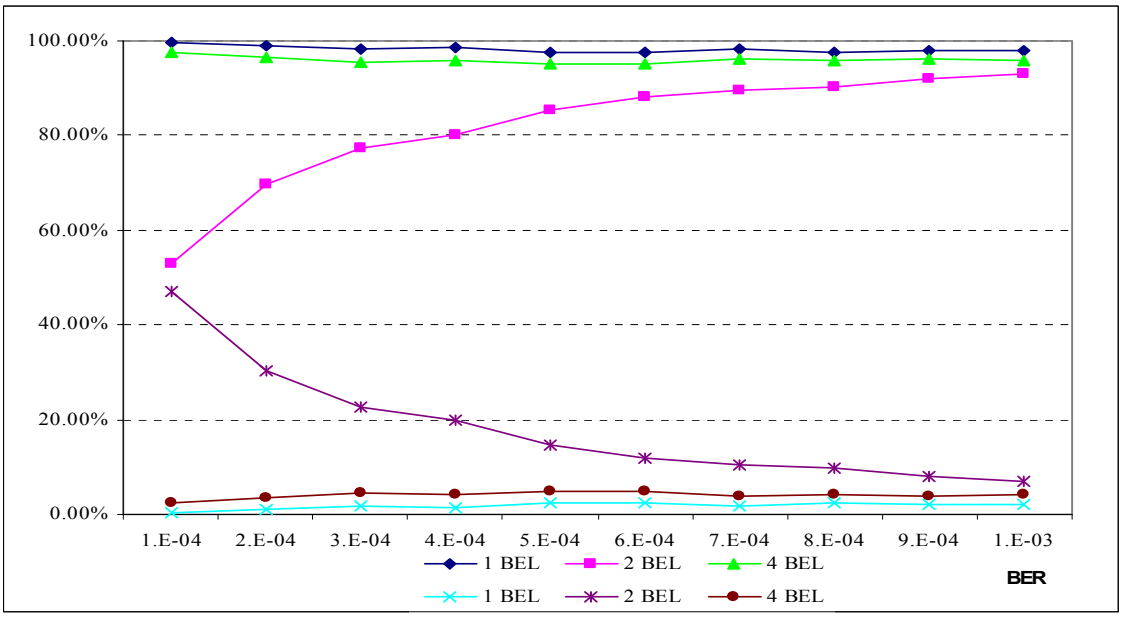

Figura 5.14 - ***Interrupção do Serviço do Sistema vs Interrupção do Serviço da Estação.***

No cenário 2 BEL é observado um número significativo de eventos que originam *Interrupções do Serviço da Estação*. Isto ocorre contudo para valores baixos de BER, o aumento de taxa de erros faz convergir para o padrão de comportamento apresentado nos cenários 1 e 4 BEL. Esta maior susceptibilidade do sistema para exibir saídas de estações do anel está associada ao maior espaçamento entre faltas verificado para baixos valores de BER (superiores a T_{SL}) o que corresponde a uma diminuição da importância das perdas de token devido ao *Erro Durante o Slot Time*.

Numa operação real os erros tendem a não apresentar de forma repetida este padrão que foi assumido na configuração da duração dos erros das experiências. Pelo contrário, é espectável uma variação da duração das faltas, e em conformidade com este cenário o comportamento do sistema na presença de faltas é inerentemente dominado pelo evento *Interrupção do Serviço do Sistema*. Este cenário é ainda mais reforçado pelo facto do valor da *Interrupção do Serviço da Estação* ser calculado com base na probabilidade do *Inconsistência na LAS* apresentado na figura 5.11, o que reforça de forma considerável o valor obtido.

Assume assim relevância para a caracterização da operação da rede em cenários de faltas a identificação dos contributos dos eventos para a *Interrupção do Serviço do Sistema*. Na figura 5.15 fica evidente que o conjunto dos eventos *Erro Durante o Slot Time* e *Erro no Token* representam as principais causas de perturbação da rede, com um valor relativo que varia entre 97.2% a 98.0% das perdas de *token*. As perdas de *token* resultantes de erros severos no *token – Erro Fatal* representam somente cerca de 2.0% a 2.8% dos casos.

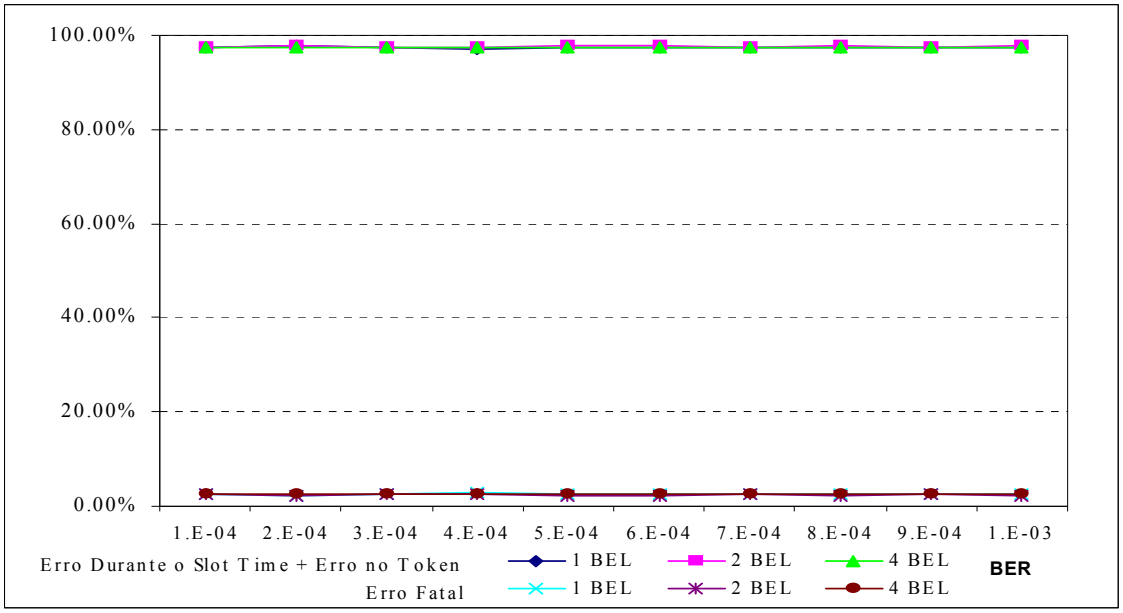

Figura 5.15 - Importância dos contributos para o *Interrupção do Serviço do Sistema*.

Em síntese, tendo em consideração que, o mecanismo de perda de *token* que está na origem dos eventos *Erro Durante o Slot Time* e *Erro no Token* é semelhante, diferindo basicamente no tempo da segunda ocorrência do erro que produz a perda do token, é possível afirmar que o evento *Erro Durante o Slot Time* é consideravelmente mais provável que o *Erro no Token*.

Com base nesta consideração e suportado nos resultados experimentais é possível concluir que o *Interrupção do Serviço do Sistema* representa a perturbação mais provável e mais importante num contexto de faltas, em que o *Erro Durante o Slot Time* é o evento dominante.

5.3.2.7 Resposta Temporal

O comportamento de excepção induzido pelas faltas tem repercussões no desempenho temporal da rede. De forma a analisar o efeito das perturbações que foram identificadas e caracterizadas nas secções anteriores é avaliada a componente temporal da *Interrupção do Serviço do Sistema e da Estação*, assim como o seu impacto na rede. Essa avaliação é efectuada com base em três medidas [Carvalho05b]:

- ***Tempo de Interrupção do Serviço do Sistema* (*System Outage Time*)**: Representa o tempo esperado para uma interrupção de serviço da rede devido a uma perda de *token*. Este tempo é obtido através da contagem do tempo que medeia a ocorrência de dois eventos: ***(i)*** Último byte de uma trama transmitida no barramento. ***(ii)*** Disparo do temporizador de timeout.
- ***Tempo de Interrupção do Serviço da Estação* (*Station Outage Time*)**: Representa o tempo esperado para uma estação que foi removida do anel voltar a ser readmitida pelo mecanismo de inserção

de estações na rede. Este tempo é calculado pela diferença entre o tempo de remoção e o tempo de inserção da estação no anel.

- ***Tempo de Ciclo* (*Bus Cycle Time*)**: Representa o tempo esperado para que o sistema complete uma rotação do *token*. Este tempo é medido através da recepção de dois *tokens* numa mesma estação. A contagem do tempo é iniciada após a recepção de um token e é concluída com a recepção do *token* seguinte.

Estas medidas foram obtidas para as condições experimentais descritas em §5.3.2.1 tendo sido definidos estimadores para o valor esperado (da média) dos eventos com o mesmo nome.

$$k = \{Tempo\ de\ Interrupção\ do\ Serviço\ do\ Sistema, \\ Tempo\ de\ Interrupção\ do\ Serviço\ da\ Estação, Tempo\ \ de\ Ciclo\} \tag{5.19}$$

Considere-se m o número de experiências estatisticamente independentes, $T_{k,j}$ o tempo obtido para os eventos do tipo k durante a experiência j e, $N_{k,j}$ o número total de eventos k observados na experiência j. Assumindo que no início de cada experiência o anel se encontra em estado estacionário, o estimador para o evento do tipo k é definido por:

$$\hat{\mu}_k = \frac{1}{m}\sum_{j=1}^{m}\frac{T_{k,j}}{N_{k,j}} \tag{5.20}$$

Os dados amostrais foram obtidos de acordo com o método das replicações independentes §4.7.2. Os estimadores foram definidos para um intervalo de confiança de 95% e uma largura do intervalo de confiança inferior a 5% (*Tempo de Interrupção do serviço do Sistema*) do valor estimado para o evento mais frequente (*Interrupção do Serviço do Sistema*). Os restantes estimadores *Tempo de Interrupção do Serviço da Estação* e *Tempo de Ciclo* foram obtidos com uma largura do intervalo de confiança respectivamente de cerca de 20% e de 2%.

I. Tempo de Interrupção do Serviço do Sistema

O comportamento do *Tempo de Interrupção do Serviço do Sistema* em função do BER e do BEL é apresentado na figura 5.16.

Os resultados mostram que o *Tempo de Interrupção do Serviço do Sistema* não é afectado pelas variações na configuração do parâmetro BEL utilizadas nas experiências. Isto decorre do facto da perda de *token* ser independente da configuração do erro. Não obstante a independência ao padrão de erro o *Tempo de Interrupção do Serviço do Sistema* é extremamente sensível à taxa de erros BER.

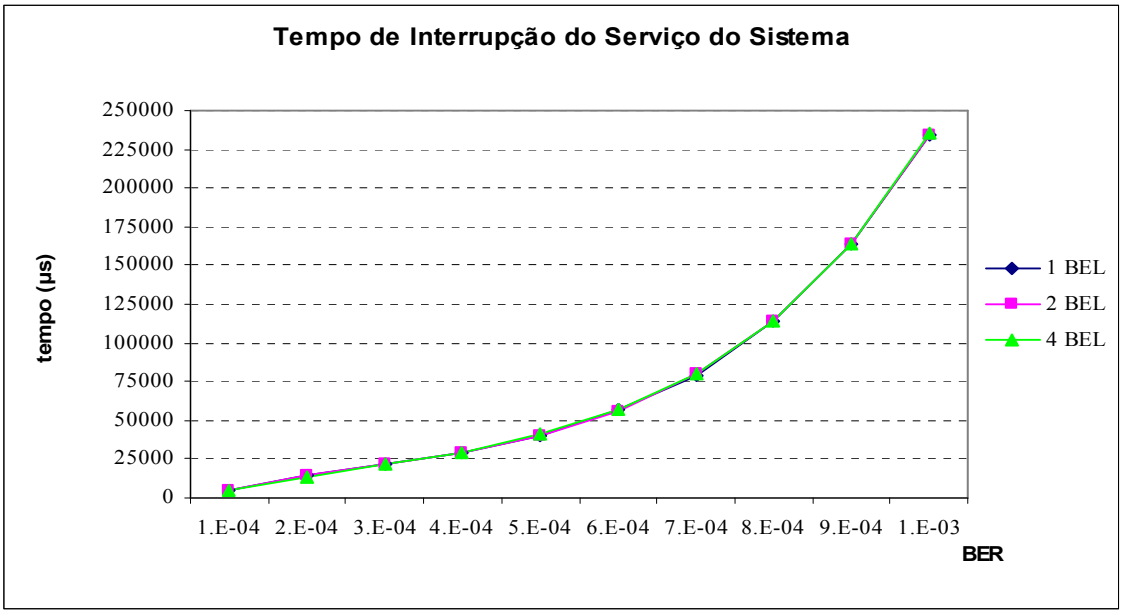

Figura 5.16 - Tempo de Interrupção do Serviço do Sistema.

Nas experiências com taxa de erro 10^{-3} BER, foi observado que o valor esperado do *Tempo de Interrupção do Serviço do Sistema* é superior a 25 vezes o seu valor teórico obtido pela equação (5.14) T_{TO}=9600µs.

Esta diferença está intrinsecamente relacionada com o mau funcionamento do mecanismo de *timeout* em cenário de grandes perturbações dos sinais do barramento. Para valores mais elevados de BER as réplicas dos erros tendem a ter um espaçamento temporal inferior aos 9600µs do limite de *timeout* da estação com menor endereço. Neste cenário de operação, o temporizador de *timeout* é reiniciado pelos erros do barramento e este padrão de operação tende a prolongar o processo de recuperação da perda de *token*, o que se traduz num aumento significativo do *Tempo de Interrupção do Serviço do Sistema*.

A distribuição relativa dos tempos do *Tempo de Interrupção do Serviço do Sistema* em função do BER é apresentada na figura 5.17.

Na figura pode ser observado para baixos valores de BER 10^{-4} que 90% das *Interrupções do Serviço do Sistema* são recuperadas nos primeiros tempos. Com aumento do BER este valor decai significativamente e para 10^{-3} BER só 10% dos eventos são recuperados nos primeiros tempos.

Para as condições de ensaio, o pior caso de recuperação (*Worst Case Recovery Time* - WCRcT), foi identificado nas experiências com taxa de erro 10^{-3} BER e quantificado em 2.09 s.

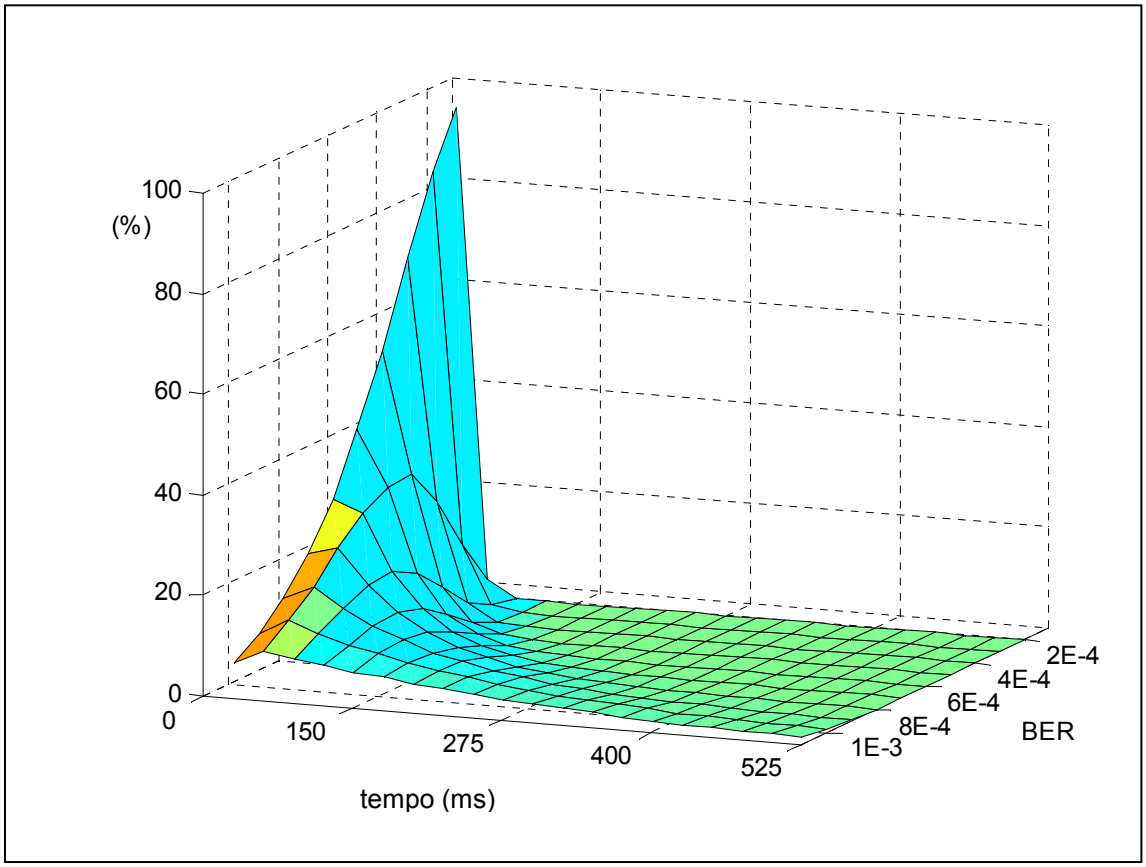

Figura 5.17 - Frequência relativa do Tempo de Interrupção do Serviço do Sistema.

II. Tempo de Interrupção do Serviço da Estação

O *Tempo de Interrupção do Serviço da Estação* é o resultado dos tempos que derivam de três eventos de remoção de estações do anel: ***(i)*** *Erro no Endereço de Estação*; ***(ii)*** *Remoção por Salto de Estação*; ***(iii)*** *Inconsistência na Lista das Estações Activas*. A detecção dos dois primeiros eventos é caracterizada por uma baixa latência desde a ocorrência do evento no barramento de comunicações e a sua sinalização e desencadeamento de medidas de recuperação pela FDL. Desta forma os seus contributos são reflectidos na íntegra no *Tempo de Interrupção do Serviço da Estação*.

A componente temporal da *Inconsistência na Lista das Estações Activas* é constituída por duas fases: ***(i)*** entrada da LAS num estado inconsistente; ***(ii)*** confirmação do estado de inconsistência e desencadeamento de medidas de recuperação.

A entrada da LAS num estado de inconsistência é provocada por qualquer evento que modifique o seu estado para um que não seja representativo da operação da rede. Em §5.3.2.2 – VI, foi descrito o padrão de um evento que induz este efeito na LAS. Igualmente de acordo com a descrição §5.3.2.2 – VI o ASPC2 só confirma este estado de operação após a detenção de um determinado número de *tokens* que não estão de acordo com a sua LAS. Este valor é parametrizável.

A segunda fase inclui o evento que assinala a inconsistência da LAS e todo o processo de recuperação até que a estação seja integrada de novo no anel com a LAS sincronizada com as restantes estações. Em termos de contributo para o *Tempo de Interrupção do Serviço da Estação*, a componente parametrizável não é contada sendo unicamente contabilizada a componente temporal da fase dois.

Neste enquadramento o comportamento do *Tempo de Interrupção do Serviço da Estação* em função do BEL e do BER é apresentado na figura 5.18.

A comparação entre os cenários relativos às diferentes configurações de erros torna evidente a sensibilidade do *Tempo de Interrupção do Serviço da Estação* a este parâmetro. Na figura destaca-se o comportamento para o cenário 2 BEL quando comparado com os cenários 4 e 1 BEL. Este comportamento é justificado pela elevada sensibilidade do evento *Interrupção do Serviço da Estação* a erros não detectados pelo mecanismo de paridade nomeadamente do *Erro no Endereço de Estação* e do *Inconsistência na Lista de Estações Activas*.

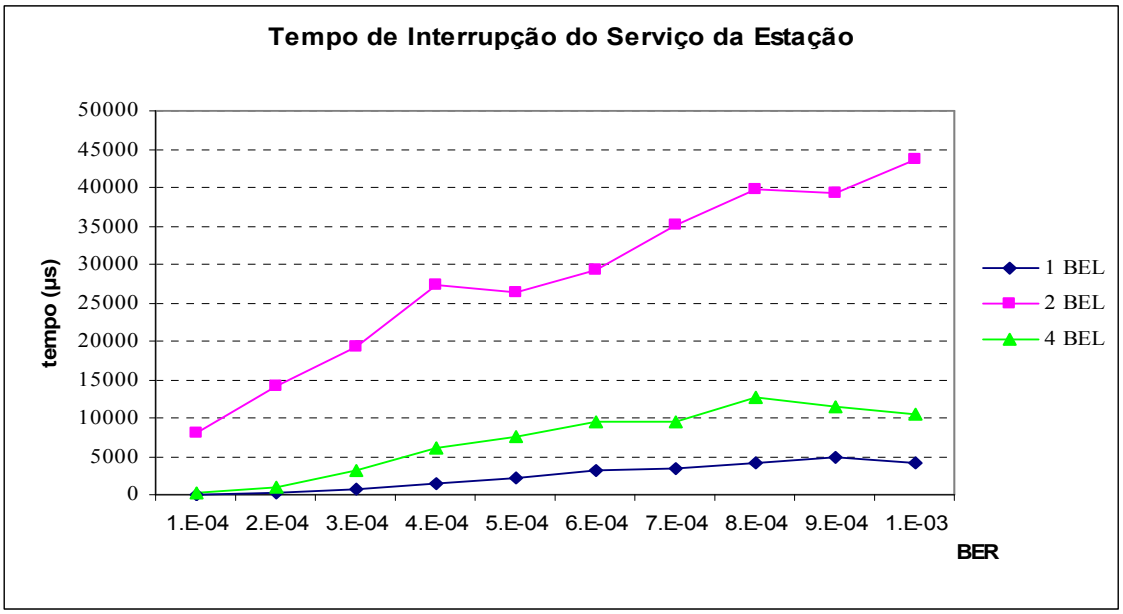

Figura 5.18 - Tempo de Interrupção do Serviço da Estação.

Nas figuras 5.19 a 5.21 são apresentadas as frequências relativas do *Tempo de Interrupção do Serviço da Estação* em função do BER, respectivamente para as configurações de erro 1, 2 e 4 BEL. Os resultados mostram que a maior parte dos eventos *Interrupção do Serviço da Estação* são recuperados para valores temporais relativamente baixos, e que o aumento do BER tem influência na dispersão dos tempos de recuperação[3].

Os resultados revelam igualmente a importância da dimensão da GAP List na componente temporal associada ao *Tempo de Interrupção do Serviço da Estação*.

[3] Quando ocorre uma perda de token durante uma recuperação de saída de estação do anel, o período de tempo em que o anel está inibido de comunicar (*Interrupção do Serviço do Sistema*) não é contabilizado para o *Tempo de Interrupção do Serviço da Estação*.

Isto pode ser constatado pelas três réplicas que sobressaem na figura 5.20 em relação aos demais cenários apresentados nas figuras 5.19 e 5.21. Essas réplicas ficam a dever-se ao evento *Erro no Endereço de Estação* que ocorre somente no cenário apresentado na figura 5.20.

Quando uma estação sai do anel, a GAP aumenta e consequentemente existe uma tendência para estação ser inserida no anel de forma mais demorada. Se a este factor for associado a probabilidade de em cenários de faltas a trama *Request FDL Status* que é usada para a inserir ser corrompida, produzirá um acréscimo significativo do *outage time* em virtude de ter de aguardar por um novo ciclo do T_{GUD} (*Gap Update Time*). Tendo em consideração os endereços das estações que são afectadas nas experiências pelo evento *Erro no Endereço de Estação*, é facilmente constatável que a remoção da estação 83 ou 51 associada a um qualquer erro na trama *Request FDL Status* que inquire o seu estado, resulta em tempos necessariamente longos. Neste contexto, a dimensão da *GAP List* em cenários de faltas pode ser um factor bastante penalizador da disponibilidade das estações da rede.

Igualmente se verifica que essas réplicas só são notadas para baixos valores de BER. O desaparecimento das réplicas para BER's mais elevados (Fig. 5.20) resulta do facto do aumento da taxa de erros perturbar de forma mais significativa os mecanismo de inserção da estação e desta forma produzir uma maior dispersão dos tempos de inserção. Facto, que não ocorre para baixos valores de BER onde essa perturbação é menos provável e em consequência susceptível de produzir um padrão de concentração dos tempos de recuperação destes eventos.

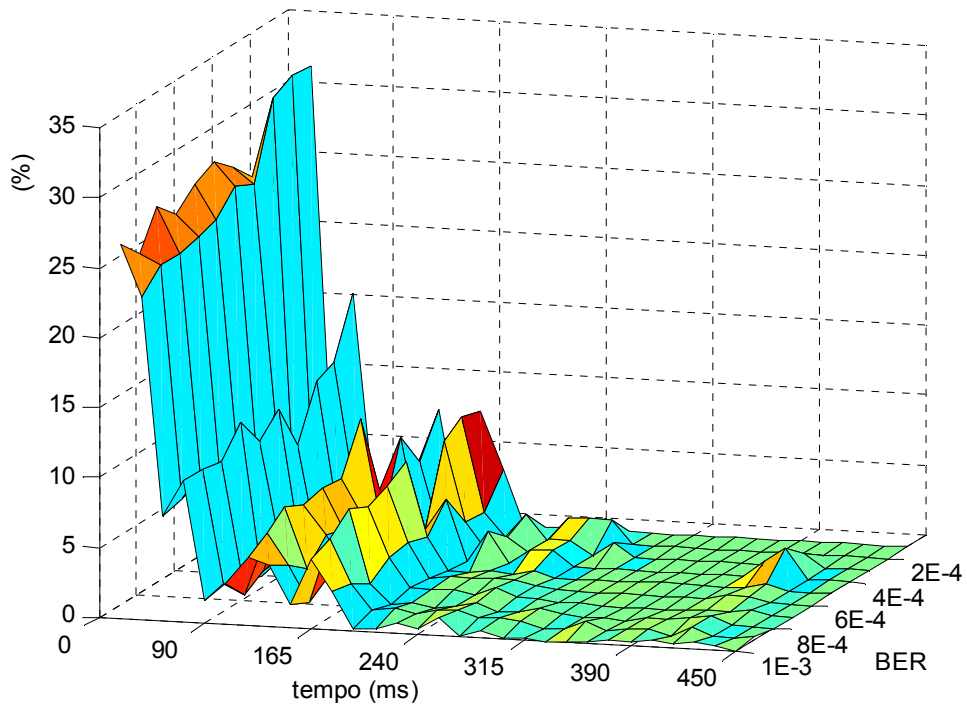

Figura 5.19 - Frequência relativa do Tempo de Interrupção do Serviço do Sistema para a configuração de erro BEL 1.

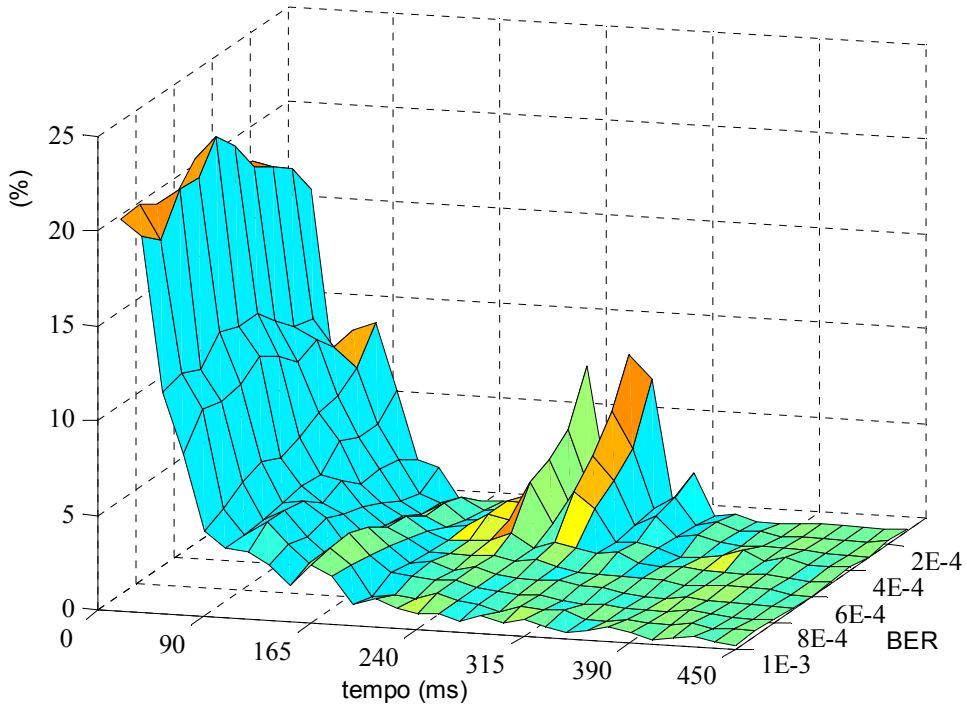

Figura 5.20 - Frequência relativa do Tempo de Interrupção do Serviço do Sistema para a configuração de erro BEL 2.

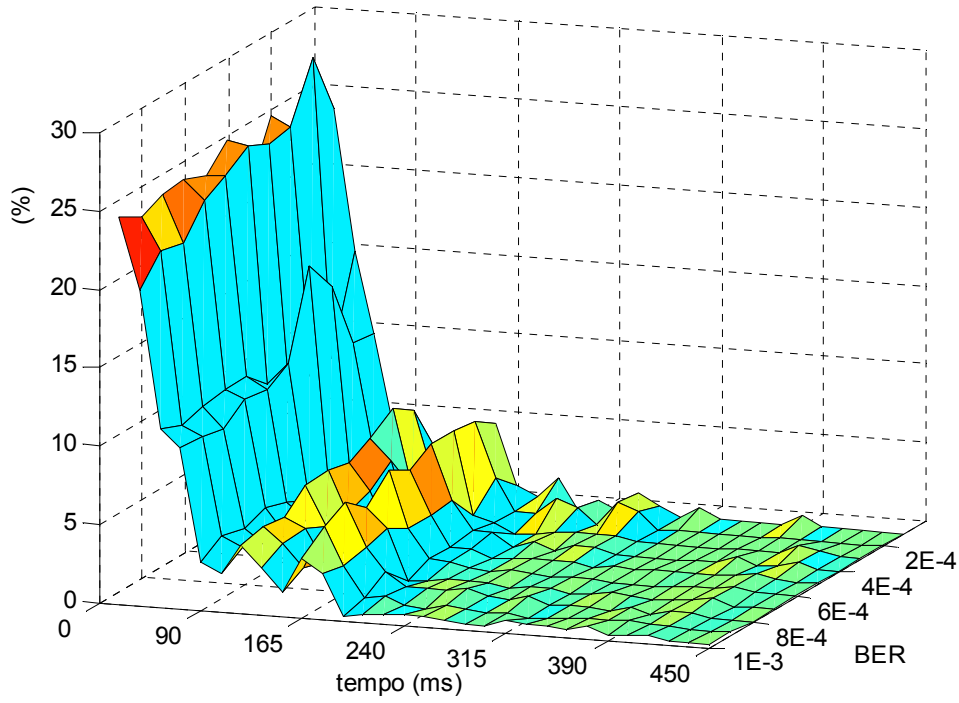

Figura 5.21 - Frequência relativa do Tempo de Interrupção do Serviço do Sistema para a configuração de erro BEL 4.

III. Tempo de Ciclo

A comunicação de dados no PROFIBUS-DP é regulada com recurso a dois importantes parâmetros. Um estático designado por *Target Rotation Time* T_{TR}, que é especificado aquando da configuração da rede. Este parâmetro é projectado para um tráfego tipo na rede devendo prever uma margem que garanta a execução de tráfego de alta e baixa prioridade assim, como, potenciais repetições.

Um parâmetro dinâmico designado por *Real Rotation Time* T_{RR}, calculado por cada estação em cada ciclo que o *token* completa ao anel.

A diferença entre estes dois parâmetros dá origem ao *Token Holding Time* T_{TH}, que representa o tempo que a estação pode usar o *token*.

Neste contexto, o T_{TR} deve absorver as flutuações do T_{RR}. Num cenário de faltas é previsível um aumento do T_{RR}. Este aumento mesmo que pontual poderá implicar uma diminuição na disponibilidade para as estações efectuarem os seus serviços de comunicações na rede.

O *Tempo de Ciclo* da forma como é definido na dissertação, representa uma réplica do T_{RR}. A forma como é estimado não fornece indicações pontuais das perturbações do T_{RR}, mas antes uma indicação pesada dos contributos do *Tempo de Interrupção do Serviço do Sistema* e *Tempo de Interrupção do Serviço da Estação* para o valor esperado (média) do tempo de ciclo do *token*.

O comportamento do *Tempo de Ciclo* em função do BEL e do BER é apresentado na figura 5.22. Os resultados mostram que a configuração do erro não tem influência de registo no *Tempo de Ciclo*. Contudo a taxa de erros afecta de forma significativa este parâmetro. Para valores abaixo de $4 \cdot 10^{-4}$ BER o valor médio do *Tempo de Ciclo* está estabilizado nos 4ms. Para taxas de erros superiores existe uma inflexão deste estado verificando-se um crescimento exponencial que se fixa na ordem dos 17ms para 10^{-3} BER.

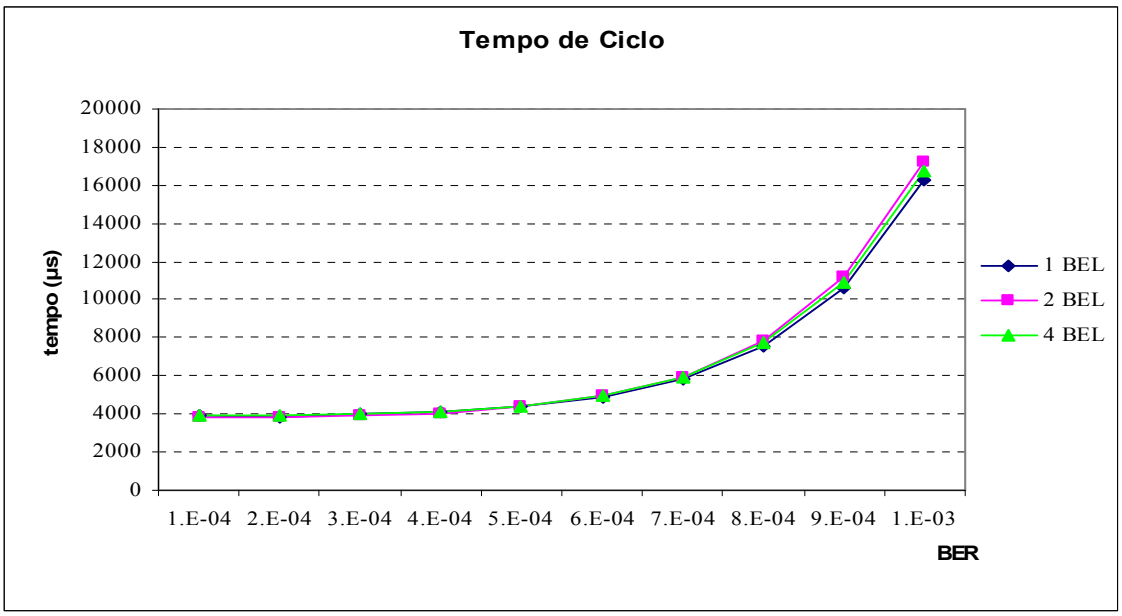

Figura 5.22 - *Tempo de Ciclo*.

O valor médio teórico do T_{RR} em situação normal ou seja não afectado por erros estará contido no intervalo (5.21)

$$\left[n \cdot T_{TC} \,,\, n \cdot \left(T_{TC} + T_{M_{\mathrm{Re}\, qFDL\, C}} + T_{ID} \right) \right] = \left[1269 \,,\, 3123 \right] \mu s \qquad (5.21)$$

Em que no limite inferior T_{TC} do tempo de ciclo de *token* (*Token Cycle Time*) representa o tempo de passagem do *token* isenta de erros e *n* o número de estações do anel.

$$T_{TC} = T_{TF} + T_{TD} + T_{ID} \qquad (5.22)$$

São componentes do T_{TC} o tempo de transmissão do *token* T_{TF}, o tempo de atraso devido à propagação do sinal no barramento T_{TD} e do tempo que a estação deve aguardar até poder estabelecer actividade no barramento T_{ID}.

No limite superior assume-se que existe sempre um serviço *Request FDL Status* efectuado em cada estação, sem contudo existir resposta e com tempo de ciclo do serviço $T_{M_{\mathrm{Re}\, q\, FDL} C}$.

$$T_{M_{\mathrm{Re}\, q\, FDL} C} = T_{FDL} + T_{ID} + T_{TD} \qquad (5.23)$$

Quando comparado o limite superior do intervalo que contem valor teórico do T_{RR} com o obtido experimentalmente para o *Tempo de Ciclo* (Fig 5.20), verifica-se que este último é 1.28 vezes superior para valores baixos de BER e 5.44 vezes superior para o cenário 10^{-3} BER.

Este é um indicador que revela que mesmo em termos médios as perturbações identificadas podem afectar significativamente o desempenho da rede. Estas perturbações podem traduzir-se em inibições de comunicação ou atrasos do *token* que levam a uma redução do *Token Holding Time*. Este cenário pode conduzir ao incumprimento de deadlines de mensagens em sistemas de tempo-real, ou mesmo a fazer o sistema evoluir para estados de segurança que fazem diminuir a sua disponibilidade.

5.3.3 Avaliação do Funcionamento com Comunicações de Tempo-real

A resposta de tempo-real é um dos importantes parâmetros que devem ser tidos em consideração no projecto de redes de campo. Neste sentido assume grande relevância a forma como na execução dos serviços de comunicação e para os mais diversos cenários de operação as redes desempenham.

O PROFIBUS-DP no seu perfil base *DP-V0* direcciona a totalidade dos serviços de comunicação de dados para suportar aplicações, em que os requisitos típicos exigem a troca de informação em mensagens de alta prioridade, com curtos ciclos de produção e recepção. Este perfil abrange assim, aplicações em que é necessário assegurar o cumprimento de restrições temporais críticas.

O desempenho de tempo-real da rede PROFIBUS-DP assim como, da sua degradação em condições de faltas é apresentada para um tráfego típico do perfil *DP-V0*, de acordo com as duas configurações de operação características desta rede: multi-mestre e mono-mestre.

5.3.3.1 Operação em Modo Multi-Mestre

A avaliação da operação no modo multi-mestre representa uma extensão da avaliação preliminar tendo contudo um objecto de avaliação diferente: agora centrado no desempenho da rede na execução dos serviços de comunicação de dados. Em relação à avaliação preliminar verifica-se uma evolução da estrutura da rede com a inclusão de estações passivas, adicionando uma nova dimensão no tráfego da rede correspondente às comunicações mestre escravo.

Os serviços de comunicação de dados suportam uma estrutura de dados com 6 bytes no campo de dados (DU - *Data Unit*), trocada nos dois sentidos da comunicação mestre escravo (fig 5.4). Esta configuração foi especificada para conciliar dois propósitos:

- Ser representativa da informação típica de um conjunto de aplicações industriais;
- Não contribuir de forma acentuada para o comprimento final da trama, diminuindo assim a probabilidade deste se tornar num factor que aumente de forma considerável a incidência de erros nestas tramas.

O conhecimento adquirido na avaliação preliminar foi utilizado para proporcionar as melhores condições de desempenho da rede, mesmo quando esta opera em cenários de faltas. Designadamente através da redução dos tempos de recuperação dos eventos que geram instabilidade no anel lógico. Neste contexto, na configuração da rede foram incluídas as seguintes opções:

- Atribuir à primeira estação do anel um endereço baixo (um). Esta opção, reduz de forma significativa o tempo de *timeout* e consequentemente o tempo de recuperação da perda de *token*. A redução temporal do *timeout*, contribui igualmente para atenuar o atraso introduzido pela má operação deste mecanismo quando o barramento é afectado por faltas, e que deriva num aumento efectivo do tempo de *timeout*;
- Diminuir o comprimento da GAP assim como, do parâmetro HSA. Desta forma é possível reduzir tempos de inserção de estações na rede, bem como, de reduzir o número de mensagens geradas nos ciclos de actualização da GAP *List*.

A infra-estrutura de comunicação que suportou as experiências foi configurada com três estações activas, tendo cada uma delas na sua esfera de controlo três estações passivas. Na vertente funcional da configuração, a FDL das estações manteve os parâmetros apresentados na tabela 5.1, à excepção do T_{TR} que assume

para estas condições de ensaio, uma maior relevância na regulação da operação da rede.

O valor de T_{TR} foi dimensionado para um cenário, que possibilite a cada estação transmitir uma trama de dados endereçada a cada uma das estações passivas, realizar um serviço *Request FDL Status* ($T_{M_{\mathrm{Re}q\,FDL}C}$), bem como garantir tempo para a retransmissão de uma mensagem por cada rotação do *token* ($T_{M_{Data\,\mathrm{Re}try}C}$).

$$T_{BCT} = n_a \cdot \left[n_p \cdot T_{M_{Data}C} + T_{M_{\mathrm{Re}q\,FDL}C} + T_{TC} \right] + T_{M_{Data\,retry}C} \qquad (5.23)$$

n_a - número de estações activas;

n_p - número de estações passivas.

$T_{M_{Data}C}$ - tempo do ciclo de uma mensagem de dados;

$$T_{M_{Data}C} = T_{S/R} + T_{SDR} + T_{A/R} + T_{ID} + T_{TD} \qquad (5.24)$$

As componentes do ciclo de mensagem de dados são respectivamente:

- $T_{S/R}$ - tempo de transmissão de uma trama com os dados de saída e respectivo pedido de dados de entrada;
- T_{SDR} - tempo de atraso imposto pela estação na resposta ou na confirmação a um serviço de comunicação;
- $T_{A/R}$ - tempo da transmissão da trama de confirmação ou de resposta a um serviço de envio ou pedido de dados.

O valor do *Tempo de Ciclo* (T_{BCT}) para a configuração da rede e cenário de carga apresentado é de 8.2ms. De forma a evitar um efeito de deslizamento ao longo do anel, ou seja que de forma sequencial uma estação fique sem *Holding Time* para transmitir os serviços de comunicação, o dimensionamento de T_{TR} foi efectuado de acordo com (5.25).

$$T_{TR} = T_{BCT} + T_{M_{Data}C} \qquad (5.25)$$

No ASPC2 este parâmetro foi ajustado para 11.26 ms.

Configuração das experiências

As experiências foram realizadas para condições operacionais definidas pelo produto cartesiano $Op = L \times F$.

O PROFIBUS-DP tem um padrão de operação síncrono, o qual apresenta um nível de utilização do barramento máximo, ou seja a largura de banda é sempre utilizada no seu valor máximo independentemente das condições de carga da rede. Desta forma controlado o tempo de chegada das mensagens de dados à FDL é possível definir padrões de tráfego no barramento nos quais ora existe uma predominância de tramas de dados ou pelo contrário o tráfego de gestão ganha primazia.

Desta forma foram definidas as condições de carga (L) para um sistema típico de *polling*, em que o tempo de chegada de mensagens de dados à FDL (*interarrival time*), oriundas da camada de aplicação contempla três cenários:

- **Carga elevada**: Com tempo de chegada de mensagens de dados à FDL de cerca de $0.4 \cdot T_{TR}$;
- **Carga média**: Com tempo de chegada de mensagens de dados à FDL de cerca de $1.17 \cdot T_{TR}$;
- **Carga baixa**: Com tempo de chegada de mensagens de dados à FDL de cerca de $2.86 \cdot T_{TR}$.

Desta forma os valores temporais escolhidos para a chegada de mensagens à FDL configuram três padrões de tráfego na rede, que permitem verificar influência dos erros que afectam as mensagens e designadamente as tramas de gestão, no desempenho de tempo real.

O cenário de carga elevada gera um padrão de tráfego no qual a relação entre a ocupação do barramento com tramas de dados e as tramas de gestão está acentuadamente desequilibrada em favor do primeiro (fig. 5.23). Concretamente 69,2% são tramas de dados, o restante tráfego refere-se às tramas de *token* e *FDL Request Status* que apresentam respectivamente valores de 23,7% e 7%. Nesta situação a rede apresenta o valor mínimo de tráfego de gestão essencial ao funcionamento da rede.

Os dois restantes cenários representam situações em que as aplicações fazem uma utilização menos intensiva dos serviços de comunicação. Assim, engloba um cenário de média carga no qual a percentagem de tramas de dados decresce para 17,7% do tráfego observado no barramento, e um cenário de comunicação mais esporádico com o volume de tráfego de dados a situar-se nos 7% do tráfego total no barramento. Este decréscimo resulta num aumento de tramas de *token* que cresce para 58,8% no cenário de média carga e atinge o valor 69,7% no cenário de carga baixa.

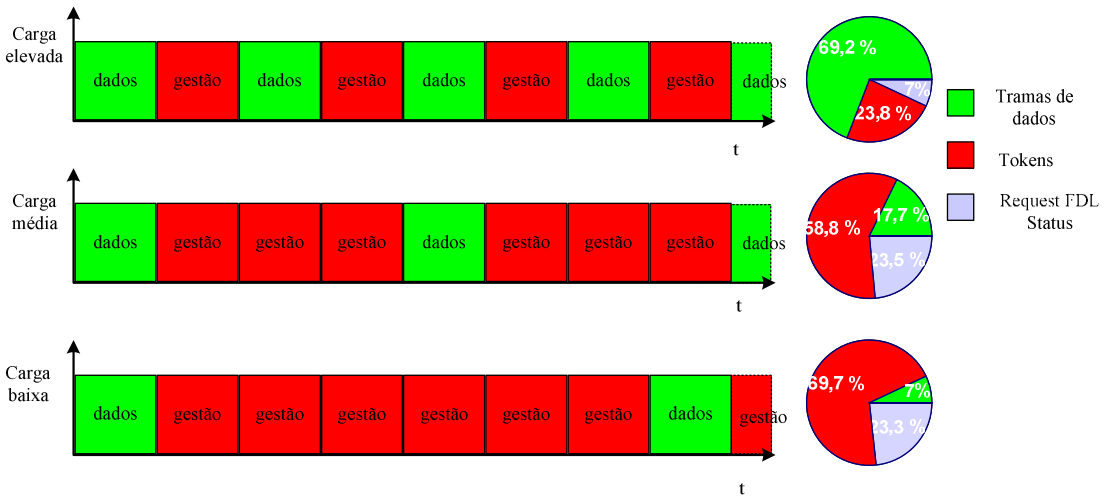

Figura 5.23 - Padrão do tráfego no barramento.

Não obstante a máxima utilização da largura de banda, mesmo em cenários onde a carga é baixa, a variação do *Holding Token Time* permite fazer um ajuste dessa mesma utilização orientando-a para o desempenho da rede. Neste contexto T_{RR} vai variando segundo as condições de utilização da rede fornecendo a cada estação um maior ou menor tempo de *token*. No limite, esta gestão do tempo de acesso ao meio produzirá numa primeira instância, uma supressão de mensagens de baixa prioridade disponibilizando tempo suplementar para os ciclos de mensagens de tempo-real.

O cenário de falta (*F*), que afecta o barramento é representado pelo conjunto de três taxas de erro (BER), que durante a avaliação preliminar foram identificadas como representativas de pontos que definem intervalos de tendências do funcionamento da rede em cenários de faltas.

$$F = \left\{1 \cdot 10^{-4}, 5 \cdot 10^{-4}, 1 \cdot 10^{-3}\right\}$$

A configuração das faltas nos ensaios permanece constante com valor igual 1 BEL.

Na avaliação à operação da rede são efectuadas as seguintes medidas:

(i) ***Interrupção do Serviço do Sistema***: À imagem da avaliação efectuada à estabilidade do anel lógico, este estimador é novamente especificado para proceder à avaliação da susceptibilidade da rede perder *tokens* em diferentes condições de carga. No caso da configuração multi-mestre, serve para confirmar nos termos em que o estimador é definido, a independência do evento face a outra actividade no barramento. Mas essencialmente é definido para avaliar a susceptibilidade a este evento da configuração mono-mestre e correlacioná-la com a obtida na configuração evolvendo mais que um mestre.

O estimador é especificado como a probabilidade de perda de um *token* sempre que este é transmitido no barramento, e resulta da relação entre o número de *tokens* transmitidos e *tokens* perdidos durante as experiências de injecção de faltas.

$$\hat{p}_{SO} = \frac{1}{n}\sum_{j=1}^{n}\frac{T_{KL_j}}{T_{K_j}} \qquad (5.26)$$

n - número de experiências;

T_{KL_j} - número de *tokens* perdidos durante a realização da experiência *j*;

T_{K_j} - número de *tokens* observados no barramento durante a realização da experiência *j*.

(ii) ***Tempo de Ciclo***: Este estimador foi especificado para fornecer uma indicação, do tempo médio que uma estação aguarda pelo *token* em configuração multi-mestre. Em termos de cálculo é obtido a partir da

média dos tempos médios contabilizados a partir de cada ciclo (rotação) do *token* obtido em cada experiência de injecção de faltas.

$$\hat{\mu}_{BCT} = \frac{1}{n}\sum_{j=1}^{n}\frac{T_{BCT_j}}{N_{BC_j}} \qquad (5.27)$$

n - número de experiências;

T_{BCT_j} - tempo dos ciclos de barramento total da experiência j;

N_{BC_j} - número de ciclos de barramento realizados pela rede durante a experiência j.

(iii) ***Tempo Médio de Resposta***: Este estimador foi especificado para determinar o valor médio da latência dos serviços de comunicação de dados ao nível da FDL. No PROFIBUS-DP a troca de informação entre estações activas e passivas é efectuada por iniciativa da estação activa de acordo com um ciclo *polling*, cuja frequência é imposta pela aplicação. A passagem de dados da aplicação processa-se quase de imediato para a camada do utilizador que invoca os serviços da FDL para efectuar a transmissão. A FDL coloca os pedidos em fila de espera, e só procede à execução dos serviços de comunicação, quando ganhar o acesso ao meio e tiver tempo de posse de *token* (*Token Holding Time*) suficiente para os processar. Desta forma a componente principal da latência das comunicações ocorre ao nível da FDL e pode ser significativamente afectada pelos eventos que afectem o acesso da estação ao meio de comunicação, ou de erros que são tratados ao nível da camada de ligação de dados.

Assim, este estimador é calculado com base na média das diferenças dos tempos de chegada dos pedidos para serviço de comunicação de dados à FDL e correspondente recepção da resposta por parte da estação destinatária.

$$\hat{\mu}_{RT} = \frac{1}{n}\sum_{j=1}^{n}\frac{T_{Rm_j}}{N_{m_j}} \qquad (5.28)$$

n - número de experiências;

T_{Rm_j} - tempo de latência total verificada nos serviços de comunicação de dados realizados na experiência j;

N_{m_j} - número de serviços de comunicação efectuados durante a experiência j.

(iv) ***Tempo de Resposta no Pior Caso***: Nos sistemas de controlo de tempo real, as tarefas por este realizadas devem ser contidas no tempo, e ser conhecido o seu limite superior. Este limite superior é conhecido como tempo de resposta no pior caso. O estimador definido diz respeito ao pior tempo verificado na execução dos

serviços de comunicação de dados. O processo de contabilização deste tempo é efectuado de forma semelhante à descrita para a determinação do tempo médio de resposta, sendo neste caso unicamente registado o maior tempo observado. O valor final do estimador resulta da média dos tempos de resposta no pior caso obtidos em cada experiência de injecção de faltas.

$$\hat{\mu}_{WCRT} = \frac{1}{n}\sum_{j=1}^{n} T_{WCRT_j} \qquad (5.29)$$

n - número de experiências;

T_{WCRT_j} - pior caso de resposta temporal verificado na transacção de um serviço de comunicação de dados registado durante a experiência j.

Incumprimento do Tempo Limite: Este estimador fornece uma indicação da probabilidade do não cumprimento de uma *deadline* associada à recepção de uma mensagem. No PROFIBUS-DP o T_{TR} é um parâmetro, que deve ser configurado de forma a permitir que todas as estações activas tenham pelo menos a possibilidade de encetar troca de dados com as estações passivas da sua esfera de controlo, estabelecendo assim, um tempo máximo de *polling* para a rede. Neste contexto, foram atribuídas às mensagens *deadlines* com base em múltiplos do T_{TR}. O cálculo da probabilidade da perda de uma *deadline* é efectuado verificando se a recepção das mensagens se produz dentro dos limites temporais especificados. Quando isto não se verifica é registado o número da mensagem em que ocorreu o incumprimento da *deadline* e com base no seu inverso obtém-se um elemento da amostra que permite calcular a probabilidade de perda de uma *deadline*. Com o objectivo de evitar uma possível fonte de enviesamento do estimador, os dados recolhidos da experiência que são posteriores a este evento são descartados.

$$\hat{p}_D = \frac{1}{n}\sum_{j=1}^{n} \frac{D_j}{N_{m_j}} \qquad (5.30)$$

n - número de experiências;

D_j - estado da integridade das *deadlines* dos serviços de comunicação de dados realizados durante a experiência j. Esta variável assume o valor 0 se não ocorreu nenhuma violação de *deadline*, ou 1 caso contrário.

N_{m_j} - número de serviços de comunicação de dados realizado na experiência j. Esta variável fornece a totalidade dos serviços de comunicação de dados efectuados na experiência j no caso de não ter ocorrido nenhuma violação das *deadlines*. No caso do desrespeito de

uma *deadline* a variável assume o número de serviços de comunicação de dados verificados até esse evento.

A metodologia utilizada no cálculo dos estimadores é baseada no método das replicações independentes descrito em §4.7.2. Os estimadores foram calculados para um intervalo de confiança de 95%. A avaliação foi centrada na obtenção do valor da latência dos serviços de comunicação, pelo que a largura deste intervalo apresenta um valor inferior a 1% do valor do estimador. Valor que é igualmente apresentado para o intervalo do estimador do *Tempo de Ciclo*. Nos restantes estimadores este indicador de precisão foi relaxado, apresentando a média do *Tempo de Resposta no Pior Caso* e *Interrupção do Serviço do Sistema* larguras de intervalo inferiores a 15% do valor do estimador. O estimador que mede a susceptibilidade da rede perder deadlines, apresenta valores elevados para este parâmetro, pelo que são apresentados na dissertação como indicadores de tendência do comportamento da rede para as condições de ensaio.

I. Interrupção do Serviço do Sistema

Durante a avaliação preliminar a *Interrupção do Serviço do Sistema* foi identificado como a principal causa de perturbação temporária da função de comunicação da rede PROFIBUS-DP. Da mesma forma este evento representa o evento com maior impacto na operação global da rede.

A avaliação da *Interrupção do Serviço do Sistema* nas condições de carga que agora são descritas vem complementar a caracterização da susceptibilidade a faltas PROFIBUS-DP.

Na Figura 5.24 está representada a probabilidade da *Interrupção do Serviço do Sistema* em função do BER para cada um dos cenários de carga. A influência da carga, e principalmente variações desta não têm influência na probabilidade de um *token* vir a ser perdido quando as estações negoceiam a sua passagem. Apesar da ocorrência do evento nas experiências com carga ser evidentemente inferior à verificada nas condições de ausência ou mesmo de carga reduzida, em cada geração de *token* existe a mesma probabilidade de este se perder e levar a rede para uma situação de inibição de comunicação da rede.

Esta descrição do comportamento da *Interrupção do Serviço do Sistema* pode ser suportada pelos valores que a probabilidade deste evento apresenta em condições de ausência de carga (Fig. 5.6 – 1 BEL), que são semelhantes às apresentadas na Figura 5.24. A diferença verificada para 10^{-3} BER pode ser enquadrada em grande parte na precisão do valor do estimador $\hat{p}_{SO}$ (5.26), e na fonte suplementar que contribui para a ocorrência deste evento descrita em §5.3.2.6 - I, que é ela própria fonte das divergência verificadas para taxas elevadas de erros, das curvas apresentadas na (Fig. 5.6).

Em suma pode considerar-se que as condições que levam à perda de *token* e sua probabilidade, obtidas na avaliação preliminar, são aplicáveis para a operação da rede com carga.

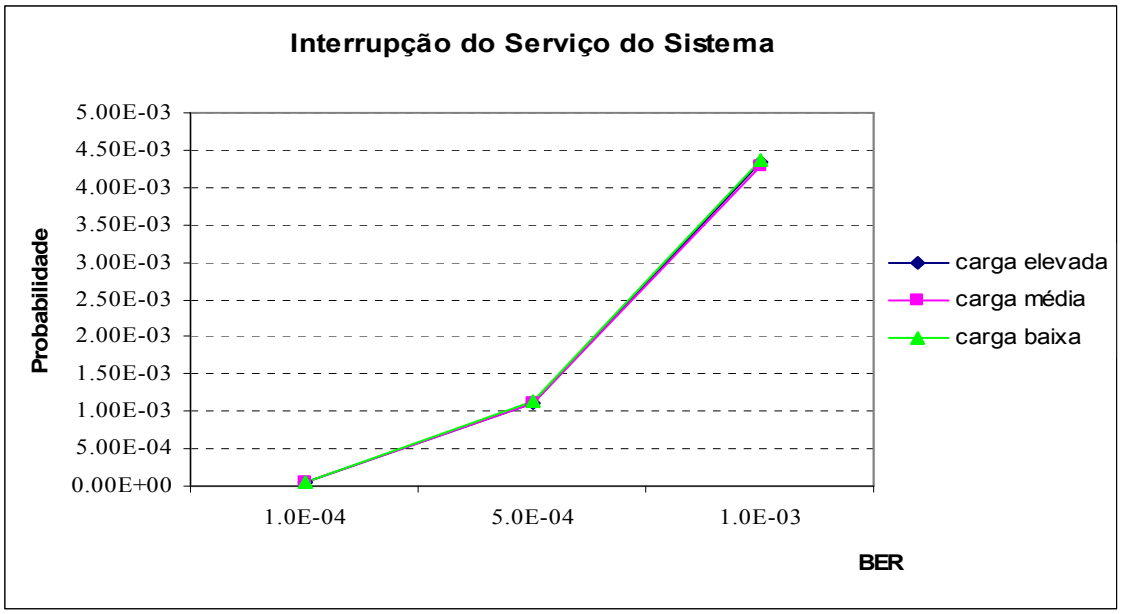

Figura 5.24 - Influência da carga na Interrupção do Serviço do Sistema.

II. Tempo de Ciclo

O *Tempo de Ciclo* fornece uma indicação do tempo que uma estação aguarda em média pela recepção de um *token*, ou seja o tempo médio para obter o acesso ao meio.

Na Figura 5.25 está representado o tempo médio entre recepções de *token* observado em cada uma das estações que constituem o anel lógico. O valor apresentado está fortemente dependente do nível de carga da rede. Isto pode ser comprovado para taxas de erros baixas 10^{-4}, onde o valor representado na figura é o valor aproximado do tempo de ciclo do barramento T_{BCT}. Este valor pode ser obtido através da expressão (5.23) para cada cenário de carga.

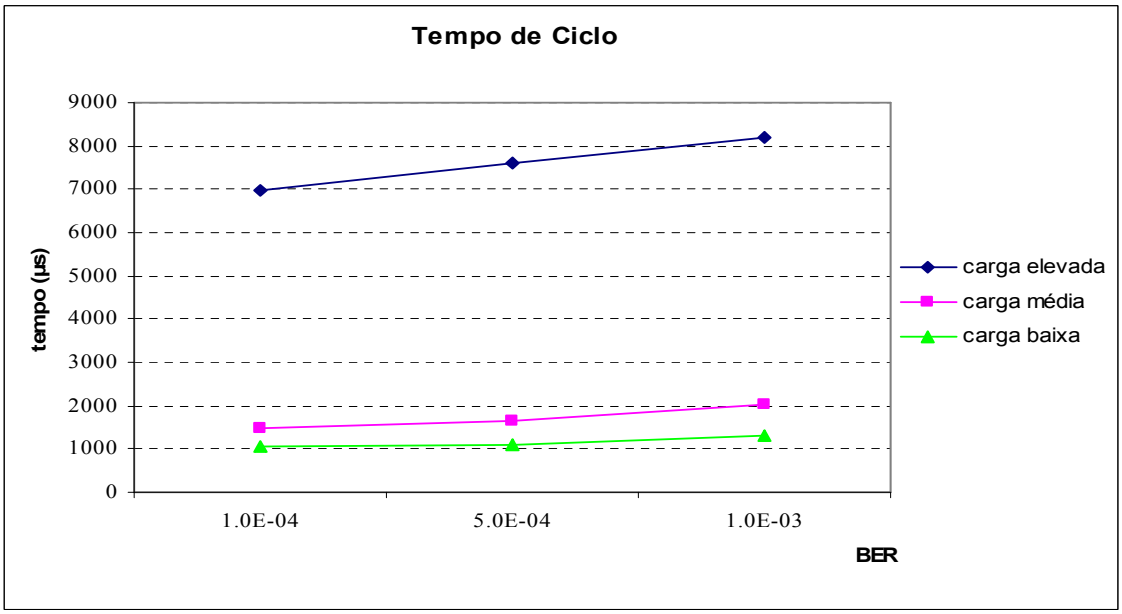

Figura 5.25 - Comportamento *Tempo de Ciclo* em cenários de carga.

O efeito dos erros no *Tempo de Ciclo* manifesta-se de forma moderada mas com um aumento consolidado que em conjunto com o aumento da carga produz pequenas diferença no declive entre linhas que descrevem este indicador para os três cenários de carga. Uma forte componente desta variação fica a dever-se a retransmissões das tramas de dados que são corrompidas pelos erros.

Este comportamento contrasta com o observado na avaliação preliminar no qual, para taxas de erros elevadas, o *Tempo de Ciclo* cresce de acordo com uma lei exponencial. Esta diferença de comportamento deve-se essencialmente à alteração dos endereços de rede das estações, nomeadamente à atribuição do endereço 1 à estação com endereço mais baixo. Esta alteração tem um impacto a dois níveis:

- Em condições normais reduz o tempo de geração de um novo *token* de 9.6ms para 3.2ms;
- O efeito evidenciado pela má operação do mecanismo de timeout é atenuado nesta configuração. O facto de reduzir em três vezes o tempo de *timeout* faz com que este valor esteja significativamente mais próximo do valor médio esperado para a ocorrência das faltas com BER elevados, como é constatável através da expressão (5.8). Esta aproximação do valor médio esperado para a ocorrência das faltas reduz o atraso do tempo de recuperação da perda de *token*.

O comportamento do *Tempo de Ciclo* apresentado na Figura 5.25 mostra que o comportamento da rede PROFIBUS-DP é influenciado de forma significativa pelo valor atribuído ao endereço dos nós de comunicação. Mostra igualmente que uma escolha cuidada desse parâmetro embora não resolva os problemas associados ao mecanismo de recuperação de perda de *token*, a sua resposta em termos médios é melhorada mascarando muitas das suas insuficiências.

III. Tempo de Ciclo das Mensagens

A resposta temporal fornecida pela rede nos seus serviços de comunicação são um aspecto de grande relevância em redes de comunicações de tempo-real e consequentemente também o são nas redes de campo. Na avaliação da resposta de tempo-real da rede PROFIBUS-DP foram definidos dois estimadores que permitem obter informação do desempenho da rede na execução dos serviços de comunicação de dados de alta prioridade. Esses estimadores incluem nomeadamente a latência dos serviços de comunicação e a pior resposta que pode ser esperada na execução dos mesmos.

Na Figura 5.26 são apresentados os tempos de latência dos serviços de comunicação de dados em função do BER para os três cenários de carga.

Em cenários de carga elevada o aumento da latência dos serviços de comunicação de dados tem uma componente elevada de repetições das mensagens. Neste cenário, a capacidade de ajustamento do tempo de posse do *token* (*Holding Time*) para absorver as alterações de carga provocadas pelas

repetições é mais limitada que em outras situações. Isto reflecte-se numa maior taxa do aumento da latência.

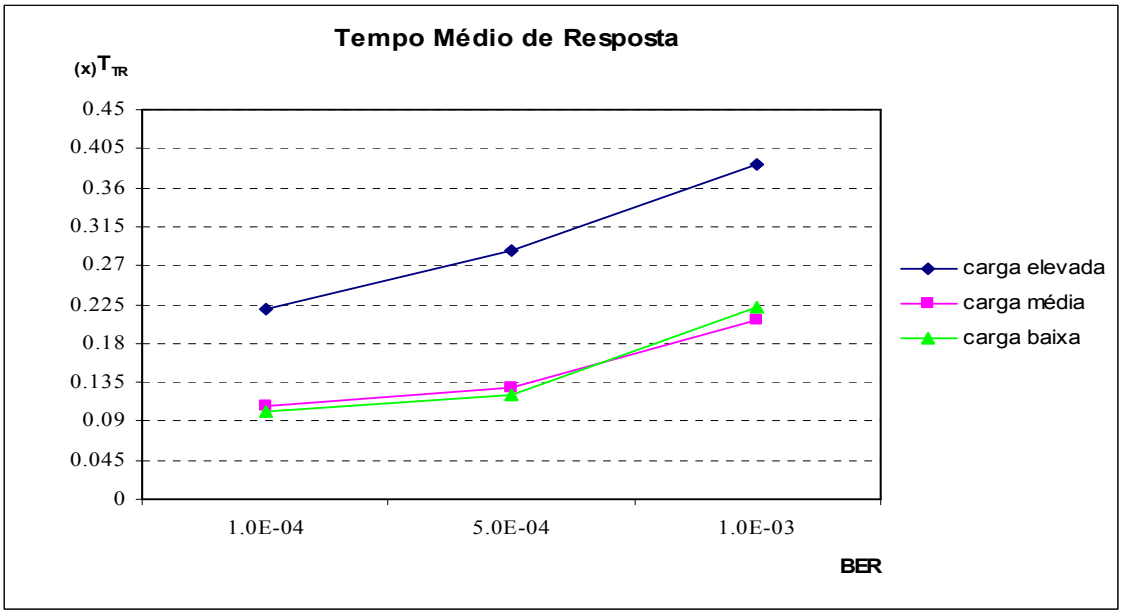

Figura 5.26 - Latência dos serviços de comunicação de dados.

Em cenários de baixa e média carga a rede apresenta melhores características de resposta temporal aos dos serviços de comunicação, o que pode ser constatado pela baixa latência apresentada na figura 5.26. Nestes cenários de carga o contributo dos tempos de latência associados a perdas de *token* e sua recuperação tornam-se mais relevantes. A degradação do tempo de latência torna-se mais acentuada para taxas superiores a $5 \cdot 10^{-4}$ BER em linha com o comportamento da *Interrupção do Serviço do Sistema* (Fig. 5.24).

Os tempos de latência representados na Figura 5.26 fornecem uma representação pesada do tempo dos ciclos de mensagem. Em sistemas de tempo-real o pior caso de resposta assume particular importância para o desenvolvimento de escalonamentos que satisfaçam os requisitos destes sistemas. Na figura 5.27 encontra-se uma representação do tempo de resposta no pior caso WCRT do PROFIBUS-DP para as condições de ensaio.

Ao contrário do estimador da latência média esperada, que entra em consideração com a totalidade dos tempos de ciclos dos serviços de comunicações, o estimador representado na figura 5.27 é obtido com base nas contribuições dos WCRT obtido em cada ensaio. Desta forma, o estimador é construído a partir das situações mais severas de operação e em consequência a sua representação tenderá a revelar esses mesmos contributos. Neste contexto, fica evidenciada para taxas elevadas de erros um forte contributo dos tempos de recuperação de perdas de *token* para o estimador da média dos piores casos de resposta. Igualmente para cenários de baixa e media carga são gerados mais *tokens* que em cenários de carga elevada e consequentemente produzir-se-ão mais perdas de *token*. Estes factores propiciam condições para que as insuficiências latentes no mecanismo de recuperação do *token* se manifestem.

Isto pode ser constatado pela intercepção das curvas dos cenários de média carga com a curva do cenário carga alta para taxas elevadas de erros.

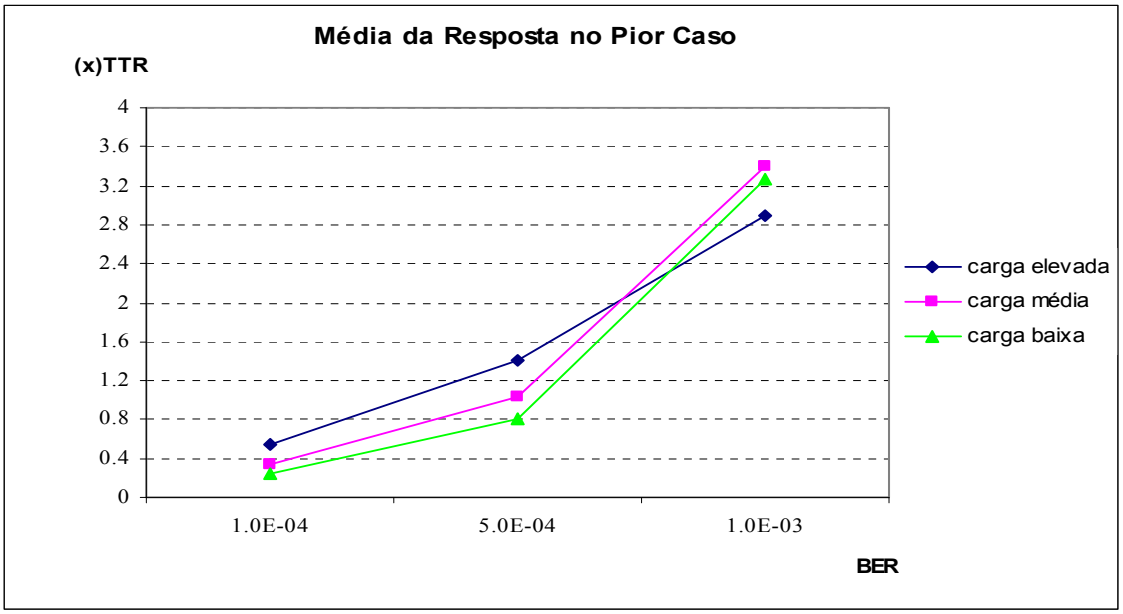

Figura 5.27 - Pior caso de resposta em serviços de comunicação de dados.

Em síntese a rede tende a apresentar um tempo de resposta aos serviços de comunicação substancialmente melhor quando a carga do sistema se tende a afastar do cenário de carga máxima. Contudo, para taxas de erros elevadas o WCRT tende a assumir um valor superior ao exibido em cenários de carga elevada devido à maior probabilidade de serem geradas situações que leve o mecanismo de recuperação de perda de *token* a apresentar tempos de recuperação elevados.

IV. Incumprimento do Tempo Limite de Recepção da Mensagem

Nos sistemas de controlo, variações ao normal tempo de execução das acções de controlo resultam de uma forma geral na degradação do serviço fornecido pelo sistema. De acordo com a função do sistema, atrasos de algumas acções de controlo podem mesmo colocar em risco pessoas e bens. Essas acções são geralmente alvo de um escalonamento, atribuindo limites temporais – *deadlines* à sua realização.

Quando a informação de tais sistemas é trocada através de serviços de comunicação é também usual atribuir limites temporais à sua recepção. Assim, se a recepção dos serviços de comunicação não cumprirem as suas *deadline* o sistema pode falhar a sua função, ou poderá mesmo ter que suspender a sua operação e evoluir para um estado tido como seguro.

Nas figuras 5.28 a 5.30 são apresentadas curvas de probabilidade do não cumprimento de *deadlines*. Estas curvas são determinadas com base nos serviços de comunicação que não são concluídos num horizonte temporal de 1 T_{TR} e 2 T_{TR},

consequência de faltas que provocam alterações no equilíbrio do tráfego da rede. Na base da perda do equilíbrio está a necessidade de retransmissões de mensagens, e devido a situações de inibição temporária de comunicação provocada por eventos do tipo detectado durante a avaliação preliminar.

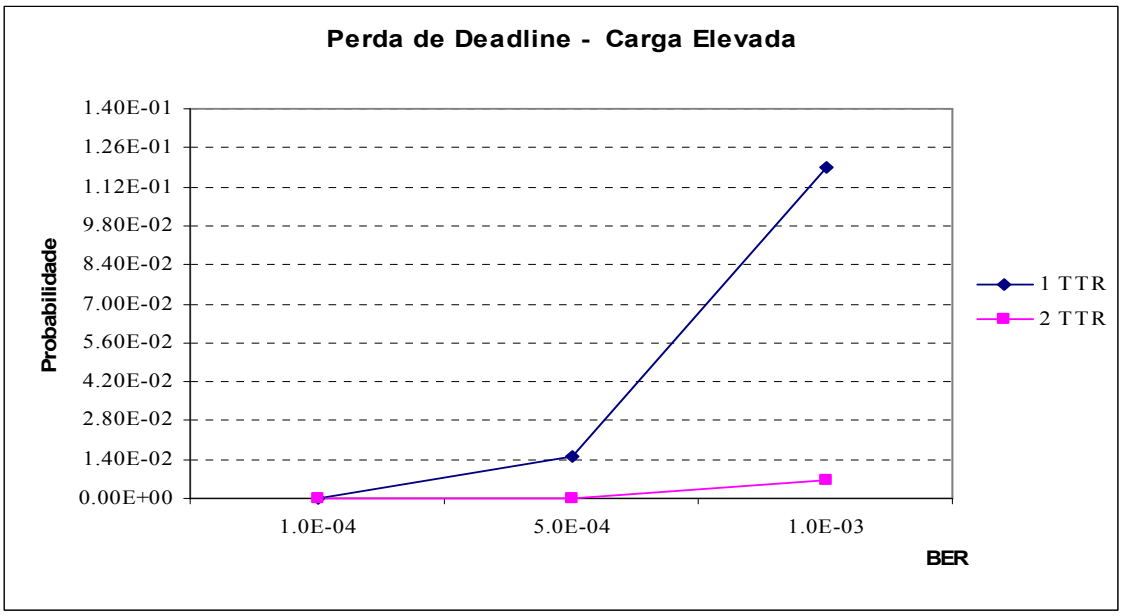

Figura 5.28 - Perdas de *deadline* em cenários de carga elevada.

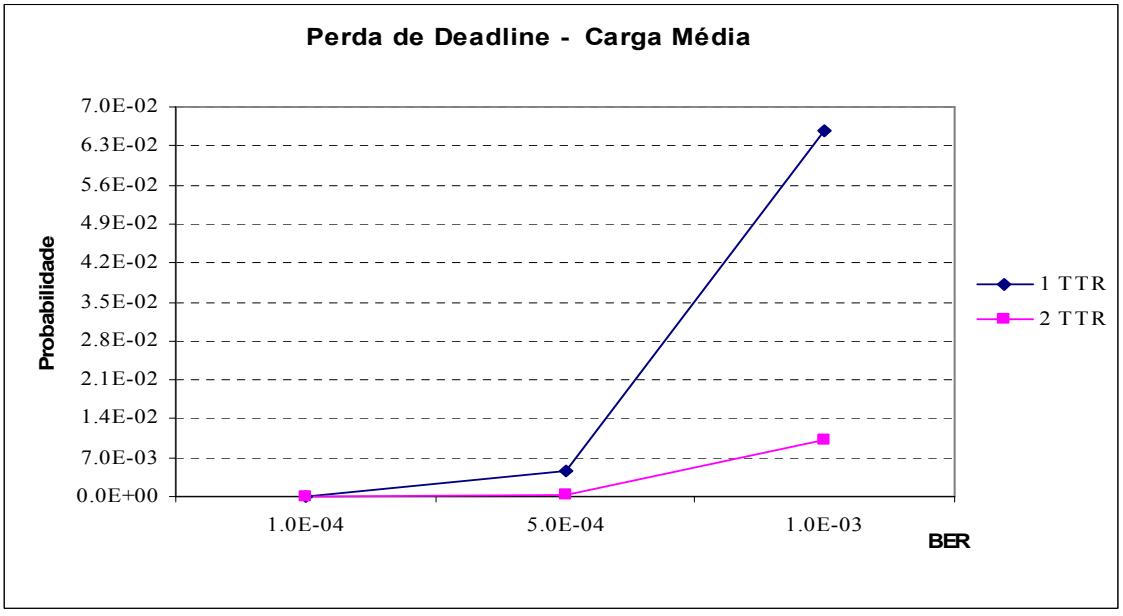

Figura 5.29 - Perdas de *deadline* em cenários de carga média.

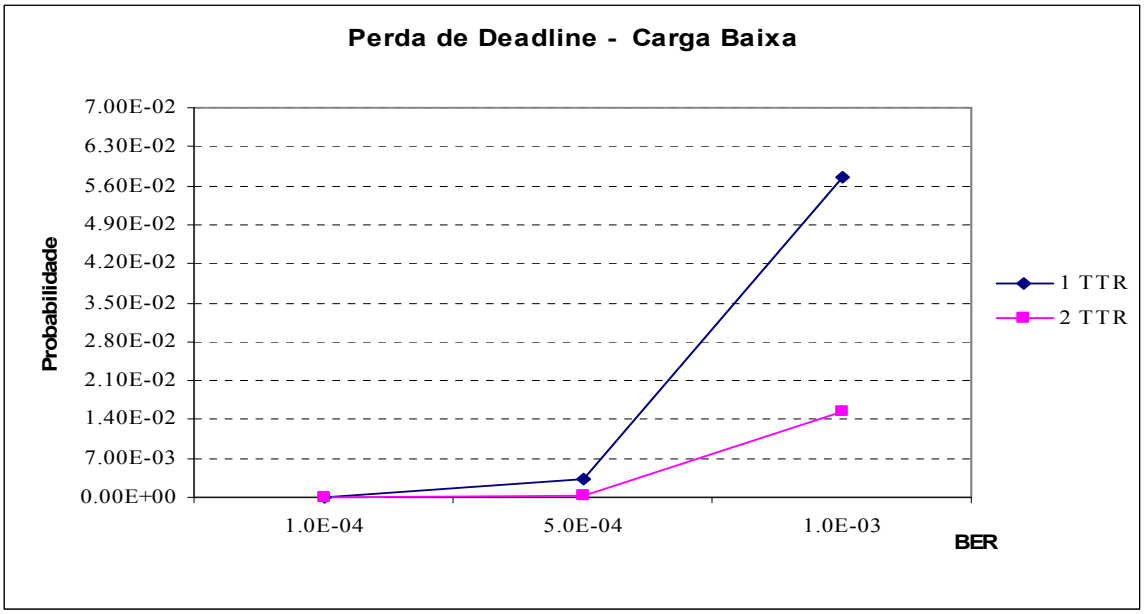

Figura 5.30 - Perdas de *deadline* em cenários de carga baixa.

5.3.3.2 Operação em Modo Mono-mestre

Quando existe um única estação activa numa rede PROFIBUS-DP o mecanismo descentralizado de controlo de acesso ao meio deixa de ser necessário evoluindo a rede para um modo de operação mestre escravo puro. A não partilha do acesso ao meio com outras estações reverte em melhores características temporais na execução dos serviços de comunicação da estação. A inexistência de um anel lógico estabelecido também poderá indiciar melhores características de estabilidade da rede.

Neste contexto, a avaliação da operação da rede PROFIBUS-DP em cenários de faltas foi estendida ao modo mono-mestre. Os ensaios foram efectuados para os mesmos parâmetros utilizados nas experiências em modo multi-mestre à excepção do T_{TR} que foi especificado para $T_{TR_{mono\ master}} = 0.54T_{TR}$. A estrutura física da rede é constituída por uma estação activa e três estações passivas.

O padrão do tráfego é naturalmente alterado relativamente à configuração multi-mestre (Fig. 5.23). No modo mono-mestre, a estação no final dos serviços de comunicação de dados efectua o envio de uma trama *Request FDL Status*, separando um novo ciclo de mensagens de dados através do envio de uma trama de *token* endereçada a si própria. Apesar destas diferenças, os tempos de produção de mensagens (*interarrival time*), permanece igual ao utilizado na configuração multi-mestre.

As medidas obtidas nas experiências são suportadas em quatros estimadores: *Interrupção do Serviço do Sistema* (5.26); *Tempo Médio de Resposta* (5.28); a média do *Tempo de Resposta no Pior Caso* (5.29) e *Incumprimento do Tempo Limite* (5.30)

À imagem da metodologia usada nas restantes experiências, os estimadores são obtidos através do método das replicações independentes com um intervalo de confiança de 95%. Os ensaios privilegiaram a avaliação da resposta temporal da rede. A largura do intervalo apresenta um valor inferior a 2% do valor do estimador – *Tempo Médio de Resposta*. Para os restantes estimadores a largura do intervalo não garante uma precisão elevada, pelo que estes são utilizados como indicadores de tendência de comportamento da rede para as condições de avaliação.

I. Interrupção do Serviço do Sistema

Um aspecto relevante na avaliação da operação da rede PROFIBUS-DP em modo mono-mestre consiste na verificação da susceptibilidade da rede a eventos que resultem da ocorrência de faltas, e contextualizar essa susceptibilidade com a observada na operação em modo multi-mestre.

Uma análise preliminar aos dados recolhidos nas experiências de injecção de faltas revela um aumento muito significativo da estabilidade da rede neste modo de operação. Para as condições de ensaio, não foi observado qualquer tipo de evento que conduza a perturbação do tipo *Interrupção do Serviço da Estação*. Da mesma forma, as fontes de *Interrupção do Serviço do Sistema* são consideravelmente reduzidas.

Na Figura 5.31 é apresentada a probabilidade do evento *Interrupção do Serviço do Sistema*. Não obstante a largura do estimador não garantir uma precisão suficiente para obter com rigor o valor da sua probabilidade, o conhecimento adquirido acerca do comportamento dos mecanismos que conduzem a este evento, permite inferir a existência de uma redução significativa relativamente à operação multi-mestre.

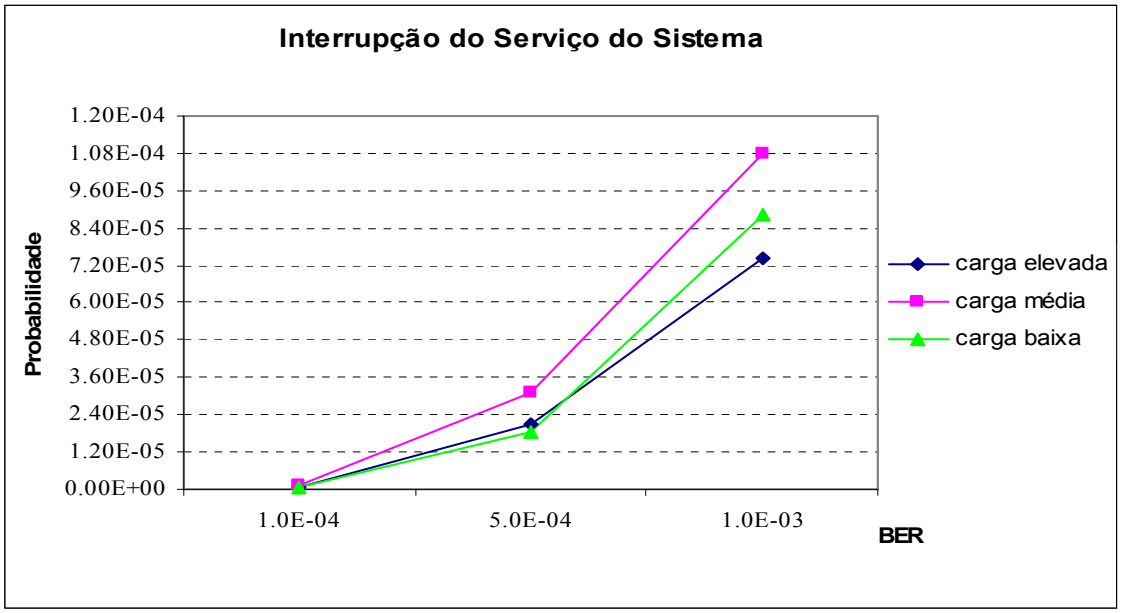

Figura 5.31 - Probabilidade do evento Interrupção do Serviço do Sistema em modo de operação mono-mestre.

A análise dos dados das experiências revela que a única fonte de perda de token detectada neste modo de operação está relacionada com a falsa detecção de erros permanentes nos *trasnsceivers – Erro Fatal*. Assim, tendo em consideração que a probabilidade do evento *Interrupção do Serviço do Sistema* como é definida nesta dissertação não é afectada por variações de carga, ou do tipo de erro, um aumento da precisão do estimador fará tender os valores apresentados na Figura 5.31, para os apresentados na Figura 5.7 (§5.3.2.6 –I).

II. Tempo de Ciclo das Mensagens

O PROFIBUS-DP apresenta no modo mono-mestre uma resposta temporal bastante homogénea mesmo em condições de carga elevada. O desempenho da rede está representado na Figura 5.32, onde é possível verificar para baixas taxas de erro (10^{-4} BER) uma diferença mínima da resposta média aos serviços de comunicação de dados relativamente aos diferentes cenários de carga.

A contrastar com a homogeneidade do modo mono-mestre na operação em modo multi-mestre existe uma diferença assinalável quando a rede opera com cargas elevadas (Fig 5.26). A mudança para cenários de carga mais moderados confere à rede desempenhos semelhantes aos obtidos pela configuração mono-mestre (Fig. 5.26, Fig. 5.32). Não obstante apresentar boas características temporais em cenários de carga moderada, o aumento da taxa de erro provoca uma degradação de desempenho muito mais significativa no modo multi-mestre. Este comportamento está associado aos fenómenos de instabilidade identificados, que se verificam no modo multi-mestre e que estão fortemente diminuídos no modo mono-mestre.

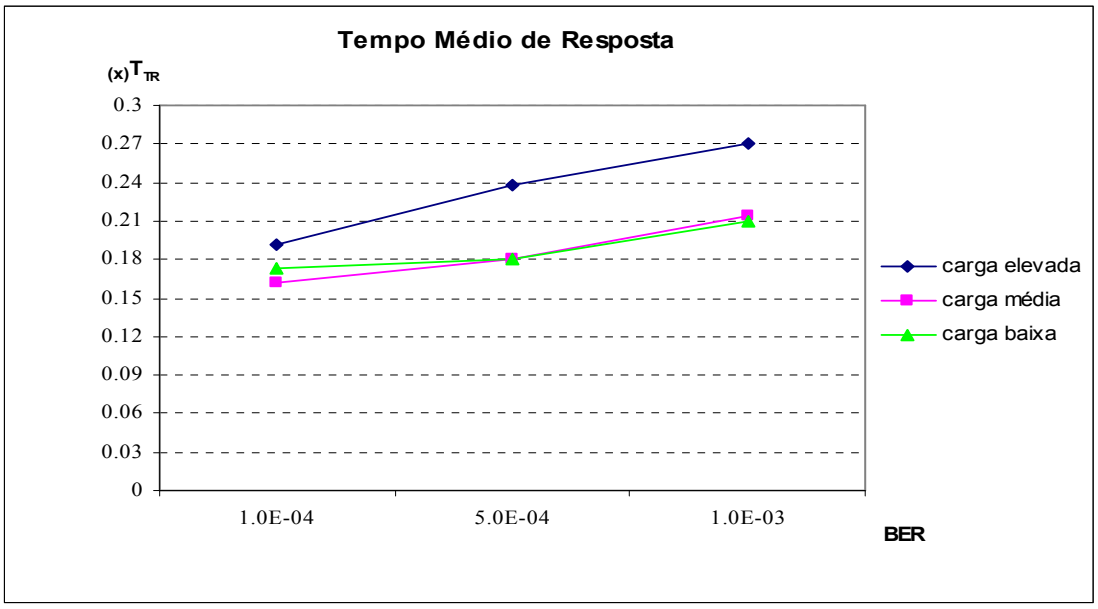

Figura 5.32 - Latência dos serviços de comunicação de dados no modo mono-mestre.

Apesar da melhor estabilidade da rede apresentada na operação em modo mono-mestre, esta continua a ser susceptível à entrada não desejável em modo de não operacionalidade (*Offline*) devido a falsas detecções de erros permanentes

nos *transceivers*[4]. Não obstante a probabilidade deste evento ser baixa, continua a representar uma situação bastante severa devido ao seu impacto ao nível da disponibilidade do sistema.

A avaliação da média WCRT mostra igualmente melhores características da configuração mono-mestre relativamente à configuração multi-mestre. Na Figura 5.33 está representado o comportamento do valor esperado para o pior caso de resposta, dos serviços de comunicação em função de três cenários de carga e da taxa de erros.

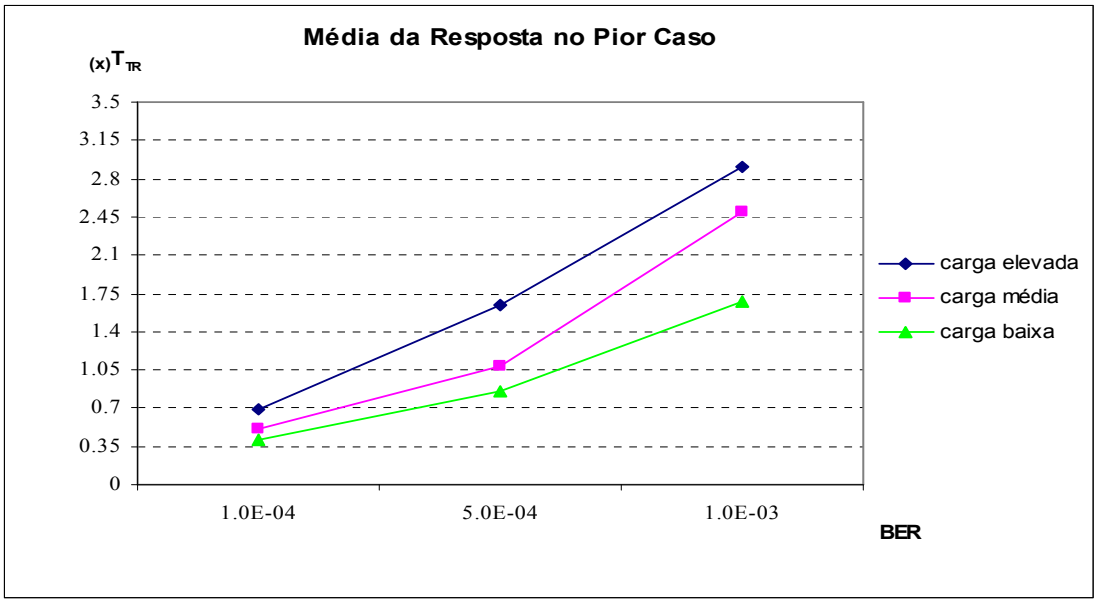

Figura 5.33 - Pior caso de resposta a serviços de comunicação no modo mono-mestre.

A figura revela um comportamento aproximadamente linear do aumento do tempo da média da resposta no pior caso em função da taxa de erros. Para este aumento contribui essencialmente as repetições de serviços de comunicação afectados por erros. O que confere piores casos de resposta substancialmente melhores que na configuração multi-mestre (Fig. 5.27).

Não obstante para baixos valores de BER, a configuração multi-mestre apresentar valores semelhantes aos verificados no modo mono-mestre, o aumento da taxa de erros revela os fortes impactos que as perturbações associadas à perda de *token* têm sobre o desempenho da rede.

III. Incumprimento do Tempo Limite de Recepção da Mensagem

À semelhança da avaliação efectuada no modo multi-mestre, são apresentados nas Figuras 5.34 a 5.36, gráficos com o valor esperado da probabilidade de perda de *deadline*, em função da taxa de erro para os três cenários de carga usados nas experiências.

[4] Os efeitos deste evento não são considerados na componente temporal da avaliação da rede.

As deadlines foram especificadas para múltiplos de 1 e de 2 do valor parametrizado para o T_{TR} da configuração mono-mestre. De forma a manter uma leitura coerente com os resultados dos demais ensaios, estes valores foram normalizados em relação ao valor do T_{TR} da configuração multi-mestre.

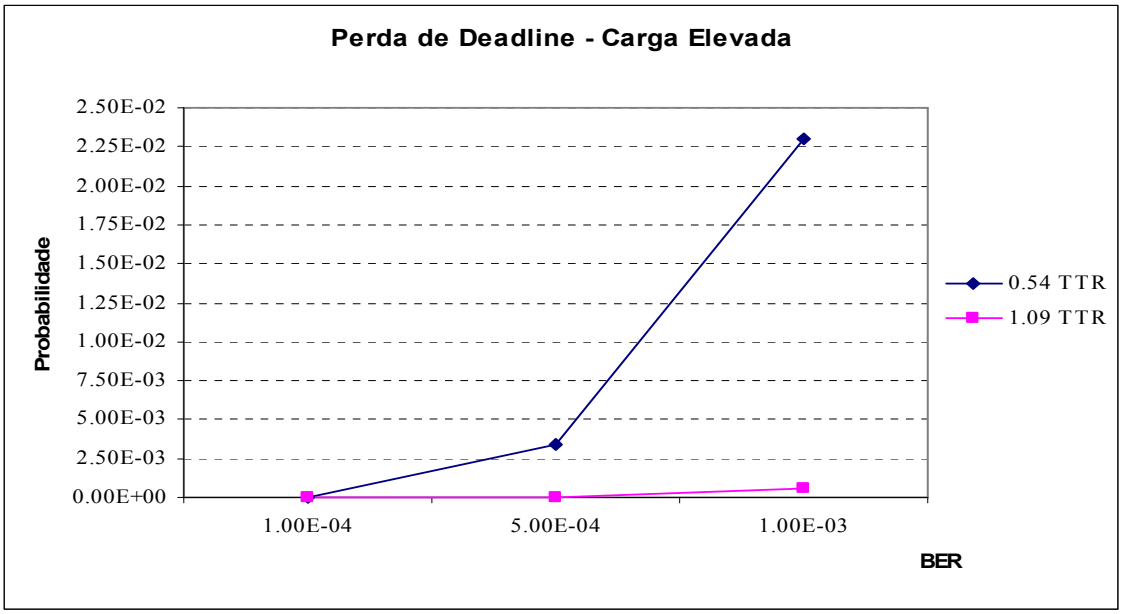

Figura 5.34 - Probabilidade de perda de *deadline* em cenário de carga elevada.

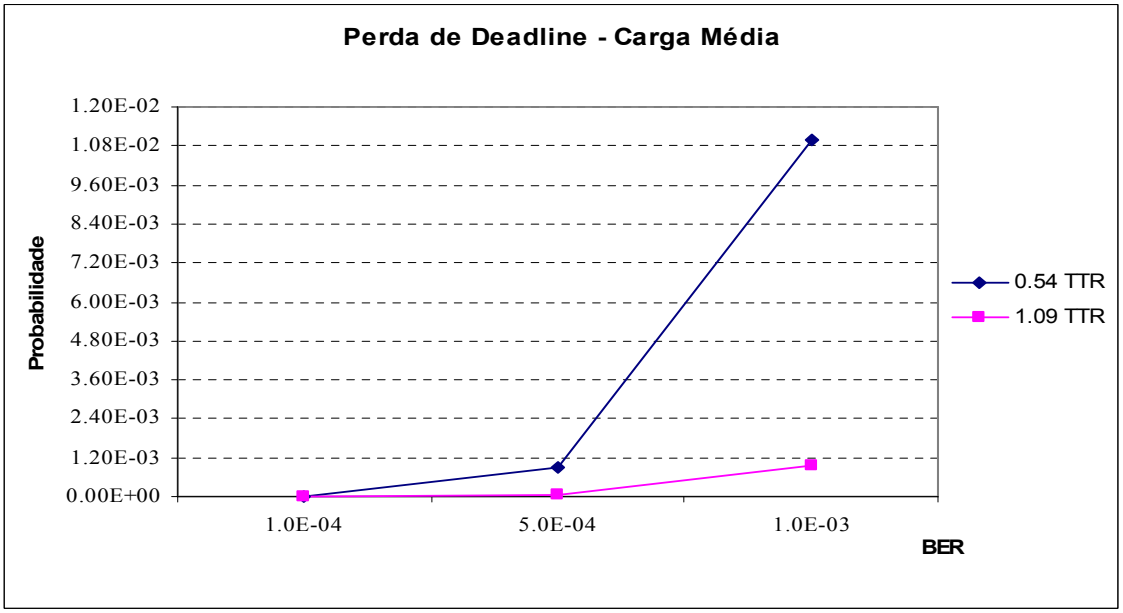

Figura 5.35 - Probabilidade de perda de *deadline* em cenário de carga média.

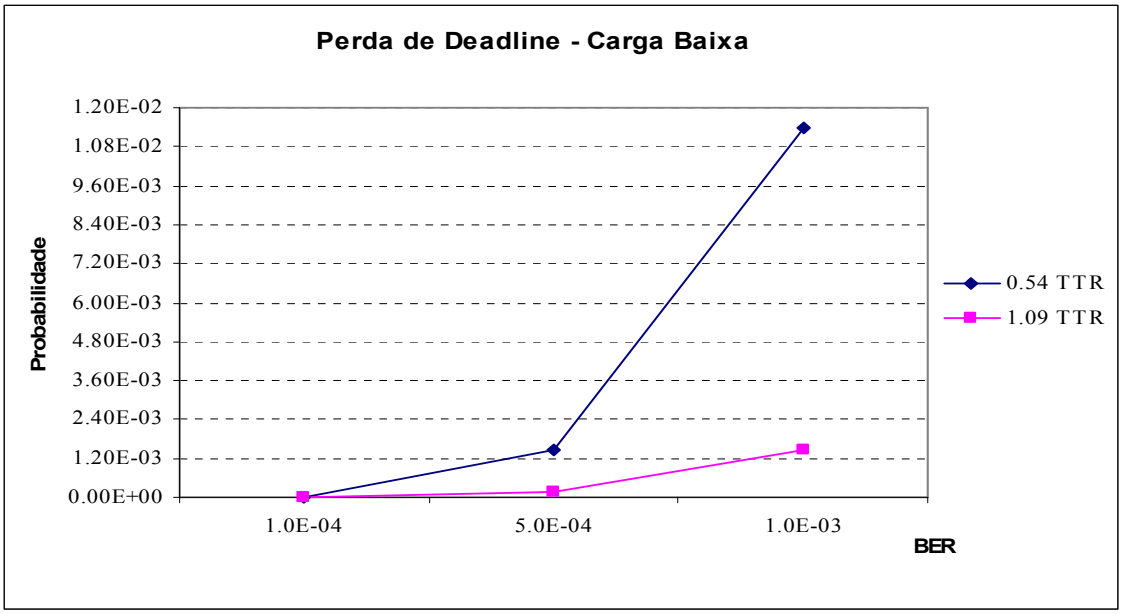

Figura 5.36 - Probabilidade de perda de *deadline* em cenário de carga baixa.

5.4 Síntese

Durante os ensaios de injecção de faltas, a rede foi avaliada em dois importantes aspectos do seu funcionamento.

(i) Identificação de modos de operação desencadeados pela ocorrência de faltas;

(ii) Desempenho das comunicações quando a rede é afectada por faltas que activam os modos de operação identificados.

No primeiro caso foi efectuada uma avaliação detalhada aos modos de operação e dos mecanismos que os desencadeiam. Muito do detalhe da avaliação está relacionada com a elevada complexidade da operação da rede em cenários de faltas, e das inúmeras cambiantes que resultam da interacção dos mecanismos de tolerância a faltas com os demais mecanismos que garantem a operação normal da rede, designadamente no controlo de acesso ao meio e na gestão da estrutura lógica da rede. O detalhe da explicação teve contudo o objectivo de aumentar o conhecimento da rede nestas circunstâncias, e também de fornecer uma base de trabalho para outras avaliações do comportamento da rede.

Neste contexto, foi identificado um conjunto de eventos cuja probabilidade de ocorrência e seus potenciais efeitos nas comunicações, os tornam relevantes e susceptíveis de serem considerados na análise do comportamento da rede.

Foi ainda avaliado o comportamento dos mecanismos de recuperação, quando a sua operação é igualmente afectada pela ocorrência de faltas. Em consequência da forma como é implementado, o mecanismo de *timeout* apresenta um

desempenho bastante distante do desejável, verificando-se nestas condições tempos de recuperação de situações de perdas de *token* que podem assumir tempos significativamente elevados.

Um mecanismo que tem como primeira função a inserção de estações no anel lógico, mas que acaba também por gerir a inserção de estações que foram removidas de forma não intencional do anel é igualmente afectado pela ocorrência de faltas. Embora de forma não tão pronunciada como no caso anterior a alteração da dimensão da GAP ou da corrupção de tramas *Request FDL Status*, perturbam de forma considerável a inserção de estações no anel lógico.

Uma análise às fontes de perturbação da operação da rede, aconselham a uma configuração cuidada de certos parâmetros da rede de forma a minimizar as perturbações introduzidas no processo de recuperação de situações de excepção.

A segunda vertente da avaliação consistiu na verificação do impacto destes modos de operação nas comunicações da rede. Neste caso a avaliação incidiu nas comunicações quando a rede apresenta, quer uma configuração multi-mestre, quer apresente uma configuração mono-mestre pura. Indicadores de desempenho como tempo de resposta médio das mensagens, a média da resposta no pior caso, revelam uma maior propensão da configuração multi-mestre para sofrer uma degradação de desempenho. Este comportamento está em grande medida associado à estabilidade do anel lógico, que é afectada de forma significativa quer por eventos que interrompem o serviço do sistema – *perdas de token*, quer por eventos que interrompem o serviço da estação – *remoção não intencional de estações*. Neste mesmo cenário de operação a configuração mono-mestre, pela maior simplicidade das funções de acesso ao meio e gestão da rede, vê reduzida de forma muito significativa as fontes de instabilidade verificadas na configuração multi-mestre.

Como resultado desta avaliação pode-se inferir que a rede PROFIBUS-DP no modo multi-mestre é sensível a um conjunto de eventos susceptíveis de reduzir a confiança no funcionamento da rede. Esta redução não afecta os índices de segurança da rede que, prevê para este efeito a existência do PROFIsafe, mas antes nas questões que interferem com a disponibilidade da rede.

Capítulo 6

Conclusão

Verificação e previsão de faltas são importantes tarefas no processo de validação da operação de um sistema. Durante a fase de desenvolvimento permitem melhorar as características do sistema, assim como em fases posteriores, avaliar o seu comportamento nos mais diversos cenários de operação.

Existem duas classes de métodos cuja aplicação está em parte dependente do tipo de avaliação a efectuar, e da fase do ciclo de vida do sistema em que esta é efectuada. Esses métodos são de cariz analítico ou experimental.

Os primeiros fornecem ferramentas cujas descrições formais permitem reduzir o número de faltas introduzidas ao nível da concepção ou de implementação do sistema. Nesta classe existem também ferramentas poderosas capazes de efectuar a previsão de modos de operação tendo por base modelos que descrevem o sistema.

Os segundos requerem a existência do sistema ou de um seu protótipo sobre o qual são efectuados um conjunto de experiências conducentes à avaliação da sua operação.

As ferramentas baseadas em métodos analíticos são soluções economicamente atractivas e de fácil implementação quando comparadas com o esforço e investimento necessário a aplicar no desenvolvimento de um protótipo. Contudo, um dos componentes fundamentais destas ferramentas, - *o modelo do sistema*, carece ele próprio de uma validação de forma que a ferramenta esteja conforme para a função.

Em contraste os métodos experimentais sendo eles implementados sobre uma réplica do sistema fornecem valores de operação como os que ocorrem na realidade. Esta particularidade pode ser usada também na validação de modelos de ferramentas analíticas e na obtenção de parâmetros de funcionamento desses mesmos modelos.

Não obstante os significativos inconvenientes apresentados pelos métodos experimentais quando comparados com as soluções baseadas em ferramentas de simulação, estes são uma solução que não pode deixar de ser considerada. Antes pelo contrário, em situações especiais como são a validação de sistemas com requisitos de segurança (*safety*) críticos, apresentam-se como a principal ferramenta de validação e certificação de tais sistemas.

A injecção de faltas é uma das técnicas amplamente difundidas na validação e avaliação da confiança no funcionamento de sistemas. Não obstante a sua grande aceitação, e de fazer parte de uma área em que existe um elevado trabalho investigação e desenvolvimento, não existe nenhuma ferramenta standard cuja aplicação possa ser generalizada.

Isto resulta em parte da grande heterogeneidade dos problemas que se colocam no processo de validação, assim como, a grande diversidade tecnológica existente nos sistemas a validar. Neste contexto, as técnicas de injecção de faltas têm um processo evolutivo condicionado pelo objecto a validar e comparativamente muito pouca influência sobre a estrutura desse mesmo objecto. Este processo evolutivo reactivo está muito dependente dos desenvolvimentos tecnológicos e por conseguinte é um panorama de difícil alteração, nomeadamente ao nível das camadas base da técnica relacionadas com o processo de injecção (*injectores*).

Excluindo a área dos injectores, existe margem de evolução nos níveis superiores à camada física de injecção que é possível explorar, nomeadamente na integração e automação da informação e das metodologias de análise implementadas nas ferramentas.

Apesar do elevado esforço requerido no desenvolvimento das ferramentas de injecção de faltas, estas são ainda num primeiro estágio um dos instrumentos mais poderosos de obtenção de dados de operação dos sistemas.

Neste contexto, foi desenvolvida uma infra-estrutura para avaliação da operação do PROFIBUS-DP baseada em técnicas de injecção de faltas. A infra-estrutura tem uma arquitectura distribuída que lhe permite recolher informação dos vários agentes do sistema de comunicação.

A informação obtida das experiências de injecção de faltas é armazenada estabelecendo um repositório de informação que pode ser consultado em tempo diferido. Desta forma possibilita a execução de largo espectro de análises, num processo por vezes iterativo no qual a operação dos diversos agentes é correlacionada permitindo a identificação de modos operação particulares.

O PROFIBUS-DP é uma rede destinada à interligação de sistemas de controlo de tempo-real. Como tal a sua operação não deve ser um factor de redução da confiança no funcionamento global do sistema. Da mesma forma, para que a sua resposta possa ser correctamente considerada no projecto de aplicações de tempo-real, as suas características de operação nos mais diversos cenários devem estar bem caracterizadas.

Numa das vertentes da confiança no funcionamento – a *segurança*, o PROFIBUS-DP através do seu perfil *PROFIsafe* assegura níveis de segurança

SIL 3. Assim, este parâmetro da confiança no funcionamento está perfeitamente identificado, e compatível com requisitos de um largo espectro de aplicações no domínio da automação industrial.

Contudo, para assegurar elevados níveis de segurança é por vezes necessário implementar mecanismos que quando activados garantem a segurança do sistema, em detrimento de um certo grau da redução da disponibilidade do mesmo. Desta forma, em determinadas condições operacionais, quando o desempenho na presença de faltas da rede não permite cumprir os requisitos mínimos da operação do sistema esses mecanismos podem ter de ser activados. Outros cenários resultam directamente da falha da operação do sistema fruto do não cumprimento das restrições temporais da aplicação. Neste contexto, resulta uma diminuição da confiança no funcionamento derivada da redução da disponibilidade do sistema ou da não execução da função para que foi concebido.

Ao centrar a avaliação da rede na perspectiva do seu desempenho na presença de faltas, nomeadamente na identificação de eventos que produzem degradação significativa da sua operação, pretendeu-se fornecer um contributo para a caracterização da operação da rede em cenários de faltas.

A avaliação teve incidência particular nos efeitos das faltas sobre as tramas de gestão da rede e o seu impacto no desempenho da mesma, abrangendo as duas configurações típicas de operação da rede:

- O modo multi-mestre, no qual existe mais que uma estação activa a partilhar o acesso ao mesmo barramento;
- O modo mono-mestre, onde existe somente uma estação activa a operar no barramento numa operação puramente mestre escravo.

Na operação multi-mestre foram identificados seis eventos com impacto significativo no desempenho das estações ou da totalidade da rede. Desses eventos três resultam em perdas de *token* com a correspondente inibição temporária do estabelecimento de comunicações no sistema.

Acresce que a probabilidade de um *token* se perder durante o processo da sua passagem entre estações não é influenciado, quer pela carga da rede, quer pela configuração das faltas que perturbam a sua operação. Em contraste, a probabilidade de perda de *token* cresce exponencialmente com o aumento da taxa de erros, especialmente se a perturbação tiver um padrão com pelo menos uma replica num intervalo temporal da ordem de grandeza do parâmetro T_{SL}.

Este padrão de comportamento é indicativo da sensibilidade do PROFIBUS-DP a erros transitórios que tendem a gerar instabilidade na operação do anel lógico. Instabilidade essa que é agravada pelo facto do mecanismo que recupera destes eventos ser também ele afectado pela ocorrência de faltas, o que se traduz num aumento por vezes muito significativo dos tempos de recuperação.

Uma forma de mitigar os impactos destes eventos na operação da rede pode ser conseguida através do recurso a uma cuidada configuração da mesma. Na qual devem ser restringidos parâmetros que são importantes do ponto de vista da

flexibilidade da rede em facilmente comportar alterações na sua estrutura, como sejam a livre atribuição de endereços às estações, distância entre endereços e o número máximo de endereços utilizados no anel lógico (*HSA*).

Uma particular atenção deve ser depositada na atribuição dos endereços das estações activas, e principalmente à estação com o menor endereço da rede. Isto resulta do facto do valor do temporizador de *timeout* T_{TO}, que está implementado em cada estação e que regula a recuperação das perdas de *token*, depender do valor do seu endereço.

A escolha de um endereço elevado tem uma acção duplamente prejudicial para o desempenho da rede. Numa primeira instância e em condições normais aumenta o tempo de recuperação da perda de *token*. Numa segunda fase potencia os efeitos da má operação do mecanismo de recuperação da perda do *token* em cenários de faltas.

Apesar de não terem sido efectuados testes comparativos que permitam suportar quantitativamente esses efeitos, o padrão de comportamento da rede observado durante a avaliação revelou as potenciais implicações de uma configuração menos cuidada em situações de perturbação significativas.

Um exemplo extremo dessa má operação pode ser constatado para estações com endereços elevados, e para taxas de erro igualmente elevadas. Durante uma pequena experiência em que a estação com endereço mais baixo era o 51, e taxa de erros $1 \cdot 10^{-3}$ BER, foi observado que após ocorrer uma perda de *token* o mecanismo incumbido de recuperar desta situação é afectado de tal forma, que a rede não consegue já mais recuperar e efectuar qualquer envio de trama pelo barramento.

Outras alterações podem ser efectuadas nos parâmetros da FDL. Neste caso, o T_{SL} pela sua relação com o mecanismo de perda de token pode ser um bom candidato. Naturalmente que valores de T_{SL} mais elevados propiciam condições para a existência de mais replicas de faltas dentro do intervalo T_{SL} e maior sensibilidade a perdas de *token* pelo evento *Erro Durante o Slot Time*. Importa assim verificar qual o impacto da alteração deste parâmetro. Contudo, uma alteração do mesmo não pode ser dissociada do padrão de faltas que afectam a rede.

Tendo presente que na realidade estas faltas estão associadas a perturbações com um comportamento aleatório ao nível do número de réplicas e distância entre elas, uma redução ou variação de T_{SL} dentro dos parâmetros aceitáveis para a operação da rede não deverá modificar significativamente o padrão de ocorrência do evento *Interrupção do Serviço do Sistema*. Isto é principalmente válido para as taxas de erros mais elevadas. Esta projecção é suportada nas características dos três eventos que contribuem para a *Interrupção do Serviço do Sistema*. Em consequência de uma redução de T_{SL}, será previsível uma transferência do peso dos contributos do evento *Erro Durante o Slot Time* para o evento *Erro no Token* ou mesmo provocar um aumento do evento *Erro Fatal*.

Da perspectiva do desempenho na presença de faltas, existe um impacto muito positivo na escolha dos parâmetros da rede na limitação da degradação do seu desempenho. Contudo, esta ainda é significativa para as taxas de erros mais elevadas.

A latência média dos serviços de comunicação de dados e WCRT têm aumentos importantes devido à componente introduzida pela perda de *token* e atrasos na sua recuperação.

Em contraste a operação mono-mestre apresenta uma estabilidade da rede muito significativa, nomeadamente quando esta é comparada com o modo de operação multi-mestre. Contribui para o facto a redução das fontes de instabilidade, nomeadamente daquelas que conduzem à perda de *token*. Não obstante as boas características temporais que apresenta a rede na resposta aos serviços de comunicação de dados, à imagem do modo multi-mestre a rede apresenta perdas de *token* pelo evento *Erro Fatal*.

Este evento representa uma falha associada ao mau funcionamento do *transceiver* do nó de comunicações – *erro permanente*, mas que é incorrectamente identificado pelo protocolo em alguns cenários envolvendo erros transitórios. Não obstante a sua baixa probabilidade é um evento com um forte impacto na disponibilidade do sistema, uma vez que em consequência deste evento a FDL da estação evolui para o estado *Offline*, permanecendo a estação inoperacional até que seja posta de novo em serviço por acção de um operador.

Melhorias ao Funcionamento do PROFIBUS-DP

Em síntese o PROFIBUS-DP na presença de faltas apresenta alguns padrões de funcionamento inerentes à estrutura do protocolo, que podem revelar-se como factores negativos nestes cenários de operação. Neste contexto, e em função da observação do funcionamento da rede, a partir das experiências de injecção de faltas é possível apontar duas abordagens que visam minimizar os seus efeitos durante a utilização da rede. Uma das abordagens incide sobre parâmetros da configuração da rede, e desta forma podem ser aplicados pelo utilizador da rede. Na segunda abordagem, dada a profundidade da intervenção, esta só pode ser considerada como fazendo parte da incorporação de melhorias ao nível do ASIC's que implementam partes do protocolo.

A primeira abordagem aponta para escolha cuidada dos seguintes parâmetros numa configuração de rede multi-mestre:

- Atribuição do endereço 1 a uma das estações activas da rede. Esta estação será quem recupera das perdas de *token*, pelo que é aconselhável que seja seleccionada a estação que apresente requisitos mais críticos;
- Minimização da GAP. No projecto da rede sempre que possível não devem ser deixados endereços livres. Desta forma são reduzidos os tempos de inserção de estações da rede que tenham sido removidas do anel lógico de forma não intencional. Esta configuração deve ser

acompanhada da atribuição do HSA de acordo com a dimensão da rede projectada.

- Um cuidado no planeamento do padrão de mensagens da rede e do seu tempo de produção. Este planeamento deve ser acompanhado de um dimensionamento do parâmetro T_{TR} que assegure uma margem de tempo suficiente para acomodar as perturbações causadas pelas faltas, sem que resulte uma monopolização do barramento por parte de uma das estações. Este equilíbrio permite obter, para situações de faltas, muito bons desempenhos da rede nos tempos médios de resposta aos serviços comunicação. Estes tempos podem mesmo ser semelhantes aos apresentados pela configuração mono-mestre, como pode ser observado através da comparação das figuras 5.26 e 5.32 do capítulo 5.

A segunda abordagem passa idealmente pela incorporação das seguintes alterações:

- Reinicialização do mecanismo de *timeout* com base em eventos confiáveis. Neste contexto, o temporizador de *timeout* deveria deixar de ser reiniciado por qualquer actividade no barramento, mas antes pelo identificador (*Start Delimiter*) das tramas que são transmitidas no barramento de comunicações;
- O mecanismo de verificação de erros permanentes nos *transceivers* deverá ser implementado em hardware de forma a evitar falsas detecções de faltas permanentes, geradas por erros transitórios no *token*. Não obstante este erro ter uma probabilidade relativamente baixa, a sua ocorrência tem um impacto muito severo no desempenho do sistema quer este esteja em configuração multi-mestre quer esteja em configuração mono-mestre. Quando este evento ocorre a FDL evolui para um estado de não operacionalidade (*Offline*), com a consequente paragem do sistema que utiliza os seus serviços de comunicação.

Qualquer uma destas alterações não acarreta nenhuma alteração funcional ao protocolo, pelo que é garantida a compatibilidade com as soluções já implementadas.

Perspectivas de Trabalho Futuro

Actualmente assiste-se a um período de grande dinâmica na área das comunicações. No domínio de aplicação da automação industrial estão a emergir redes de comunicação que oferecem um conjunto de funcionalidade substancialmente mais poderosas que as disponibilizadas pelas soluções já existentes. De entre estas encontram-se vários perfis de rede que operam sobre *ethernet* (ex. EtherCat, EtherNET/IP, EPL, SERCUS III, Modbus/TCP e PROFINET [Felser04]). A utilização de redes baseadas em *ethernet* teve durante algum tempo algumas restrições à sua aplicação em aplicações de controlo. As restrições derivavam da impossibilidade de assegurar determinismo ao processo de comunicação. Tecnicamente ultrapassadas estas questões, a utilização destas

redes apresenta significativas vantagens relativamente às soluções já estabelecidas. Vantagens que derivam nomeadamente da possibilidade da utilização de tecnologia amplamente difundida, o que torna os custos de hardware substancialmente inferiores, e da disponibilização de serviços de comunicação com funcionalidades superiores acompanhadas de capacidade para suportar comunicação de tempo real.

Neste contexto, é previsível que estas redes venham a disputar e mesmo a ocupar uma posição de relevo nas aplicações de controlo de tempo real. Contudo, estas são também redes que suportam protocolos que tendencialmente são mais complexos, e que irão suportar aplicações com requisitos específicos de confiança no funcionamento. Estes requisitos caracterizam-se fundamentalmente segundo duas vertentes: segurança e disponibilidade.

Embora a questão da segurança seja por si só um aspecto fundamental em muitas aplicações de controlo, um considerável número de redes possui perfis que asseguram elevados níveis de integridade de segurança. Para tal, seguem uma abordagem na qual se assume que os serviços de comunicação são estabelecidos sobre um sistema de comunicação intrinsecamente inseguro, e providenciam numa camada superior um conjunto de mecanismos que asseguram a segurança das comunicações [Piggin00]. Acresce que, estes perfis são usualmente validados por organizações certificadoras, e assim, o grau de confiança no funcionamento que deriva da segurança está perfeitamente caracterizado.

A disponibilidade é também um importante aspecto a considerar para a definição da confiança no funcionamento. O tipo de aplicações que é comum encontrar em ambiente industrial, é caracterizado pelo seu elevado custo quer em termos de investimento, quer nos custos intrínsecos a matérias-primas e custos de exploração. Assim, a paragem de um equipamento traduz-se geralmente em avultados prejuízos, pelo que a disponibilidade dos equipamentos é muito importante.

Neste enquadramento, torna-se relevante avaliar condições de funcionamento das novas redes de comunicação, que objectivamente possam interferir na disponibilidade dos sistemas de controlo. Designadamente aquelas que resultam de perturbações típicas (ex. EMI) do ambiente industrial em que operam.

O tipo de funcionalidades disponibilizadas por estas redes traduz-se tendencialmente em acréscimos significativos da complexidade da sua operação. Desta forma, o processo de avaliação do funcionamento é também ele afectado por essa complexidade. Consequência disso torna-se pertinente a escolha da uma metodologia a aplicar à avaliação. Tipicamente existem duas abordagens que podem ser aplicadas a este tipo de sistemas: utilização de métodos analíticos, ou através do recurso a métodos experimentais.

A aplicação de métodos analíticos a sistemas complexos está essencialmente condicionada à capacidade de se obter um modelo que represente o seu funcionamento com o grau de detalhe necessário. Contudo, o desenvolvimento de um modelo detalhado é de difícil execução.

Os métodos experimentais permitem a realização de experiências sobre sistemas reais e obter directamente resultados da sua operação. Assim, estes métodos podem desempenhar um papel importante na avaliação do funcionamento destas redes.

Desta forma, o capital de conhecimento obtido durante a realização desta dissertação abre perspectivas interessantes de continuação do trabalho, agora aplicado à avaliação do desempenho destas novas redes na presença de faltas.

Bibliografia

[Agrawal94] G. Agrawal, B. Chen, W. Zhao, ***Guaranteeing Synchronous Message Deadlines with the Timed Token Medium Access Control Protocol***, Transactions on Computers, Vol.43, Nº 3, pp.327-339, IEEE, 1994.

[Alexopoulos04] C. Alexopoulos, D. Goldsman, ***To Batch Or Not To Batch?*** Transactions on Modeling and Computer Simulation, Vol. 14, Nº 1, pp. 76-114, ACM, 2004.

[Amendola03] A. M. Amendola, R. Di Maio, M. L. Iacobuzio, F. Poli, F. Scalabrini, ***Lessons Learned in Designing and Evaluating Railway Control Systems***, Proceedings of 9th International Workshop on Object-Oriented Real-Time Dependable Systems, pp. 355-358, IEEE, 2003.

[Antoni00] L. Antoni, R. Leveugle, B. Feher, ***Using Run-Time Reconfiguration for Fault Injection in Hardware Prototypes,*** Proceedings of International Symposium on Defect and Fault Tolerance in VLSI Systems, pp. 405–403, IEEE, 2000.

[Arlat03] J. Arlat, Y. Crouzet, J. Karlsson, P. Folkesson, E. Fuchs, G.H. Leber, ***Comparison of Physical and Software Implemented Fault Injection Techniques***, Transactions on Computers, Vol. 52, Issue 9, pp.1115 – 1133, IEEE, 2003.

[Arlat93] J. Arlat, A. Costes, Y. Crouzet, J. C. Laprie, D. Powell, ***Fault Injection and Dependability Evaluation of Fault-Tolerant Systems***, Transactions on Computers, Vol 42, Nº8, pp.913-923, IEEE, 1993.

[Arlat90] J. Arlat, M. Aguera, L. Amat, Y. Crouzet, ***Fault Injection for Dependability Validation: A Methodology and Some Applications***, Transactions on Software Engineering, Vol. 16, No. 2, pp.166-182, IEEE, 1990.

[Avizienis04] A. Avizienis, J. Laprie et all, ***Basic Concepts and Taxonomy of Dependable and Secure Computing***. Transactions on Dependable and Secure Computing, Vol 1, Nº 1, pp. 11-33, IEEE, 2004.

[Baback04] B. Rahbaran, A. Steininger, T. Handl, ***Built-in Fault Injection in Hardware -The FIDYCO Example***, Proceedings of Second International Workshop on Electronic Design, Test and Applications, pp. 327-332, IEEE, 2004.

[Banks96] J. Banks, J. Carson II, B. Nelson, ***Discrete-Event System Simulation***, Second Edition, Prentice Hall International Editions, 1996.

[Baraza05] J. C. Baraza, J. Garcia, J. Gil, ***Improvement of fault injection techniques based on VHDL code modification***, Proceedings of 10th International High-Level Design Validation and Test Workshop, pp.19-26, IEEE, 2005.

[Bello99] L. Lo Bello, O. Mirabella. ***A Fault Tolerance Analysis of PROFIBUS Systems by Means of Generalised Stochastic Petri Nets***, Proceedings of 25th Conference on Industrial Electronics Society - IECON99, pp.1210-1215, IEEE, 1999.

[Bondavalli99] A. Bondavalli, A. Fantechi, D. Latella, L. Simoncini, ***Integrated Validation of Dependable Systems***, Proceedings of 4th International Conference on Integrated Design and Process Technology – IDPT99, 1999.

[Buchner90] S. Buchner, K. Kang, W.J. Stapor, A.B. Campbell, A.R. Knudson, P. McDonald, S. Rivet, ***Pulsed Laser-Induced SEU in Integrated Circuits: A Practical Method for Hardness Assurance Testing***, Transactions on Nuclear Science, Vol. 37, Issue 6, Part 2, pp. 1825-1831, IEEE, 1990.

[Carreira98] J. Carreira, H. Madeira, J. G. Silva, ***Xception: a Technique for the Experimental Evaluation of Dependability in Modern Computers***, Transactions on Software Engineering, Volume 24, Issue 2, pp.125 – 136, IEEE, 1998.

[Carvalho05a] J. A. Carvalho, A. S. Carvalho, P. J. Portugal, ***Assessment of PROFIBUS Networks Using a Fault Injection Framework***, Proceedings of 10th Conference on Emerging Technologies and Factory Automation, ETFA 2005, Vol. 1, pp. 415-423, IEEE, 2005.

[Carvalho05b] J. A. Carvalho, A. S. Carvalho, P. J. Portugal, ***Experimental Analysis of Outage Times for PROFIBUS Networks***, Proceedings of 32nd Conference on Industrial Electronics Society, IECON 2005, pp. 421-426, IEEE, 2005.

[Carvalho03] J. A. Carvalho, P. J. Portugal, A. S. Carvalho, ***A Framework for Dependability Evaluation of PROFIBUS Networks***, Proceedings of the International Symposium on Industrial Electronics, IEEE, 2003.

[Cavalieri02] S. Cavalieri, S. Monforte, E. Tovar, F. Vasques, ***Evaluating worst case response time in mono and multi-master PROFIBUS-DP***, Proceedings of the 4th International Workshop on Factory Communication Systems, pp. 233-240, IEEE, 2002.

[Cavalieri97] S. Cavalieri, A. Di Stefano, O. Mirabella, ***Impact of Fieldbus on Communication in Robotic System***, Transactions on Robotics and Automation, Vol. 13, No. 1, IEEE, 1997.

[Cavalieri95] S. Cavalieri, A. Di Stefano, O. Mirabella, ***Centralized versus Distributed Protocols for FieldBus Applications***, Proceedings of 21st International Conference on Industrial Electronics, Control, and Instrumentation - IECON, pp.1580-1585, IEEE, 1995.

[Chen92] B. Chen, G. Agrawal, W. Zhao, ***Optimal Synchronous Capacity Allocation for Hard Real-Time Communications with the Time Token Protocol***, Symposium on Real-Time Systems, pp.198-207, IEEE,1992.

[Choi93] G. S. Choi, R. K. Iyer, D. G. Saab, ***Fault Behavior Dictionary for Simulation of Device-Level Transients***, International Conference on Computer-Aided Design, pp. 6-9, IEEE/ACM, 1993.

[Civera01] P. Civera, L. Macchiarulo, M. Rebaudengo, M. S. Reorda, M Violante, ***FPGA-based Fault Injection for Microprocessor Systems,*** Proceedings of 10th Asian Test Symposium, pp. 304-309, IEEE, 2001.

[Clark95] J. A. Clark, D. K. Pradhan, ***Fault Injection: A Method for Validating Computer-System Dependability***, Computer, pp. 47-56, IEEE, 1995.

[CSIM] Mesquite Software Inc. ***http://www.mesquite.com/products /CSIM19_datasheet.pdf***.

[Cucej04] Z. Cucej, D. Gleich, M. Kaiser, ***Industrial Networks***, Proceedings of 46th International Symposium Electronics in Marine, pp. 59-66, Croatia, 2004.

[Cukier99] M. Cukier, D. Powell, J. Arlat, ***Coverage Estimation Methods for Stratified Fault Injection***, Transactions on Computers, Vol. 48, Nº7, pp. 707-723, IEEE, 1999.

[Daigle88] J. N. Daigle, A. Seiddmann, J. R. Pimentel, ***Communications for Manufacturing: An Overview***, IEEE Network, Vol.2, Nº 3, pp. 6-13, IEEE, 1988.

[Decotignie05] J. D. Decotignie, ***Which Network for Which Application***, The Industrial Information Technology Handbook, pp46.1 – 46.15, CRC Press 2005.

[Decotignie93] J. P. Decotignie, P. Pleinevaux, ***A Survey on Industrial Communications Networks***, in Annales des Télécommunications, vol. 48, No.9-10, pp. 435–448, 1993.

[Dostie95] B. Dostie, H. Hulvershorn, S. Adham, ***A New Hardware Fault Insertion Scheme For System Diagnostics Verification***, Proceedings of international Test Conference, pp. 994-1002, IEEE, 1995.

[DSTni-LX03] ***DSTni-LX Data Book***, Revision E, Grid Connect, 2003.

[EN02] EN 50325-4: ***Industrial communication subsystem based on ISO 11898 (CAN) for controller-device interfaces. Smart distributed systems* (*SDS*)**, CENELEC, 2002.

[EN99a] EN 50170/A2: ***General purpose field communication system***, CENELEC, 1999.

[EN99b] EN 50295: ***Low-voltage switchgear and controlgear. Controller and device interface systems. Actuator sensor interface* (*AS-i*)**, CENELEC, 1999.

[EN98] EN 50254: ***High Efficiency Communication Subsystem for Small Data Packages,*** CENELEC, 1998.

[EN96a] EN 50170, ***General purpose field communication system*****,** Volume 2/3 (PROFIBUS), CENELEC, 1996.

[EN96b] EN 50170, ***General purpose field communication system Part 8-2 – User Specifications,*** Volume 2/3, CENELEC, 1996.

[Fabre00] J.-C. Fabre, M. Rodriguez, J. Arlat, J.-M. Sizun, ***Building Dependable COTS Microkernel-based Systems using MAFALDA***, Proceedings of Pacific International Symposium on Dependable Computing, pp.85-92, IEEE, 2000.

[Felser04] M. Felser, T. Sauter, ***Standardization of Industrial Ethernet – the Next Battlefield***, Proceedings of International Workshop on Factory Communication Systems, pp.413-421, IEEE, 2004.

[Felser02] M. Felser, T. Sauter, ***The Fieldbus War: History or Short Breaks Between Battles?*** Proceedings of 4th International Workshop on Factory Communication Systems, pp.73-80, IEEE, 2002.

[Folkesson99] P. Folkesson, **Assessement** ***and Comparation of Physical Fault Injection Techeniques***, PHD thesis, Chalmers University of Techenology, Göteborg Sweden 1999.

[Folkesson98] Folkesson P, S. Svensson, J. Karlsson, ***A Comparison of Simulation Based and Scan Chain Implemented Fault Injection***, Proceedings of 28th International Symposium on Fault Tolerant Computing, pp.284-293, IEEE, 1998.

[Gaisler02] J. Gaisler, ***A Portable and Fault Tolerant Microprocessor Based on the SPARC V8 Architecture***, Proceedings of the International Conference on Dependable Systems and Networks, pp 409-415, IEEE, 2002.

[Gaisler97] J. Gaisler, ***Evaluation of a 32-bit microprocessor with built-in concurrent error-detection***, Proceedings of 27th International Symposium on Fault-Tolerant Computing, pp 42-46, IEEE, 1997.

[Garcia02] J. Garcia, D. Gil, J. C. Baraza, ***Using VHDL-based fault injection to exercise error detection mechanisms in the time-triggered architecture***. Proceedings of the International Symposium on Dependable Computing, pp. 316-320, IEEE, 2002.

[Goswami97] K. Goswami, K. Iyer, ***A simulation-Based Environment for System Level Dependability***. Transactions on Computers, Vol. 46, Nº1, pp.60-74, IEEE,1997.

[Groover00] M. Groover, ***Automation Production Systems, and Computer-Integrated Manufacturing***, Prentice Hall, 2000.

[Gunneflo89] U. Gunneflo, J. Karlsson, J. Torin, ***Evaluation of Error Detection Schemes Using Fault Injection by Heavy-ion Radiation***, Proceedings of 27th International Symposium on Fault-Tolerant Computing, pp. 340-347, IEEE, 1989.

[Gunneflo87] U. Gunneflo, J. Karlsson, J. Torin, ***A Fault Injection System for the Study of Transient Fault Effects on Computer Systems***, Technical Report No. 47, Department of Computer Engineer, Chalmers University of Technology, Sweden, 1987.

[Güthoff96] J. Güthoff, V. Sieh, ***Combining Software-Implemented and Simulation-Based Fault Injection into a Single Fault Injection Method***, Proceedings of Twenty-Fifth International Symposium on Fault-Tolerant Computing, pp. 196-206, IEEE, 1996.

[Han95] S. Han; K. G. Shin, H. A. Rosenberg, ***DOCTOR: An Integrated Software Fault Injection Environment for Distributed Real-time Systems***, Proceedings of International Symposium on Computer Performance and Dependability, pp. 204-213, IEEE, 1995.

[Haverkort96] B. Haverkort, R. Marie, G. Rubino, K. Trivedi, ***Performability Modelling Tools and Techniques***, Performance Evaluation, Vol. 25, pp. 17-40, 1996.

[Hedberg01] J. Hedberg, Y. Wang, ***Methods for Verification and Validation of Distributed Control Systems***, PALBUS Work Package 10.10, V.4.0, 2001.

[Heffernan97] D. Heffernan, ***A technical Overview of Fieldbus Developments from Origins to Present Day Standards***, Technical Report, Department of Electronic and Computer Engineering, University of Limerick, 1997.

[Hsueh97] M-C. Hsueh, T. K. Tsai, R. K. Iyer. ***Fault Injection Techniques and Tools, Computer***, Vol. 19, Issue 4, pp.75-82, IEEE, 1997.

[IEC04] IEC-61508 ***Functional Safety of Electrical/Electronic /Programmable Electronic Safety Related Systems***, IEC, 2004.

[IEC03a] IEC61158, ***Digital Data Communications for Meassurement and Control – Fieldbus for use in Industrial Control Systems***, IEC, 2003.

[IEC03b] IEC61784-1, ***Digital Data Communications for Measurement and Control – Profiles Sets for Continuous and Discrete Manufacturing Relative to Fieldbus use in Control Systems***, IEC, 2003.

[IEC88] IEC 801-4 – ***Electrical Fast Transient/Burst (EFT)***, IEC, 1988.

[IEEE01] IEEE Std 1149.1, ***IEEE Standard Test Access Port and Boundary-Scan Architecture***, IEEE, 2001.

[ISO99] ISO 13849 – ***Safety of Machinery – Safety-Related Parts of Control Systems***, ISO, 1999.

[ISO94] ISO/IEC 7498-1: ***Information Technology – Open Systems Interconnection- Basic Reference Model***, ISO, 1994.

[Jecht05] U. Jecht, W. Stripf, P. Wenzel, *PROFIBUS* – ***Open Solutions for the World Automation***, The Industrial Information Technology Handbook, pp39.1 – 39.23, CRC Press 2005.

[Jenn94] E. Jenn, J. Arlat, J. Rimen, ***Fault Injection into VHDL Models: The MEFISTO Tool***, Proceedings of 24th International Symposium on Fault- Tolerant computing, pp. 66-75, IEEE, 1994.

[Johnson88] A. Johnson, M. Malek, ***Survey of Software Tools for Evaluating Reliability, Availability, and serviceability***, ACM Computing Surveys, Vol. 20, Nº 4, pp.227-269, ACM 1988.

[Joseph03] Ng. Joseph, S. Chan, T. Erdner, ***A Time Analysis on the Maximum Delay Bound in a Field Bus System Priority***, Technical Report, Department of Computer Science, Hong Kong Baptist University, 2003.

[Juan93] H.-P. Juan, N. D. Holmes, S. Bakshi, ***Top-down Modelling of RISC Processors in VHDL***, Proceedings of Design Automation Conference EURO-DAC'93, pp. 454-459, IEEE, 1993.

[Kalbarczyk93] Z. Kalbarczyk, J. Christmansson, H. Edler, ***Design Principles for Software Fault Tolerance – A Survey***, Technical Report nº 149, 1993.

[Kanawati95] N. A. Kanawati, G. A. Kanawati, J. A. Abraham, ***Dependability Evaluation Using Hybrid Fault/Error Injection***, Proceedings of International Computer Performance and Dependability Symposium, pp. 224 – 233, IEEE, 1995.

[Kanawati92] G. A. Kanawati, N. A. Kanawati, J. A. Abraham, ***FERRARI: a Tool for the Validation of System Dependability Properties***, Proceedings of Twenty-Second International Symposium on Fault-Tolerant Computing, pp. 336 – 344 IEEE, 1992.

[Kao94] W.-I. Kao, R.K. Iyer, ***DEFINE: A Distributed Fault Injection and Monitoring Environment***, Proceedings IEEE Workshop on Fault-Tolerant Parallel and Distributed Systems, pp.252-259, IEEE, 1994.

[Kao93] W.-I. Kao, R.K. Iyer, D. Tang, **FINE:** ***A Fault Injection and Monitoring Environment for Tracing the UNIX System Behavior under Faults***, Transactions on Software Engineering, Volume 19, Issue 11, pp. 1105 – 1118, IEEE, 1993.

[Karlsson95] J. Karlsson, J. Arlat, G. Leber, ***Application of Three Physical Fault Injection Techniques to the Experimental Assessment of the MARS Architecture***, Proceedings 5th IFIP Working Conference on Dependable Computing for Critical Applications, pp. 267-287, IEEE, 1995.

[Karlsson94] J. Karlsson, P Lidén, P. Dahlgren, R. Johansson, ***Using Heavy-Ion Radiation to Validate Fault-Handling Mechanisms***, IEEE Micro, pp. 8-23, IEEE, 1994.

[Ke96] W. Ke, ***Hybrid Pin Control Using Boundary-Scan and its Applications***, Proceedings of 5th Asian Test Symposium, pp. 44-49, IEEE, 1996.

[Kieckhafer88] R. M. Kieckhafer, C. J. Walter, A. M. Finn, ***The MAFT Architecture for Distributed Fault Tolerance***, Transactions on Computers, Vol. 37, Issue 4, pp. 398-404, IEEE, 1988.

[Kim94] H. Kim, K. G. Shin, ***On the Maximum Feedback Delay in a Linear/Nonlinear Control System With Input Disturbances Caused by Controller-Computer Failures***, Transactions on Control Systems Technology, Vol. 2, No. 2, pp.110-122, IEEE, 1994.

[Koopman04] P. Koopman, T. Chakravarty, ***Cyclic Redundancy Code (CRC) Polynomial Selection for Embedded Networks***, Proceedings of International Conference on Dependable Systems and Networks, pp.145-154, IEEE, 2004.

[Kopetz98] H. Kopetz, ***Real-Time Systems: Design Principles for Distributed Embedded Applications***, Kluwer Academic Publishers 1998.

[LaBel96] K. A. LaBel, M. M. Gates, P. Marshal, ***SEECA: Single Event Effect Criticality Analysis***, http ://radhome.gsfc.nasa.gov/radhome/papers/seeca1.htm, Nasa, 1996.

[Laprie89] J. C. Laprie. ***Dependability: a unifying concept for reliable computing and fault tolerance****, in Dependable of Resilent Computers*. T. Anderson BSP, 1989.

[Lee90] P. A. Lee, T. Andreson, ***Fault Tolerance: Principles and Practice***, Dependable Computing and Fault-Tolerant Systems Vol3, Springer-Verlag, Second revised edition, 1990.

[Leviti01] P. Leviti, ***IEC61158: An Offence to Technicians***, IFAC International Conference on Fieldbus Systems and Their Applications, pp.36, FeT07, 2001.

[Madeira94] H. Madeira, M. Ruela, F. Moreira and J. G. Silva, ***RIFLE: General Purpose Pin-level Fault Injector***, Proceedings of 1st European Dependable Computing Conference (EDCC-1), Berlin, Germany, pp 199-216, Spring-Verlag, 1994.

[Martínez99] R. J. Martinez, P. J. Gil, et al, ***Experimental Validation of High-Speed Fault Tolerant Systems Using Physical Fault Injection***, Proceedings of Dependable Computing for Critical Applications, pp.249-265, IEEE, 1999.

[Mayer95] J. F. Mayer, ***Performability Evaluation: Where It Is What Lies Ahead***, Proceedings of International Computer Performance and Dependability Symposium, pp.334-343, IEEE, 1995.

[Mayer89] J. F. Mayer, K. H. Muralidhar, W. H. Sanders, ***Performability of a Token Bus Network under Transient Fault Conditions***, pp. 175-182, IEEE, 1989.

[MIL91] MIL-HDK-217F, ***Reliability Prediction of Electronic Equipment*****,** Department of Defense, USA, 1991.

[Monforte00] S. Monforte, M. Alves, F. Vasques, E. Tovar, ***Designing Real-Time Systems Based on Mono Master Profibus-DP***, Proceedings of the 16th IFAC Workshop on Distributed Computer Control Systems, pp. 36-43, 2000.

[Moon98] H. Moon, H. S. Park, A. C. Ahn, W. H. Kwon, ***Performance degradation of the IEEE 802.4 token bus network in a noisy environment***, Computer Communications 21, pp. 547-557, Elsevier, 1998.

[Moss95] S.C. Moss, S.D. LaLumondiere, J. R. Scarpulla, K.P. MacWilliams, W.R. Crain, R. Koga**, *Correlation of Picosecond Laser-Induced Latchup and Energetic Particle-Induced Latchup in CMOS Test Structures***, Transactions on Nuclear Science, Vol. 42, Issue 6, Part 1, pp. 1948 – 1956, IEEE, 1995.

[National05] ***SCANSTA101 Low Voltage IEEE 1149.1 STA Master***, National Semiconductor Corporation, 2005.

[Parrotta00] B. Parrotta, M. Rebaudengo, M. S.Reorda, ***New Techniques for Accelerating Fault Injection in VHDL Descriptions***, Proceeding of 6th International On-Line Workshop, pp.61-66, IEEE, 2000.

[Piggin00] R. Piggin, ***A fieldbus for machine safety***, IEE Review, pp33-37, IEE, 2000.

[Pimentel90] Juan R. Pimentel, ***Communication Networks for Manufacturing***, Prentice-Hall International Editions, 1990.

[Pleinevaux88] P. Pleinevaux, J. D. Decotignie, ***Time Critical Comunication Networks: Field Buses***, IEEE Network, Vol. 2, No. 3, pp. 55-63, IEEE, 1998.

[Popp03] Manfred Popp, ***The New Rapid Way to PROFIBUS-DP – From DP-V0 to DP-V2***, PROFIBUS 2003.

[Portugal05] P. J. Portugal, ***Avaliação da Confiança no Funcionamento de Redes de Campo – Contribuição no Domínio dos Sistemas Industriais de Controlo***, Tese de Doutoramento, Faculdade de Engenharia da U.P. 2005.

[Portugal02] P. J. Portugal, J. A. Carvalho, A. Carvalho, ***Dependability Modelling Techniques For Electronics Systems***. Proceedings of Sixth European Space Power Conference, pp335-342, 2002.

[Powell95] D. Powell, E. Martins, J. Arlat, Y. Crouzet, ***Estimators for Fault Tolerance Coverage Evaluation***, Transactions on Computers, Vol. 44, Nº2, pp. 261-274, IEEE, 1995.

[Rembold93] U. Rembold, B. O. Nnaji, A. Storr, ***Computer Integrated Manufacturing and Engineering***, Prentice Hall 1993.

[Renovell95] M. Renovell, P. Huc, Y. Bertrand, ***Serial Transistor Network Modelling for Bridging Fault Simulation***, Proceedings of the Fourth Asian Test Symposium, pp. 100-106, IEEE, 1995.

[Ries94] G. L. Ries, G. S. Choi, R. K Iyer. ***Device-Level Transient Fault Modelling***, Proceedings of 24th International Symposium on Fault-Tolerant Computing, pp. 86-94, IEEE, 1994.

[Rodriguez02] M. Rodriguez, A. Albinet, J. Arlat, ***MAFALDA-RT: a Tool for Dependability Assessment of Real-Time Systems***, Proceedings of International Conference on Dependable Systems and Networks, pp. 267-272, IEEE, 2002.

[Rosenberg93] H. A. Rosenberg, K. G. Shin, ***Software Fault Injection and its Application in Distributed Systems***, Proceedings of 23th International Symposium on Fault-Tolerant Computing, pp. 208-217, IEEE, 1993.

[Samson98] J.R. Samson, W. Moreno, F. Falquez, ***A Technique for Automated Validation of Fault Tolerant Designs Using Laser Fault Injection (LFI)***, Proceedings of 28th International Symposium on Fault-Tolerant Computing, pp. 162–167, IEEE,1998.

[Schutz88] H. A. Schutz, The Role of MAP in Factory Integration, Transactions on Industrial Electronics, Vol. 35, Nº I, IEEE, 1988.

[Segall88] Z. Segall, D. Vrsalovic, D. Siewiorek, D. Yaskin, J. Kownacki, ***FIAT- Fault Injection Based Automation Testing Environment***, Proceedings of 18th International Symposium on Fault-Tolerant Computing, pp.102-107, IEEE,1988.

[Shin95] K. G. Shin, X. Cui, ***Computing Time Delay and Its Effects on Real-Time Control Systems***, Transactions on Control Systems Technology, Vol.3, No. 2, pp.218-224, IEEE, 1995.

[Siemens05] **Simatic Net:** ***ASPC2 Hardware User Description - Advanced PROFIBUS Controller According to IEC61158***, Version: V2.2, Siemens 2005.

[Statgraphics] ***Statgraphics Centurion, version XV***, http://www.statgraphics.com.

[Steininger02] A. Steininger, C. Scherrer, ***Identifying Efficient Combination of Error Detection Mechanisms Based on Results of Fault Injection***, Transaction on Computers, Vol. 5, Nº 2, IEEE, 2002.

[Steininger97] A. Steininger A, C. Scherrer, ***On finding an optimal combination of error detection mechanisms based on results of fault injection experiments***, Proceedings of Twenty-Seventh Annual International Symposium on Fault-Tolerant Computing, pp238 – 247, IEEE, 1997.

[Stott00] D.T. Stott, B. Floering, D. Burke, Z. Kalbarczpk, R.K. Iyer, ***NFTAPE: a framework for assessing dependability in distributed systems with lightweight fault injectors***, Proceedings of International Symposium on Computer Performance and Dependability, pp. 27-29, IEEE, 2000.

[Stripf05] W. Stripf, H. Barthel, ***PROFIsafe – safety Technology with PROFIBUS***, The Industrial Information Technology Handbook, pp56.1 – 56.20, CRC Press 2005.

[STSARCES00] P. Gil, L. Badiola, ***Safety Validation of Complex Components***, Annex 9 of Final Report of WP3.2, European Project STASRCES-Standard for Safety Related Complex Electronic Systems.

[Thomesse05] J. P. Thomesse, ***Fieldbus Technology in Industrial Automation***, Proceedings of the IEEE, Vol. 93, No. 6, pp.1073-1101, IEEE, 2005.

[Tovar99a] Eduardo Tovar, ***Supporting Real-Time Communications with Standard Factory-Floor Networks***, PHD thesis, Faculdade de Engenharia da U.P. 1999.

[Tovar99b] E. Tovar, F. Vasques, ***Real-Time Communications Using PROFIBUS Networks***, Transactions on Industrial Electronics, Vol.46, Nº6, pp.1241-1251, IEEE ,1999.

[Tovar99c] E. Tovar, F. Vasques, ***Cycle Time Properties of PROFIBUS Timed Token Protocol,*** Computer Communications, Vol.22, pp.1206-1216, 1999.

[Tovar99d] E. Tovar, F. Vasques, ***Analysis of the Worst-Case Real Token Rotation Time in PROFIBUS Networks***, Proceedings of FieldBus Technology (FET'99), pp. 359-366, 1999.

[Tovar98a] E. Tovar, F. Vasques, ***Guaranteeing Real-Time Message Deadlinein PROFIBUS Networks***, Proceedings of the 10th Euromicro Workshop on Real-Time Systems, pp. 79-86, IEEE, 1998.

[Tovar98b] E. Tovar, F. Vasques, ***Setting Target Rotation Time in PROFIBUS Based Real-Time Applications***, Proceedings of the 15th IFAC Workshop on Distributed Computer Control Systems, 1998.

[Trivedi93] K. Trivedi, M. Malhotra, ***Reliability and Performability Technique and Tools: A Survey***, Proceedings of the 7th ITG/GI Conference on Measurement, Modelling and evaluation of Computer and Communication Systems, 1993.

[Tsai96] T.K. Tsai, R.K. Iyer, D. Jewitt, ***An Approach Towards Benchmarking of Fault-Tolerant Commercial Systems***, Proceedings of Annual Symposium on Fault Tolerant Computing, pp. 314-323, IEEE, 1996.

[TÜV05] PROFIBUS Specifications ***PROFIsafe- Profiles for Failsafe Technology, Evaluation Report***, TÜV, 2005.

[Tyszer99] J. Tyszer, ***Object-Oriented Computer Simulation of Discrete-Event Systems***, Kluwer Academic Publishers, 1999.

[Veríssimo97] P. Veríssimo, J. Rufino, L. Ming, ***How hard is hard real-time communication on field-buses?***, Proceedings of the 27th International Symposium on Fault-Tolerant Computing, pp.112-121, IEEE, 1997.

[Veríssimo89] P. Veríssimo, R. Lemos, ***Confiança no Funcionamento: Proposta para uma Terminologia em Português***, INESC Technical Report (RT748-89), 1989.

[Vitturi04] S. Vitturi, ***On the effects of the acyclic traffic on PROFIBUS-DP networks***, Computer Standards & Interfaces, Vol. 26pp.131-144, Elsevier, 2004.

[Walter90] C. J. Walter, ***Evaluation and design of an ultra-reliable distributed architecture for fault tolerance***, Transactions on Reliability, Vol. 39, Issue 4, pp. 402-499, IEEE, 1990.

[Wells98] R. B. Wells, ***Coding and Information Theory for Engineers***, Prentice Hall, 1998.

[Willig02] A. Willig, ***Analysis of the PROFIBUS Token Passing Protocol over Wireless Links***, Proceedings of 25th International Symposium on Industrial Electronics – ISIE02, pp. 56-60, Vol. 1, IEEE, 2002.

[Willig01] A. Willig, A. Wolisz, ***Ring Stability of the PROFIBUS Token Passing Protocol over Prone Links***, Transactions on Industrial Electronics, Vol. 48, Issue 5, pp. 1025-1033, IEEE, 2001.

[Willig99a] A. Willig, ***Analysis of the PROFIBUS Token Passing Protocol over Error Prone Links***, TKN Technical Report TKN-99-001, Technical University Berlin, Telecommunication Network Group, 1999.

[Willig99b] A. Willig, ***Analysis of the PROFIBUS Token Passing Protocol over Error Prone Links***, Proceedings of 25th Conference on Industrial Electronics Society - IECON99, pp. 1246-1252, IEEE, 1999.

[Willig99c] A. Willig, ***Markov Modeling of PROFIBUS Ring Membership over Error Prone Links***, TKN Technical Report TKN-99-004, Technical University Berlin, Telecommunication Network Group, 1999.

[Yang92] F. Yang. ***Simulation of Faults Causing Analog Behavior in Digital Circuits***, Ph.D. Thesis, University of Illinois, 1992.

[Yook02] J. K. Yook, D. M. Tilbury, ***Trading Computation for Bandwidth: Reducing Communication in Distributed Control System Using State Estimators***, Transactions on Control Systems Technology, Vol. 10, No. 4, pp. 503-518, IEEE, 2002.

[Yook00] J. K. Yook, D. M. Tilbury, ***A Design Methodology for Distributed Control Systems to Optimize Performance in the Presence of Time Delays***, Proceedings of American Control Conference, pp. 1959-1964, IEEE, 2000.

[Zarandi03] H. R. Zarandi, S. G. Miremadi, A. Ejlali, ***Dependability analysis using a fault injection tool based on synthesizability of HDL models***, Proceedings of 18th International Symposium on Defect and Fault Tolerance in VLSI Systems, pp. 485-492, IEEE, 2003.

[Zhang99] S. Zhang, ***Cycle-Time Properties of the Token Medium Access Control Protocol***, Technical Report, Centre for communication Systems Research, University of Cambridge, 1999.

[Zurawski05] R. Zurawski, ***Scanning the Issue: Special Issue on Industrial Communication Systems***, Proceedings of the IEEE, Vol. 93, No. 6, pp.1067-1072, IEEE, 2005.