

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais -
PPGEM

Estudo da Utilização do Protocolo PROFIBUS para Aplicações
com Grande Número de Estações Escravas

por
Miriam Noemi Cáceres Villamayor

Dissertação para obtenção do título de Mestre
em Engenharia

Porto Alegre
1999

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais -
PPGEM

Estudo da Utilização do Protocolo PROFIBUS para Aplicações
com Grande Número de Estações Escravas

por
Miriam Noemi Cáceres Villamayor
Engenheira Eletricista,

Trabalho realizado no Departamento de Engenharia Elétrica da Escola de Engenharia da UFRGS, dentro do Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais – PPGEM, como parte dos requisitos para a obtenção do título de Mestre em Engenharia

Área de Concentração: Instrumentação Eletro-Eletrônica

Porto Alegre
1999

Esta DISSERTAÇÃO foi julgada adequada para a obtenção do título de Mestre em Engenharia, área de concentração Instrumentação Eletro-Eletrônica e aprovada em sua forma final, pelo Orientador e pela Banca Examinadora do Curso de Pós-Graduação.

Orientador: _____

Altamiro Amadeu Susin

Doutor em Informática

Professor do Programa de Pós-Graduação em Engenharia

Metalúrgica e dos Materiais da UFRGS

Banca examinadora:

Prof. Dr. Carlos Eduardo Pereira

Prof. Dr. Marcelo Soares Lubaslewski

Prof. Dr. João Cesar Netto

Prof. Dr. Telmo Roberto Strohaecker

Coordenador do PPGEMM

AGRADECIMENTOS

Um dos requisitos exigidos para obtenção do grau de mestre é o de apresentar o trabalho de dissertação. Chegar a este ponto implica em horas de dedicação e muito trabalho, horas de desespero que às vezes parece que nunca irão terminar.

Terminar esta dissertação foi para mi um desafio que dificilmente teria sido conseguido sem a confiança, o incentivo e a ajuda de muitos amigos que me deram apoio para não desistir e finalmente poder chegar a este momento por demais importante na minha vida.

Vou citar alguns destes amigos, más ficam no meu coração todos aqueles que de alguma forma contribuíram com este trabalho.

Ao Engenheiro Ronaldo que me acompanhou muitas noites e feriados adaptando o programa dos votadores para os testes, gravando EPROM's e inclusive lixando e pintando a estante para colocar os votadores, que por sinal ficaram muito bonitos nas fotos.

Ao Engenheiro Cícero Renato Lorenzi, companheiro em todas as hora, sempre presente com seu apoio e estímulo.

Ao Professor Engenheiro Jorge Ferreira a quem recorri quando meu orientador se ausentou para fazer um curso no Canadá, e de quem não apenas recebi ajuda técnica como ele junto com a Nelci, sua esposa, muito me incentivaram acreditando em minha capacidade para vencer este desafio.

Ao Professor Dr. Brito que, também na ausência do meu orientador, lia e, para a minha felicidade, elogiava os capítulos que aos poucos foram se transformando nesta dissertação.

Ao Professor Dr. Altamiro Amadeu Susin, meu orientador, possuidor de uma paciência a toda prova, quase comparável a fé, capaz de suportar uma e outra vez a

quebra dos meus cronogramas. Com certeza ele é o principal responsável de eu não ter desistido.

Ao CNPq pelo apoio financeiro através da bolsa de mestrado. E a todos aqueles que de um modo ou outro colaboram para o surgimento e manutenção de locais de pesquisa e ensino no Brasil e no exterior.

SUMÁRIO

1. INTRODUÇÃO	16
2. REDES DE COMPUTADORES	19
2.1 INTRODUÇÃO HISTÓRICA	19
2.2 CLASSIFICAÇÃO.....	22
2.2.1 <i>Redes Locais - LAN's</i>	23
2.2.2 <i>Campus Area Networks - CAN's</i>	23
2.2.3 <i>Redes de Área Metropolitana - MAN's</i>	23
2.2.4 <i>Redes de Longa Distância - WAN's</i>	24
3. MODELO OSI/ISO E PROTOCOLOS	25
3.1 INTRODUÇÃO.....	25
3.2 CONCEITO	26
3.3 DESCRIÇÃO	28
3.3.1 <i>Camada Física</i>	28
3.3.2 <i>Camada de Enlace de Dados</i>	29
3.3.3 <i>Camada de Rede</i>	30
3.3.4 <i>Camada de Transporte</i>	32
3.3.5 <i>Camada de Sessão</i>	33
3.3.6 <i>Camada de Apresentação</i>	33
3.3.7 <i>Camada de Aplicação</i>	34
4. EVOLUÇÃO HISTÓRICA DAS REDES	35
4.1 INTRODUÇÃO.....	35
4.2 REDES PÚBLICAS.....	35
4.3 A ARPANET	38
4.4 MAP E TOP	39
4.4.1 <i>Mini - MAP</i>	41
4.5 A USENET	42
4.6 CSNET	43
4.7 BITNET.....	44
4.8 SNA	45
5. PROJETO INDUSTRIAL	46
5.1 INTRODUÇÃO.....	46
5.1.1 <i>Processos Discretos</i>	48
5.2 CIM - FABRICAÇÃO INTEGRADA POR COMPUTADOR.....	50
5.2.1 <i>Nível de campo</i>	53
5.2.2 <i>Nível de célula</i>	53
5.2.3 <i>Nível de controle de processo</i>	54
5.2.4 <i>Nível de Gerenciamento</i>	55
5.2.5 <i>Nível de Planejamento</i>	55
6. O PROTOCOLO PROFIBUS	58
6.1 PROPRIEDADES BÁSICAS	58
6.2 ARQUITETURA DO PROTOCOLO PROFIBUS	59
6.3 A CAMADA FÍSICA (1).....	61
6.4 A CAMADA DE ENLACE DE DADOS (2)	63

6.4.1	<i>Transmissão de mensagens</i>	63
6.4.2	<i>Controle de Acesso ao Meio</i>	64
6.4.3	<i>Integridade dos dados</i>	67
6.5	ESTRUTURA DOS QUADROS.....	67
6.5.1	<i>Caractere de quadros (UART-Character)</i>	67
6.5.2	<i>Regra de transmissão</i>	67
6.5.3	<i>Sincronização de bit</i>	68
6.6	FORMATOS DOS QUADROS.....	68
6.6.1	<i>Quadros de tamanho fixo sem campo de dados</i>	68
6.6.2	<i>Quadros de tamanho fixo com campo de dados</i>	70
6.6.3	<i>Quadros com campo de dados de tamanho variável</i>	71
6.7	QUADRO DE TOKEN.....	72
6.8	COMPRIMENTO, ENDEREÇO, CONTROLE E TESTE.....	72
6.8.1	<i>Byte de Comprimento (LE)</i>	72
6.8.2	<i>Byte de endereço (DA/SA)</i>	73
6.9	A CAMADA DE APLICAÇÃO (7).....	74
6.9.1	<i>Modelo de Comunicação PROFIBUS</i>	74
6.9.2	<i>Objetos da Comunicação</i>	75
6.9.3	<i>Tipos de endereçamentos</i>	77
7.	A REDE DE MICROTERMINAIS	78
7.1	INTRODUÇÃO.....	78
7.2	DESCRIÇÃO DE UM APLICATIVO.....	78
7.3	O MESTRE	79
7.3.1	<i>Hardware</i>	79
7.3.2	<i>Software</i>	79
7.4	OS ESCRAVOS	81
7.4.1	<i>Hardware</i>	81
7.4.2	<i>Software</i>	81
8.	DESCRIÇÃO DOS TEMPOS	83
8.1	INTRODUÇÃO.....	83
8.2	TEMPOS DE SUPERVISÃO.....	83
8.2.1	<i>Tempo de Sincronismo - T_{SYN}</i>	83
8.2.2	<i>Tempo de Espera - T_{QUI}</i>	84
8.2.3	<i>Tempo de Atraso na Transmissão - T_{TD}</i>	84
8.2.4	<i>Margem de Segurança - T_{SM}</i>	84
8.2.5	<i>Tempo de Reação da Estação - T_{SDx}</i>	84
8.2.6	<i>Tempo de Disponibilidade - T_{RDY}</i>	85
8.2.7	<i>Tempo de Estado de Repouso - T_{ID}</i>	85
8.2.8	<i>Tempo de Espera pela Recepção - T_{SL}</i>	87
8.2.9	<i>Tempos de Transmissão de Quadro - $T_{S/R}$, $T_{A/R}$</i>	88
8.2.10	<i>Time-out - T_{TO}</i>	89
8.2.11	<i>Tempo de Atualização - T_{GUD}</i>	89
8.3	TEMPO DE CICLO DE MENSAGENS - MC	90
8.4	TAXA DE MENSAGENS R_{SYS}	91
8.5	TEMPO DE REAÇÃO DO SISTEMA - T_{SR}	91
8.6	DESENVOLVIMENTO DAS EQUAÇÕES.....	91
8.6.1	<i>Cálculo de t_{MC}</i>	91
8.6.2	<i>Cálculo de R_{SYS}</i>	92
8.6.3	<i>Cálculo de t_{SR}</i>	92
8.7	VALORES DA TABELA A.1 DA NORMA PROFIBUS.....	92
8.7.1	<i>Considerações</i>	92
9.	TESTES REALIZADOS	95
9.1	DESCRIÇÃO	95
9.2	ESTAÇÃO MESTRE.....	95

9.3	SISTEMA DE MONITORAÇÃO.....	95
9.4	ESTAÇÕES ESCRAVAS.....	96
9.5	PROGRAMA DE TESTES.....	98
9.6	TESTES A 19200 BPS.....	99
9.7	TESTES A 28800 BPS.....	100
10.	ANÁLISE DOS TESTES.....	102
10.1	CÁLCULO DOS VALORES TEÓRICOS A 19200 BPS.....	102
10.1.1	<i>Considerações.....</i>	<i>102</i>
10.2	VALORES MEDIDOS A 19200 BPS.....	103
10.2.1	<i>Análise do Ciclo SRD a 19200 bps.....</i>	<i>104</i>
10.2.2	<i>Tempo do Ciclo de Mensagem para um Ciclo SRD.....</i>	<i>108</i>
10.2.3	<i>Tempo do Ciclo de Mensagem para um Ciclo SDA.....</i>	<i>108</i>
10.3	TEMPO DE CICLO DE MENSAGEM A 28800 BPS.....	110
10.3.1	<i>Tempo do Ciclo de Mensagem para um Ciclo SRD.....</i>	<i>110</i>
10.3.2	<i>Tempo do Ciclo de Mensagem para um Ciclo SDA.....</i>	<i>111</i>
11.	CONCLUSÕES.....	112
12.	BIBLIOGRAFIA.....	116
A.	ANEXO A - RESULTADOS DOS TESTES PRÁTICOS.....	118
A.1.	RESULTADOS A 19200 BPS – 12 ESTAÇÕES ESCRAVAS.....	118
A.2.	RESULTADOS A 19200 BPS – 24 ESTAÇÕES ESCRAVAS.....	121
A.3.	RESULTADOS A 19200 BPS – 48 ESTAÇÕES ESCRAVAS.....	123
A.4.	RESULTADOS A 19200 BPS – 72 ESTAÇÕES ESCRAVAS.....	125
A.5.	RESULTADOS A 19200 BPS – 96 ESTAÇÕES ESCRAVAS.....	127
A.6.	RESULTADOS A 28800 BPS – 12 ESTAÇÕES ESCRAVAS.....	129
A.7.	RESULTADOS A 28800 BPS – 24 ESTAÇÕES ESCRAVAS.....	131
A.8.	131
A.9.	RESULTADOS A 28800 BPS – 31 ESTAÇÕES ESCRAVAS.....	133
B.	ANEXO B - ANÁLISES DOS CICLOS DE COMUNICAÇÃO.....	135
B.1.	CICLO SDA A 19200 BPS.....	135
B.2.	CÁLCULO DOS VALORES TEÓRICOS A 28800 BPS.....	139
B.3.	CICLO SRD A 28800 BPS.....	140
B.4.	CICLO SDA A 28800 BPS.....	144
C.	ANEXO C – LISTAGEM DE UM ARQUIVO DE TESTE.....	147
C.1.	INTRODUÇÃO.....	147
D.	ANEXO D - BIBLIOTECA DAS FUNÇÕES EM C.....	149
E.	ANEXO E – TESTES DE UTILIZAÇÃO DE SCANNERS NOS CAIXAS DE SUPERMERCADO.....	154
	154
E.1.	BENEFÍCIOS DIRETOS DOS SCANNERS NO PONTO DE VENDA.....	154
E.1.1.	<i>Tempo na fila do caixa.....</i>	<i>154</i>
E.1.2.	<i>Tempo de atenção ao cliente.....</i>	<i>155</i>
E.2.	BENEFÍCIOS DIRETOS, PRODUTIVIDADE NA FILA DO CAIXA.....	155

LISTA DE FIGURAS

FIGURA 1. O MODELO DE REFERÊNCIA OSI.....	27
FIGURA 2. MAP, EPA E MINI-MAP [BEN93]	42
FIGURA 3. PROCESSO TÉCNICO [BEN93]	46
FIGURA 4. TIPOS DE PROCESSOS [BEN93]	47
FIGURA 5. LAÇOS DE CONTROLE NA PRODUÇÃO [BEN93]	49
FIGURA 6. TIPOS DE MENSAGENS E CARACTERÍSTICAS TÉCNICAS DO DADOS [BEN93]	52
FIGURA 7. PROFIBUS: FAIXA DE APLICAÇÕES	59
FIGURA 8. A ARQUITETURA DO PROTOCOLO PROFIBUS-FMS E PROFIBUS-DP.....	60
FIGURA 9. CONEXÃO DO PROFIBUS-DP E PROFIBUS-FMS.....	62
FIGURA 10. ACESSO AO MEIO DO TIPO HÍBRIDO	66
FIGURA 11. UART - CHARACTER	67
FIGURA 12. VIRTUAL FIELD DEVICE (VFD) AND OBJECT DICTIONARY (OD).....	75
FIGURA 13. IDLE-TIME T_{ID1}	86
FIGURA 14. IDLE-TIME T_{ID2}	86
FIGURA 15. SLOT-TIME T_{SL1}	87
FIGURA 16. SLOT-TIME T_{SL2}	88
FIGURA 17. CICLO DE MENSAGEM	90
FIGURA 18. TERMINAL PARA VOTO ELETRÔNICO.....	97
FIGURA 19. REDE DE TESTES.....	99
FIGURA 20 – A ESTANTE COM OS VE’S DURANTE OS TESTES.....	100
FIGURA 21 – OS VE’S NAS PRATELEIRAS	101
FIGURA 22. GRÁFICO REFERENTE À TABELA 11	105
FIGURA 23. GRÁFICO REFERENTE À TABELA 12	106
FIGURA 24. GRÁFICO REFERENTE À TABELA 13	106
FIGURA 25. GRÁFICO REFERENTE À TABELA 14	107
FIGURA 26. GRÁFICO REFERENTE À TABELA 15	107
FIGURA 27. GRÁFICO REFERENTE À TABELA 16	108
FIGURA 28. GRÁFICO REFERENTE À TABELA 17	109
FIGURA 29. GRÁFICO REFERENTE À TABELA 18	110
FIGURA 30. GRÁFICO REFERENTE À TABELA 19	111
FIGURA 31. GRÁFICO REFERENTE À TABELA 20	113
FIGURA 32. GRÁFICO REFERENTE À TABELA 21	113

LISTA DE TABELAS

TABELA 1- PADRÕES DE REDES ISO E CCITT	37
TABELA 2 - AS SUÍTES DE PROTOCOLOS TOP E MAP [TAN94]	40
TABELA 3 - TAXAS DE TRANSMISSÃO X COMPRIMENTO MÁXIMO DO BARRAMENTO NO PROFIBUS.	63
TABELA 4 - SERVIÇOS DA CAMADA DE ENLACE DE DADOS DO PROFIBUS[VOL97].....	64
TABELA 5 - REGISTRO DO LIVRO	79
TABELA 6 - REGISTRO DO CLIENTE	79
TABELA 7. TABELA A.1 DA NORMA PROFIBUS PARA 19200 BPS	93
TABELA 8. RESULTADO DOS CÁLCULOS TEÓRICOS	94
TABELA 9 - ESTADOS DO VE.....	97
TABELA 10. RESULTADOS TEÓRICOS – Tx 19200 BPS.....	103
TABELA 11. TAXA:19200 - 12 ESTAÇÕES - CICLO SRD (μ S).....	105
TABELA 12. TAXA:19200 - 24 ESTAÇÕES - CICLO SRD (μ S).....	106
TABELA 13. TAXA:19200 - 48 ESTAÇÕES - CICLO SRD (μ S).....	106
TABELA 14. TAXA:19200 - 72 ESTAÇÕES - CICLO SRD (μ S).....	107
TABELA 15. TAXA:19200 - 96 ESTAÇÕES - CICLO SRD (μ S).....	107
TABELA 16- TAXA: 19200 BPS - TEMPO DO CICLO SRD (MS)	108
TABELA 17 – TAXA: 19200 BPS - CICLO SDA (MS)	109
TABELA 18 – TAXA: 28800 BPS - TEMPO DO CICLO SRD (MS)	110
TABELA 19 - TAXA: 28800 BPS – TEMPO DO CICLO SDA (MS).....	111
TABELA 20 – TEMPO DE REAÇÃO DO SISTEMA A 19200 BPS – (MS).....	113
TABELA 21 – TEMPO DE REAÇÃO DO SISTEMA A 28800 BPS – (MS).....	113
TABELA 22 – TEMPO DE REAÇÃO DO SISTEMA – DATA_UNIT = 30 BYTES / CICLO – (MS).....	114

LISTA DE ABREVIATURAS

ALI	Application Layer Interface
AP	Application Processes
ARPA	Advanced Research Projects Agency
AS	Abstract Syntax
ASN.1	Abstract Syntax Notation One
BITNET	Because Its Time NETwork
BMFT	German Federal Ministry of Research and Technology
BNS	Backbone Network Service
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CAN's	Campus Area Networks
CAP	Computer Aided Planning
CAQ	Computer Aided Quality Assurance
CCITT	Comité Consultatif International Téléphonique et Télégraphique
CIM	Computer Integrated Manufacturing
CLNS	Connection-Network Services
CLP	Controlador Lógico Programável
CNC	Controlador Numérico Computadorizado
CONS	Connection-Oriented Network Services
CP	Controlador Programável
CPD	Centro de Processamento de Dados
CSRD	Cyclic Send and Request Data with Reply
DA	Destination Adress
DARPA	Defense Advanced Research Projects Agency
DAS	Send Data with Acknowledge
DDLMM	Direct Data Link Mapper
DP	Decentral Periphery
ED	End Delimiter
EPA	Enhanced Performance Architecture
ES	End Systems
FC	Frame Control
FCS	Frame Check Sequence
FDL	Fieldbus Data Link
FMS	Flexible Manufacturing System
FMS	Fieldbus Message Specification
FTAM	File Transfer, Access and Management
FTP	File Transfer Protocol
HDLC	High Level Data Link Control) da ISO (International Organization for Standardization
IEC	International Electrotechnical Comission

IMP	Interface Message Processors
ISP	Inter-Operable Systems Project
IP	Internet Protocol
IS's	Intermediate-Systems
ISO	International Organization for Standardization
ITU	International Telecommunication Union
JTM	Job Transfer and Manipulation
L	Length
LAN's	Local Area Networks
LE	LEngth
LEr	LErepeated
LLC	Logical Link Control
LLI	Lower Layer Interface
LS	Local Syntax
MAC	Medium Access Control
MAN	Metropolitan Area Networks
MAP	Manufacturing Automation Protocol
MMS	Manufacturing Message Specification
MOTIS	Message Oriented Text Interchange Systems
NS	Next Station
NSF	National Science Foundation
OD	Object Dictionary
OSI	Open Systems Interconnection
PROFIBUS	PROcessFIeldBUS
PA	Process Automation
PAD	Packet Assembler Disassembler
PC	Personal Computer
PNO	PROFIBUS Nutzerorganisation - PROFIBUS Users Organization
PPC	Production Program Control
PS	Previous Station
PTT	Post, Telegraph & Telephone
SA	Source Adress
SC	Single Character
SD	Start Delimiter
SDN	Send Data with No acknowledge
SI	Sensory Information
SMTP	Simple Mail Transfer Protocol
SNA	System Network Architecture
SRD	Send and Request Data with Reply
SYN	SYNcronis
T _{A/R}	Acknowledgement/Response frame Time
TCP	Transmission Control Protocol
T _{ID}	IDle Time
T _{MC}	Mensage Cycle Time
TOP	Technical and Office Protocols
T _{S/R}	Send/Request frame Time
TS	This Station
TS	Transfer Sintax

TSDR	Station Delay of Responder Time
TD	Transmission Delay Time
TWA	Two Way Alternate
TWS	Two Way Simultaneous
uucp	UNIX to UNIX CoPy
VFD	Virtual Field Device
VTP	Virtual Terminal Protocol
WAN's	Wide Area Network

RESUMO

Neste trabalho são descritas a arquitetura e as propriedades características do protocolo PROFIBUS, nome dado a um entre os diversos protocolos existentes na área industrial.

Utilizando uma versão do protocolo PROFIBUS desenvolvida no Laboratório de Instrumentação Eletro-Eletrônica da Universidade Federal do Rio Grande do Sul foi implementada uma rede composta por uma estação mestre e até 96 estações escravas.

A fim de analisar o Tempo de Ciclo de Mensagem e o Tempo de Reação do Sistema foram realizados diversos testes, variando a taxa de comunicação utilizada, o tamanho dos quadros de comunicação e a quantidade de estações escravas que compunham a rede.

No final foi feita uma comparação entre os valores práticos obtidos e valores calculados a partir de fórmulas teóricas.

Os valores práticos obtidos possibilitam determinar a viabilidade da utilização desta rede em diversas aplicações reais.

ABSTRACT

In this work were described the architecture and the properties of PROFIBUS protocol, name given to one of the various protocols in the industrial area.

Using a version of PROFIBUS protocol developed in the Electroelectronic Instrumentation Laboratory of the Rio Grande do Sul Federal University was implemented a network with one master station and 96 slaves stations.

Many tests were done to analyze the Message Cycle Time and the System Reaction Time, varying the communication baud rate, the size of the communication frames and the number of the slaves stations of the network.

Finally a comparison between the practical values obtained and values calculated from theoretical formulas was made.

The practical values obtained make possible to determine the utilization of this network in many actual applications.

1. INTRODUÇÃO

Nos últimos anos as redes de computadores passaram a fazer parte de nosso dia a dia. Para uma parcela cada vez maior da população o acesso doméstico à Internet já é uma realidade. Alias, esta rede é apresentada na mídia através de uma publicidade equivalente àquela utilizada para vender produtos tão comuns na nossa vida como um pacote de bolachas ou um refrigerante.

Mas aquelas pessoas que não possuem acesso à Internet desde sua própria casa ou escritório também convivem de forma consciente ou não com outros tipos de redes de comunicação cada vez que acessam às informações armazenadas nos bancos de dados nos caixas eletrônicos bancários, nos terminais de consulta de preços nos supermercados, nas pesquisas de livros em livrarias e bibliotecas, entre outros.

Estas operações de consulta tem-se tornado tão corriqueiras e presentes no nosso dia a dia que não paramos para pensar na tecnologia que se encontra por trás dessas comunicações.

Atualmente as mais diversas áreas se beneficiam com o uso de redes de computadores aproveitando assim, as facilidades da comunicação entre computadores melhorando o desempenho dos trabalhos. Como exemplos pode-se citar automação de escritórios, universidades, empresas comerciais e indústrias. O tipo de rede utilizado em cada caso apresenta características próprias pois os ambientes de comunicação são diferentes.

Ao longo desta dissertação serão apresentados vários tipos de redes passando pela sua evolução histórica, conceitos sobre arquitetura de redes e os protocolos utilizados.

O protocolo Profibus será analisado especialmente de forma teórica e prática. Este protocolo foi desenvolvido especificamente para a comunicação de redes na área industrial, conhecida como chão de fábrica. Neste trabalho será avaliada a

possibilidade de aplicar este protocolo em outras áreas, através da análises dos tempos de resposta e tamanho dos quadros, exigidos em cada caso.

Os primeiros capítulos fazem uma introdução ao mundo das redes para situar o leitor no contexto e na linguagem que será utilizada nos seguintes capítulos.

O Capítulo 2 - Nos fala do crescimento do uso dos computadores a partir da década de 50 e dos diversos motivos que determinaram a origem do uso de redes de computadores. Como também apresenta a classificação e as propriedades das redes em função da distribuição geográfica dos computadores que a compõem.

O Capítulo 3 – Traz a descrição de arquitetura de redes de computadores e a definição de um modelo de sistema de comunicação estruturado em camadas, para especificação da comunicação aberta, conhecido como modelo OSI/ISO.

O Capítulo 4 – Este capítulo apresenta um resumo da evolução histórica das redes até os nossos dias. Citando e descrevendo importantes redes tal como por exemplo a ARPANET precursora da rede mais famosa dos nossos dias a INTERNET.

O Capítulo 5 – Fala sobre o Projeto Industrial, descrevendo as necessidades nos diferentes níveis da produção industrial o que determina o uso de protocolos específicos para cada um destes níveis descritos.

O Capítulo 6 - Neste capítulo são descritas sucintamente as propriedades característica do PROFIBUS nome dado a uma entre as diversas redes proprietárias existentes na área de chão de fábrica.

O Capítulo 7 – Descreve um aplicativo que utiliza o protocolo PROFIBUS, para implementar a comunicação de uma rede monomestre cujo comportamento será analisado nesta dissertação.

No Capítulo 8 é feita uma análises temporal detalhada dos diversos quadros de comunicação do protocolo PROFIBUS. Também são desenvolvidas as equações que serão utilizadas na análises dos dados obtidos nos testes práticos.

O Capítulo 9 – Descreve os equipamentos utilizados nos testes: estação de monitoração, estação mestre e estações escravas. É neste capítulo também que são estabelecidos os procedimentos a serem utilizados na aquisição dos dados e na monitoração, a definição do tamanho dos quadros e o *lay-out* da rede.

O Capítulo 10 – Apresenta a análise dos dados obtidos nos testes e a comparação com os valores teóricos através de equações, tabelas e gráficos.

É no Capítulo 11 que estão colocadas as conclusões finais assim como as sugestões para a utilização dos dados obtidos neste trabalho para o análises de tempos de resposta de outras redes.

Finalizo estas linhas na esperança que esta dissertação possa contribuir no trabalho de outras pessoas.

2. REDES DE COMPUTADORES

2.1 INTRODUÇÃO HISTÓRICA

Na década de 1950 os computadores começaram a ser utilizados comercialmente. Por serem muito caros, sua utilização estava restrita a órgãos do governo e a grandes empresas. Uma empresa de porte médio ou universidade poderia ter um ou dois computadores, enquanto as grandes instituições tinham no máximo uma dúzia. [GIO86]

Naturalmente eram usados apenas em aplicações onde a rapidez do processamento da informação era tão vital que justificava o alto investimento na sua aquisição.

Durante as duas primeiras décadas de sua existência, o computador era utilizado de forma altamente centralizada, organizado em torno do Centro de Processamento de Dados (CPD) que operava o computador da empresa e procurava otimizar o uso desse recurso compartilhando entre as aplicações dos diversos usuários e utilizando-o o maior número de horas possível. Não raro essa sala tinha paredes de vidro através das quais os visitantes podiam olhar, admirados, para a grande maravilha eletrônica que estava lá dentro.

A indústria dos computadores tem feito um progresso espetacular em pouco tempo, graças aos avanços na tecnologia de microeletrônica que levaram a uma contínua redução do custo dos equipamentos de computação, culminando com o surgimento dos mini e microcomputadores de custo tão baixo que já estão ao alcance das pequenas empresas e do cidadão comum.

A queda rápida dos preços já se tornou marca registrada na indústria da informática. As causas desse fenômeno podem ser identificadas em diferentes pontos. O primeiro refere-se à elevada escala de produção atingida pelos microcomputadores padrão PC (Personal Computer), que atualmente contabiliza cerca de 40 milhões de

máquinas anuais. Outro fator fundamental é a própria evolução da tecnologia. A miniaturização dos dispositivos vem permitindo conciliar movimentos abertamente contraditórios. Em menor espaço, reúnem-se mais componentes capazes de trabalhar com maior capacidade. Isso, ao mesmo tempo, contribui para a conquista de menores custos de produção - ganho que em parte é repassado para o consumidor na forma de redução dos preços.[FONTE]

A competição na indústria constitui um terceiro fator que exerce destacada influência sobre os preços. O caso do microprocessador, o principal componente dos sistemas, ilustra bem esse processo. Líder absoluta do mercado a Intel enfrenta uma série de empresas menores que fabricam clones de seus chips. Para se manter à frente, ela promove uma autêntica corrida tecnológica, lançando periodicamente dispositivos mais poderosos e, em paralelo, reduzindo os preços dos *chips* de gerações menos recentes. Aos fabricantes também interessa matar as máquinas e os processadores mais velhos porque são eles que rendem as menores margens de lucro.

Como conseqüência dessa evolução tecnológica, a forma como a atividade de processamento de dados estava organizada nas empresas está mudando. Nas grandes empresas, além do computador de grande porte, é comum a existência de mini e microcomputadores instalados fora dos CPD's, junto ao usuário final. As empresas pequenas e médias podem hoje em dia utilizar processamento de dados e a maioria o faz baseada em um ou vários mini e microcomputadores.

Nesse contexto, as comunicações crescem de importância por serem o elemento que permitirá que essas máquinas trabalhem de forma cooperativa. Substituindo o velho modelo de um único computador servindo a todas as necessidades computacionais da organização por outro no qual um grande número de computadores separados, mas interconectados, executam a tarefa. Esses sistemas são chamados de **redes de computadores**.

Dez anos atrás as redes de computadores, utilizadas apenas por alguns poucos especialistas, eram consideradas ferramentas de pesquisas exóticas. A sua utilização mudou enormemente na última década, a maioria das organizações que usam computadores já tem, ou planeja instalar em breve, uma ou mais redes locais de computadores. O correio eletrônico em âmbito mundial é uma realidade diária para

milhões de pessoas. Em resumo, as redes evoluíram de simples curiosidade acadêmica para se tornarem uma ferramenta essencial aos usuários nas empresas, governos e universidades.

Muitas organizações já têm um número substancial de computadores em operação, muitas vezes situados em pontos distantes entre si. Por exemplo, uma empresa com muitas fábricas pode ter um computador em cada local para acompanhar o inventário, monitorar a produtividade e emitir folha de pagamento local. Para poder extrair e correlacionar informações acerca de toda a empresa torna-se interessante conectar estes computadores que trabalham em forma isolada numa rede.

Uma das principais vantagens de conectar os computadores em rede é o **compartilhamento de recursos**, cujo objetivo é fazer com que todos os programas, dados e equipamentos estejam disponíveis para qualquer um na rede, independentemente da localização física do recurso e do usuário. O compartilhamento da carga é um outro aspecto do compartilhamento de recursos. Nessa estrutura os micros usuários podem ter seus próprios discos e ter acesso também a um disco de maior capacidade através da rede. Podem ter a sua própria impressora ou ter acesso a uma impressora comum através da rede.

Uma outra vantagem é a **alta confiabilidade**, tendo-se fontes de suprimento alternativas. Por exemplo, todos os arquivos poderiam ser replicados em duas ou três máquinas de forma que, se uma delas não estiver disponível (devido a uma falha de hardware), as outras serão capazes de assumir o trabalho, embora com desempenho reduzido. Para aplicações militares, bancárias, de controle de tráfego aéreo e muitas outras, a habilidade de continuar operando mesmo na presença de problemas de hardware é de grande importância.

A **economia** representa mais uma vantagem. Computadores de pequeno porte têm uma razão custo/desempenho muito melhor do que os de grande porte. Os *mainframes* são aproximadamente dez vezes mais rápidos do que o mais rápido microprocessador de um *chip*, mas custam centenas a milhares de vezes mais. Esse desequilíbrio fez com que muitos projetistas de sistemas construíssem sistemas constituídos de computadores pessoais potentes, um por usuário, com dados guardados

em uma ou mais máquinas **servidoras de arquivos, de comunicação, de impressão, etc.**

Uma questão intimamente relacionada é a habilidade de aumentar o desempenho do sistema de forma gradual, conforme a carga aumenta, pela simples adição de mais processadores. Com os mainframes centrais, quando o sistema está totalmente carregado, este deve ser substituído por um maior, normalmente a um custo elevado e com maiores inconvenientes para os usuários.

Uma outra vantagem adicional para se estabelecer uma rede de computadores tem pouca relação com a tecnologia. Uma rede de computadores pode prover um **meio de comunicação** muito poderoso entre pessoas dispersas geograficamente. Utilizando-se de uma rede de computadores, é fácil para duas ou mais pessoas que moram em lugares distantes entre si escreverem um relatório em conjunto. Quando um dos autores faz uma mudança no documento, que é mantido online, os outros vêem essa mudança imediatamente, em vez de ter que esperar vários dias por uma carta. Longe de ser impossível, como era antes, essa aceleração torna até fácil a cooperação entre grupos dispersos em grandes áreas. A longo prazo, o uso de redes de computadores para melhorar a comunicação entre as pessoas talvez se mostre mais importante do que objetivos técnicos, como confiabilidade.

2.2 CLASSIFICAÇÃO

As redes de computadores são utilizadas atualmente nas mais diversas áreas aproveitando a facilidade de comunicação entre computadores para melhorar o desempenho dos trabalhos. Como exemplos pode-se citar automação de escritórios, universidades, empresas comerciais e indústrias. Cada um destes sistemas possui características diferentes pois têm ambientes de comunicação diferentes.

As redes apresentam-se, na prática, com uma grande diversidade em relação às suas topologias, meios de transmissão, métodos de comunicação, etc. Não existe uma forma genérica de rede que satisfaça os requisitos operacionais existentes em todas as aplicações.

Um dos critérios possíveis para classificar redes de computadores é o critério de localidade, isto é, distância máxima entre usuários interligados pela rede.

Definem-se a seguir as diferentes classes de redes de comunicação, segundo este critério:

2.2.1 Redes Locais - LAN's

São definidas como sistemas de comunicação a distâncias pequenas (entre centenas de metros e 2,5 Km), com número variável (até centenas) de sistemas finais ES's (*End Systems*). Em virtude da necessidade de compartilhamento do mesmo meio de transmissão, as LAN's (*Local Area Networks*) podem apresentar alto desempenho, traduzido em taxas de transmissão de dados de até alguns Mega bits/segundo (Mbps). Como estão confinadas em áreas pequenas, são geralmente de propriedade da empresa usuária. É freqüente a decomposição de uma LAN em várias sub-LAN's (sub-redes) com meios separados de comunicação. No caso, as sub-LAN's são interligadas por sistemas intermediários IS's (*Intermediate-Systems*) simples que funcionam como repetidores da mensagem de um para outro segmento. [MEN91]

2.2.2 Campus Area Networks - CAN's

São sistemas de comunicação a distâncias maiores (de 2 a 10 Km), interligando ES's ou LAN's situadas num "campus" (empresa, fábrica, universidade, etc.). Uma CAN surge como extensão do conceito de LAN. Em geral usa-se um meio de interligação chamado de *backbone* com características de desempenho semelhantes às LAN's interligadas, de forma a evitar-se congestionamento na transmissão da informação. [MEN91]

2.2.3 Redes de Área Metropolitana - MAN's

Uma MAN (*Metropolitan Area Networks*) é uma rede situada numa região metropolitana (distâncias da ordem de 5 a 100 Km), mas usa tecnologia LAN. As redes de TV a cabo são exemplos de MAN's analógicas para a distribuição de televisão. As MAN's que interconectam computadores e não aparelhos de televisão são digitais, muito embora algumas utilizem cabos coaxiais em banda larga como meio de transmissão.

Alguns trabalhos de padronização publicados, introduzem a terminologia de redes metropolitanas como extensão de LAN's e CAN's. [MEN91]

2.2.4 Redes de Longa Distância - WAN's

As WAN's (*Wide Area Networks*) são definidas para comunicação a grandes distâncias, já do domínio público (por exemplo, redes de telefonia, satélites, etc.) e cujos parâmetros de desempenho são inferiores a LAN's.

A maior WAN que existe atualmente é a Internet. Atualmente a Internet conta com mais de 100 milhões de usuários espalhados pelo mundo e este número cresce rapidamente. Mais de 100 países estão conectados e intercambiam dados, notícias e opiniões.

3. MODELO OSI/ISO E PROTOCOLOS

3.1 INTRODUÇÃO

A medida que o uso dos computadores foi aumentando surgiu a necessidade de comunicação de dados digitais, aparecendo diversas soluções específicas de cada área, de cada equipamento, de cada fabricante. Nos anos 70, os novos conhecimentos de arquiteturas de redes de computadores e o esclarecimento de conceitos de sistemas distribuídos de processamento da informação possibilitou o desenvolvimento dos primeiros padrões de comunicação destacando-se os padrões X.21 e X.25, do CCITT (*Comité Consultatif International Téléphonique et Télégraphique*), para interconexão de terminais e comutação de pacotes em redes públicas e o HDLC (*High Level Data Link Control*) da ISO (*International Organization for Standardization*), para o controle de transmissão de dados através de enlaces. Contudo a falta de um padrão de comunicação em aplicações de sistemas distribuídos tornou clara a necessidade de definir uma estrutura lógica que disciplinasse trabalhos futuros na área. Foi com este objetivo que nasceu na ISO, em 1977, O Comitê SC16 (subordinado ao TC 97) afim de desenvolver uma arquitetura para a definição, desenvolvimento e validação de padrões para sistemas de processamento distribuído.

No final de 1979 a ISO convergiu para a definição de um modelo do sistema de comunicação estruturado em camadas, para especificação da comunicação aberta. A CCITT reconheceu igualmente a importância dessa arquitetura e iniciou um trabalho conjunto com a ISO. Depois de vários estágios intermediários de formulação, foi aprovado em 1983 pela ISO o documento definitivo IS 7498, e na CCITT o modelo paralelo no documento X.200, orientado para aplicações em redes públicas.

3.2 CONCEITO

O modelo de referência da ISO/CCITT procede a uma subdivisão da funcionalidade em 7 níveis, correspondentes às 7 camadas do modelo:

- Camada de aplicação (camada 7);
- Camada de apresentação (camada 6);
- Camada de sessão (camada 5);
- Camada de transporte (camada 4);
- Camada de rede (camada 3);
- Camada de enlace (camada 2);
- Camada física (camada 1).

O conceito de projeto modular é conhecido amplamente na prática de engenharia de software. Funções semelhantes são agrupadas na mesma camada, comunicando-se cada camada N com as duas vizinhas, N-1 e N+1, através de interfaces. O conjunto de regras que define a cooperação entre funções na mesma camada, mas localizadas em estações diferentes (funções - pares = "peer") é designado como interface "intralayer" ou mais comumente de protocolo.

Esta decomposição por camadas apresenta inúmeras vantagens:

- As estruturas internas, mecanismos, codificação e algoritmos usados em cada camada são transparentes para as outras;
- Em cada camada poderão oferecer-se serviços/funções alternativas, dependendo das aplicações da rede;
- Implementações alternativas de uma camada podem coexistir;
- Uma camada ou sub-camada pode ser simplificada ou omitida desde que os serviços/funções que contenha não sejam necessários;
- O sistema pode ser mais facilmente depurado, testado e corrigido, camada por camada.

Na figura 1 representa-se o modelo com as suas sete camadas, dividido em dois grandes sistemas, o de aplicação e do transporte.

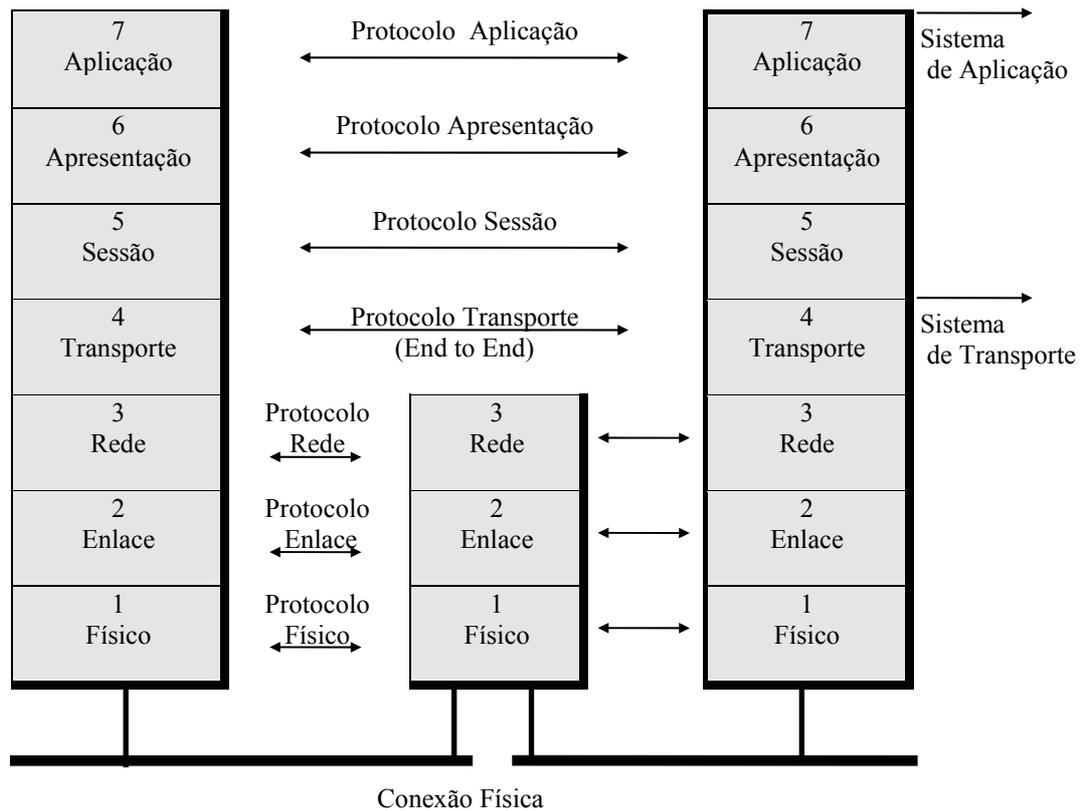


Figura 1. O Modelo de Referência OSI

O verdadeiro objetivo do SC16 não era apenas desenvolver um modelo de referência, mas criar um ambiente para a especificação ordenada e sistemática de protocolos, tendo como finalidade a comunicação aberta de sistemas computadorizados de qualquer fabricante, tipo, geração ou nível de complexidade. Estes protocolos vêm sendo especificados desde 1980, de acordo com a arquitetura e terminologia do modelo, separadamente para cada uma das camadas. Assim foram atacados os protocolos das camadas 1, 2 e 3, pelo TC 97/SC6 que tinha desenvolvido antes o protocolo HDLC, antes do modelo de referência. Já o comitê SC16 assumiu a responsabilidade das camadas 4, 5, 6 e 7. [MEN91]

Um dos principais méritos do modelo reside na viabilização de trabalhos em paralelo, simultâneos, nas diversas camadas. O modelo, no seu nível de abstração, não só não descreve a tecnologia de implementação, como também não lhe impõe quaisquer limites. Se bem que os objetos usados se assemelhem a conceitos de sistemas reais, eles constituem abstrações e não necessitam de ser reproduzidos na implementação.

3.3 DESCRIÇÃO

É feita aqui uma descrição sucinta das 7 camadas. As camadas são numeradas começando pela mais baixa (camada física) e terminando na mais elevada (camada de aplicação). A hierarquia de camadas corresponde a uma hierarquia de funções, em que algumas são conceitualmente de um nível mais baixo e outras de níveis mais elevados, as funções de nível superior. No modelo OSI exprime-se este fato dizendo que cada camada utiliza os serviços da camada adjacente inferior, fornecendo os seus serviços próprios à camada imediatamente acima.

3.3.1 Camada Física

A camada física é a primeira definida na arquitetura RM-OSI, oferece um serviço de transmissão transparente de seqüências de bits, trocadas pelas entidades da Camada de Enlace de Dados, através de um canal de comunicação (conexão-física).

As questões de projeto lidam com interfaces mecânicas, elétricas e procedimentais, e com o meio físico de transmissão que está abaixo da camada física. O Modelo OSI permite, em princípio, qualquer tipo de sinal (elétrico, ótico, acústico, eletromagnético, etc.).

A Camada Física ocupa-se, por exemplo, da regulamentação das interfaces mecânicas (número de pinos dos conectores, dimensões, etc.), das interfaces lógicas (dados, temporização, controle, etc.) e dos procedimentos de comando e gerência das conexões- físicas (estabelecimento, liberação, modo de funcionamento, etc.).

A unidade de transferência de informação na Camada Física é um bit quando a transmissão é efetuada em série e n bits quando é efetuada em paralelo.

Uma conexão física é identificada pela Camada de Enlace de Dados através de identificadores de pontos de terminação de conexão-física. Quando a conexão-física tem dois pontos de terminação tem-se uma conexão-física ponto-a-ponto. Por outro lado, quando a conexão-física é caracterizada por vários pontos de terminação, tem-se uma conexão-física multiponto.

A transmissão das seqüências de bits numa conexão-física pode ser realizada simultaneamente nos dois sentidos (duplex) ou alternando os sentidos da transmissão (semi-duplex).[GIO86]

Além disso, a transmissão das seqüências de bits pode ser efetuada segundo os modos de transmissão assíncrono ou síncrono. Os bits são transmitidos pela Camada Física, na mesma ordem em que ela os recebe, de modo que o sequenciamento da informação transmitida é mantido.

A qualidade de serviço oferecida à camada superior pela Camada Física é função dos meios físicos de transmissão que compõem as conexões-físicas. De maneira geral, os parâmetros de medida são os seguintes:

- taxa de erros;
- disponibilidade do serviço;
- taxa de transmissão;
- atraso de propagação.

3.3.2 Camada de Enlace de Dados

O objetivo principal da Camada de Enlace de Dados é o de detectar e, possivelmente, corrigir os erros que podem ocorrer na troca de informações (bits) da Camada Física. A camada de Enlace de Dados também fornece à camada superior (Camada de Rede) um serviço de conexões-de-enlace de dados. Uma conexão-de-enlace de dados é estabelecida (ou liberada) dinamicamente sobre um ou mais pares de pontos de terminação de conexão-física. No caso de conexões multiponto, a Camada de Enlace de Dados deve fornecer à camada superior a identificação dos pontos de terminação da conexão-de-enlace de dados.

A unidade de transferência de informação na Camada de Enlace de Dados corresponde a uma seqüência de bits sob a forma de quadros ou blocos. A caracterização dos quadros é feita através de delimitadores constituídos por caracteres ou seqüências especiais de bits no início e no fim de cada quadro. O comprimento da seqüência de bits (isto é, o tamanho dos quadros ou blocos) é limitado pela taxa de erros na conexão física e pela capacidade de detecção de erros da Camada de Enlace de Dados.

A Camada de Enlace de Dados pode fornecer um **serviço de controle de fluxo** para a camada superior, de modo a controlar a taxa de transferência de unidades de informação na interface com a Camada de Rede. Além disso, a Camada de Enlace de

Dados mantém o sequenciamento das unidades de informação transferidas na conexão-de-enlace de dados, quando requisitado pela camada superior. Um outro serviço fornecido é o de **controle de erros** onde a camada superior é notificada de qualquer erro detectado, mas não recuperável, pela Camada de Enlace de Dados.

Os parâmetros de qualidade de serviço na Camada de Enlace de Dados podem ser opcionalmente selecionáveis. Dessa forma, a Camada de Enlace de Dados estabelece e mantém uma qualidade de serviços selecionada para a duração de uma conexão-de-enlace de dados. Os parâmetros de qualidade de serviço incluem, por exemplo:

- o tempo médio entre erros detectados mas não recuperáveis;
- a taxa de erros residual;
- a disponibilidade do serviço;
- o atraso na transferência das unidades de informação;
- capacidade de transferência de informação (vazão)
- etc.

3.3.3 Camada de Rede

A camada de rede ocupa-se do encaminhamento ou roteamento da informação ao longo dos caminhos físicos que podem existir na subrede de comunicação. Pode também fornecer, sob demanda da camada superior, um serviço que mantenha o seqüenciamento da informação transmitida.

3.3.3.1 Roteamento

A Camada de Rede fornece endereços em nível de rede para que as entidades da Camada de Transporte possam se identificar mutuamente. O endereçamento a este nível pode ser independente daquele utilizado pelas camadas inferiores. Entre dois endereços a nível de rede, pode existir mais de uma conexão-de-rede estabelecida. A Camada de Rede fornece à camada superior os pontos de terminação da conexão de rede que, associados aos endereços em nível de rede, permitem identificar inequivocamente uma conexão-de-rede.

Roteamento é a determinação do caminho entre dois endereços a nível de rede dentre os diversos caminhos físicos existentes.

3.3.3.2 Sequenciamento

Existem na camada de rede dois tipos de serviços de rede: os serviços com conexão (“Connection-Oriented Network Services = CONS”) e os serviços sem conexão (“Connection-Network Services = CLNS”). Em particular procede-se nesta camada ao encaminhamento de pacotes provenientes de decomposição de mensagens longas. Nos serviços CONS, primeiro se estabelece uma conexão de rede ou um circuito virtual (lógico). Os serviços CONS garantem mais qualidade, mantendo a seqüência inicial dos pacotes, exercendo algum controle de fluxo e impondo parâmetros de qualidade de serviço a serem garantidos pelas camadas inferiores (vazões, atrasos, taxas de erro, etc.).

Já os serviços sem conexão, CLNS, são mais simples, mas também mais inseguros. A camada de rede repassa as mensagens para a camada inferior sem se preocupar com o estabelecimento de um circuito virtual; a transmissão será feita, sem qualquer tipo de controle de fluxo ou de seqüenciamento.

É evidente que a necessidade de protocolos de rede dependerá muito da topologia. No caso de LAN’s com topologias em barramento ou anel, poderá prescindir-se de roteamento função mais típica da camada de rede, e os protocolos serão de extrema simplicidade. Em geral só nas WAN’s é que os protocolos assumem toda a sua complexidade funcional.

Os parâmetros da qualidade de serviço podem ser selecionados e incluem, por exemplo:

- taxa de erros residuais;
- disponibilidade do serviço;
- confiabilidade de falhas numa conexão estabelecida;
- vazão ou capacidade de transferência da informação;
- atraso na transferência das unidades de informação;
- atraso no estabelecimento das conexões-de-rede.

3.3.4 Camada de Transporte

A camada de transporte é uma camada especial na arquitetura ISO que separa as camadas de aplicação (5, 6 e 7), das camadas voltadas à transmissão. Garantindo aos usuários das camadas superiores transparência e qualidade, independentemente das características das camadas inferiores, que podem ser dos mais diversos tipos: LAN com distâncias geográficas pequenas, redes de longa distância ou metropolitanas, comunicação pela rede telefônica, comunicação por satélite, etc.

Os serviços de transporte exprimem-se em termos de requisitos de qualidade de serviços como vazão (por ex. número de unidades de dados perdidas e o número total de unidades enviadas), probabilidades de falhas, atrasos de estabelecimento de conexão, etc.

A Camada de Transporte pode incluir ainda os seguintes serviços:

- controle de seqüenciamento fim-a-fim em cada conexão;
- detecção de erros fim-a-fim;
- recuperação de erros fim-a-fim;
- fragmentação de mensagens fim-a-fim;
- funções de supervisão.

A qualidade do serviço de uma conexão-de-transporte está associada à classe de serviço de transporte a que pertence. As classes de serviço, selecionadas na fase de estabelecimento da conexão, representam globalmente combinações predefinidas de parâmetros e graus de serviço tais como:

- vazão;
- atraso de transmissão;
- tempo de estabelecimento de conexão;
- taxa de erros residual;
- disponibilidade de serviço;

3.3.5 Camada de Sessão

Esta camada providencia funções de interconexão de dois processos de aplicação, AP's através de uma relação lógica de comunicação e de serviços de organização e sincronização do diálogo. Para tal tem que ser estabelecida primeiro uma conexão de sessão entre dois AP's, que não passa de um acordo, anterior ao envio dos dados propriamente ditos, no qual se estabelecem as regras e formas do diálogo. A iniciação de uma conexão de sessão tem normalmente origem no "AP-fonte" que deseja enviar informação. E dependerá dos serviços das camadas inferiores que fecharão em cada nível, acordos específicos próprios, ou seja, conexões específicas da camada.

Durante a fase de estabelecimento da conexão da sessão definem-se, entre outras, as regras de diálogo, para o qual existem três variantes:

- TWS (*Two Way Simultaneous*), em que dois AP's estão autorizados a enviar simultaneamente dados através de serviços da rede das camadas inferiores;
- TWA (*Two Way Alternate*), em que os dois lados podem enviar mensagens, mas um de cada vez, alternadamente, fluindo a informação num certo instante só num sentido;
- One Way, com a informação sendo enviada só num sentido.

Através da troca de mensagens, durante essa fase, são enviados os parâmetros que estabelecem o tipo de diálogo, que permanece inalterado até o final da conexão. A conexão terminará quando um dos AP's considere o término normal, ou em situações anormais de falhas detectadas. Para a troca de dados propriamente dita, existirão na camada de sessão uma série de funções que garantem sincronização e segurança de transmissão.

A camada de sessão é basicamente uma invenção da ISO. Antes do modelo OSI, nenhuma rede existente possuía uma camada de sessão. Ela não é tão importante quanto, digamos, a camada de transporte, e muitas aplicações nem sequer necessitam dos poucos recursos que ela possa ter.

3.3.6 Camada de Apresentação

Ao contrário das cinco camadas inferiores que lidam simplesmente com a movimentação ordenada de bits desde a origem até o destino, a camada de apresentação

se preocupa com a preservação do *significado* das informações transportadas. Cada computador pode ter sua própria forma de representação interna dos dados; assim, são necessários acordos e conversões a fim de assegurar que computadores diferentes possam compreender um ao outro.

Para descrever os dados a serem transmitidos de forma independente do equipamento definem-se os procedimentos e os tipos de informação necessários à associação entre AP's pares, através de *sintaxes abstratas* (AS's). As regras de notação de sintaxe, comuns para todos os protocolos, estão descritas na chamada ASN.1 (*Abstract Syntax Notation One*) que é hoje padrão internacional na ISO e CCITT. A notação ASN.1 permite a definição de *sintaxes abstratas* para cada protocolo de aplicação.

Por outro lado os dados e programas que existem localmente num computador, são representados segundo convenções próprias, chamadas de *sintaxes concretas*: no caso de sistemas heterogêneos é freqüente que estas *sintaxes locais* (LS) sejam diferentes, de acordo com regras locais de codificação. Para que os AP's não tenham que se acertar na mesma sintaxe local, é importante que estejam de acordo com uma *sintaxe concreta* comum de transferência de dados, chamada de *sintaxe de transferência* (TS). As funções características a serem implementadas na camada de apresentação providenciam mecanismos que permitem aos AP's definir e selecionar TS's para comunicação, através da negociação entre um conjunto de TS's que cada sistema suporte, escolhendo-se aquela mais adequada(eficiente) para a aplicação envolvida.

3.3.7 Camada de Aplicação

A camada de aplicação contém os programas do usuário, também conhecidos como aplicações ou aplicativos, que fazem o verdadeiro trabalho para o qual os computadores foram adquiridos. Esses programas utilizam os serviços oferecidos pela camada de aplicação para suas necessidades de comunicação. Entretanto, certas aplicações, tais como a transferência de arquivos, são tão comuns que foram desenvolvidos padrões a fim de eliminar a necessidade de cada empresa desenvolver sua própria aplicação e para garantir que todos usem os mesmos protocolos.

4. EVOLUÇÃO HISTÓRICA DAS REDES

4.1 INTRODUÇÃO

Existem diferentes tipos de redes de computadores espalhadas pelo mundo todo. Estas redes se diferenciam em função de suas histórias e administração, as facilidades oferecidas, seus projetos técnicos e pela comunidade de usuários.

Podem ser redes públicas gerenciadas por operadoras ou PTT's (*Post, Telegraph & Telephone*), redes de pesquisa, redes cooperativas administradas pelos seus usuários ou ainda podem ser redes comerciais ou corporativas.

A história e a administração podem variar desde uma rede cuidadosamente planejada por uma única organização com um objetivo bem definido até um conjunto *ad hoc* de máquinas que se conectaram umas às outras ao longo dos anos sem nenhum plano mestre ou administração central.

As facilidades variam desde a comunicação arbitrária processo a processo até o correio eletrônico, a transferência de arquivos, execução e *login* remotos.

Os projetos técnicos podem diferir no meio de transmissão utilizado, nos algoritmos de nomeação e roteamento empregados, no número e conteúdo de camadas presentes e nos protocolos utilizados.

Finalmente, a comunidade de usuários pode variar de uma única corporação até todos os cientistas de computação acadêmicos do mundo industrializado.

Algumas redes representativas serão descritas nos itens a seguir.

4.2 REDES PÚBLICAS

A rede pública é um serviço de rede por assinatura que em muitos países é oferecido pelo governo ou por companhias privadas. A sub-rede pertence ao operador da rede, fornecendo serviço de comunicação aos *hosts* e aos terminais do cliente.

Essas redes públicas seguem técnicas de comutação de pacotes segundo os protocolos conhecidos por X.25, que é o número da recomendação publicada pelo CCITT para as três camadas inferiores da arquitetura descrita no capítulo anterior. O CCITT (*Comité Consultatif International Téléphonique et Télégraphique*) é a organização internacional encarregada de estabelecer as normas na área das comunicações, é atualmente conhecida pela sigla ITU (*International Telecommunication Union*).

O protocolo da camada física, chamado de X.21, especifica as interfaces físicas, elétricas e procedimentais entre o *host* e a rede.

O padrão da camada de enlace de dados possui diversas variantes (ligeiramente incompatíveis). Todas são projetadas para lidar com erros de transmissão nas linhas telefônicas entre o equipamento do usuário (*host* ou terminal) e a rede pública IMP (*Interface Message Processors*).

O protocolo da camada de rede lida com o endereçamento, controle de fluxo, confirmação de entrega, interrupções e outras questões relacionadas.

Para aqueles terminais que não falam X.25, foi definido um conjunto de padrões que permitem a um terminal comum se comunicar com uma rede pública X.25. Uma interface chamada de PAD, Montador Desmontador de Pacotes (*Packet Assembler Disassembler*), é utilizada; a função da mesma é descrita em uma recomendação do CCITT conhecida por X.3. Foi definido um protocolo padrão entre o terminal e o PAD, chamado X.28, e outro entre o PAD e a rede, chamado X.29. Juntas essas três recomendações são freqüentemente chamadas de **triplo X**. [TAN94]

A tabela 1 mostra os números de alguns dos padrões internacionais principais. O estilo dos padrões muda radicalmente entre as camadas 3 e 4, dado que os inferiores foram elaborados pelo CCITT e os superiores pela ISO. Em particular, o CCITT nunca se deu ao trabalho de distinguir entre serviço e protocolo, o que foi meticulosamente feito pela ISO. Muitos outros padrões, incluindo quase todos os sem conexões, não estão listados.

A camada de aplicações contém não apenas um, mas toda uma coleção de protocolos para diversas aplicações.

O FTAM (*File Transfer, Access and Management*) fornece um meio de transferir, acessar e, em geral, manipular arquivos, remotos de maneira uniforme.

O protocolo MOTIS (*Message Oriented Text Interchange Systems*) é utilizado para correio eletrônico, sendo similar a série X.400 de recomendações do CCITT.

O protocolo VTP (*Virtual Terminal Protocol*) fornece uma forma independente de terminais para que um programa acesse terminais remotos (p.ex., para editores de tela completa).

O protocolo JTM (*Job Transfer and Manipulation*) é usado para submeter tarefas (*jobs*) para mainframes remotos, a fim de serem processados em lote. Ele pode ser utilizado para transferir tanto os programas quanto os arquivos de dados.

Também foram desenvolvidos inúmeros protocolos específicos para determinadas indústrias e para determinadas aplicações, e novos protocolos estão sendo concebidos e padronizados o tempo todo.

Tabela 1- Padrões de Redes ISO e CCITT

Camada	Padrão	Descrição
1-7	ISO 7498	Modelo de referência básico OSI da ISSO
7	ISO 8571	Serviço de Transferência, Acesso e Manipulação de Arquivos
	ISO 8572	Protocolo de Transferência, Acesso e Manipulação de Arquivos
	ISO 8831	Serviço de Transferência e Manipulação de Jobs
	ISO 8832	Protocolo de Transferência e Manipulação de Jobs
	ISO 9040	Serviço de Terminal Virtual
	ISO 9041	Protocolo de Terminal Virtual
	CCITT X.400	Manipulação de Mensagens (correio eletrônico)
6	ISO 8322	Serviço de apresentação baseado em conexões
	ISO 8323	Protocolo de apresentação baseado em conexões
5	ISO 8326	Serviço de sessão baseado em conexões
	ISO 8327	Protocolo de sessão baseado em conexões
4	ISO 8072	Serviço de transporte baseado em conexões
	ISO 8073	Protocolo de transporte baseado em conexões
3	CCITT X.25	Protocolo X.25 camada 3
2	ISO 8802	Redes Locais de Computadores
	CCITT X.25	Camada de enlace de dados HDLC/LAPB
1	CCITT X.21	Interface digital da camada física

Alguns dos principais padrões de redes ISO e CCITT. ISO se refere a um padrão internacional ISO. ISO 8802 é derivado do padrão IEEE 802 [TAN94].

4.3 A ARPANET

Precursora da Internet, ARPANET foi o fruto de um projeto desenvolvido pela ARPA (atualmente DARPA), a Agência para Projetos de Pesquisa Avançados (*Defense*) *Advanced Research Projects Agency*) do departamento de Defesa dos EUA.

Estabelecida em 1969 serviu de base para o desenvolvimento de novas tecnologias, ligando numerosas universidades e centros de pesquisa. Originalmente com quatro nós chegou a contar com centenas de computadores espalhados pelo mundo, do Haváí à Suécia. [TAN94]

Após a tecnologia da ARPANET ter-se revelado confiável, através de anos de funcionamento, foi formada uma rede militar, a MILNET, utilizando-se da mesma tecnologia. Também foi criada na Europa uma extensão da MILNET, chamada MINET. A MILNET e a MINET foram ligadas à ARPANET, sendo o tráfego entre as partes fortemente controlado. Duas redes por satélite, SATNET e WIDEBAND, também foram ligadas posteriormente. Dado que muitas das universidades e empreiteiras do governo na ARPANET possuíam suas próprias redes LAN's, eventualmente também estas foram conectadas aos IMPs, levando à ARPA internet, inter-rede da ARPA, com milhares de *hosts* e milhares de usuários.

A ARPANET antecede o modelo OSI em mais de uma década. O protocolo IMP-IMP na verdade corresponde a uma mistura dos protocolos das camadas 2 e 3. A camada 3 contém ainda um mecanismo de roteamento mais elaborado. Rigorosamente falando, esse mecanismo é mais uma camada de protocolo, o protocolo entre o IMP transmissor e o IMP receptor. No entanto esse protocolo não existe no modelo OSI.

O protocolo de rede, chamado de IP (*Internet Protocol*), é um protocolo sem conexões projetado para lidar com o vasto número de WAN's e LAN's que compreendem a Internet ARPA.

O protocolo de transporte da ARPANET é um protocolo baseado em conexões chamado de TCP (*Transmission Control Protocol*).

Não há protocolos das camadas de sessão e de apresentação na ARPANET, dado que ninguém precisou deles nos primeiros vinte anos de operação. Existem diversos protocolos de aplicação, mas estes não são estruturados como suas

contrapartidas na OSI. Os serviços da ARPANET incluem transferência de arquivos, correio eletrônico e *login* remoto. Estes serviços são oferecidos através dos notórios protocolos FTP (*File Transfer Protocol*), SMTP (*Simple Mail Transfer Protocol*), e TELNET (*login* remoto). Existem diversos outros protocolos especializados para outras aplicações. [TAN94]

A NSFnet substituiu a ARPANET como a principal rede governamental para conectar universidades e instituições de pesquisa. Entretanto em 1995, a NSF (*National Science Foundation*) extinguiu a NSFnet e a substituiu por um *backbone* comercial da Internet. Ao mesmo tempo, a NSF implementou um novo *backbone* de alta velocidade o vBNS (*Backbone Network Service*), o qual serve atualmente como base de pesquisas para desenvolver novas tecnologias da Internet.

4.4 MAP E TOP

MAP e TOP são redes voltadas para a área de automação industrial e de escritórios respectivamente. Foram desenvolvidas por grupos de empresas a fim satisfazer necessidades e interesses comuns.

O Protocolo para Automação da Fabricação, MAP (*Manufacturing Automation Protocol*), é uma rede projetada para otimizar a automação industrial baseada no padrão ISO. Foi desenvolvida pela GM e outras companhias da área da automação fabril. Esta rede utiliza o padrão *token bus* que fornece um desempenho determinístico.

Os Protocolos Técnicos e de Escritórios, TOP (*Technical and Office Protocols*) formam um conjunto de protocolos para a automação de escritórios. Desenvolvido pela Boeing, está baseado na LAN Ethernet em vez da *token bus*, dispensando o requisito de tempo real já que os Boeing's 747 não seguem uma linha de montagem.

A IBM, que também utiliza o protocolo TOP adotou uma terceira opção como o seu padrão de LAN, a *token ring*. A *token ring* é baseada em um protótipo desenvolvido nos laboratórios de pesquisa da IBM em Zurique. A IBM escolheu este projeto devido a sua alta confiabilidade e facilidade de manutenção, assim como outras vantagens técnicas.

Os três padrões de LAN's que foram aprovados estão descritos nos seguintes protocolos: IEEE 802.3 (baseado na Ethernet), 802.4 (*token bus*) e 802.5 (*token ring*).

Embora MAP e TOP difiram entre si nas suas camadas inferiores, a GM e a Boeing trabalharam juntas para garantir a compatibilidade nas camadas intermediárias e superiores.

Uma coleção de protocolos, um por camada, tal como MAP ou TOP, é chamada suíte de protocolos ou pilha de protocolos.

Tabela 2 - As suítes de protocolos TOP e MAP [TAN94]

	FTAM	DS	MNS	FTAM	DS	VTP	MNS
7							
6	Protocolo de Apresentação OSI (8823)			Protocolo de Apresentação OSI (8823)			
5	Protocolo de Sessão OSI (8327)			Protocolo de Sessão OSI (8327)			
4	Transporte Classe 4 Baseado em Conexões (8073)			Transporte Classe 4 Baseado em Conexões (8073)			
3	Modo sem conexões (8473)			Modo sem conexões (8473)			
2	Controle de Enlace Lógico (8802/2)			Controle de Enlace Lógico (8802/2)			
1		Token bus (8802/4)		Ethernet (8802/3)		Token ring (8802/5)	

Ambas as suítes seguem de perto o modelo OSI, como mostrado na tabela 2. O MAP usa o *token bus* como meio físico e o TOP usa a Ethernet (ou o *token ring*, que foi acrescentada em 1987). O serviço à camada de rede é fornecido em ambos os casos, usando na camada de enlace de dados o protocolo de enlace de dados IEEE 802.2 LLC (*Logical Link Control*) no modo sem conexões.

Ambas as suítes usam um protocolo de camada de rede sem conexões, o ISO 8473. Esse protocolo é muito próximo do protocolo IP da Internet ARPA, mas completamente diferente em estilo do protocolo CCITT X.25 utilizado nas redes públicas. A razão para esta escolha é que anos de experiência com a Internet ARPA mostraram que a abordagem por datagramas é muito mais flexível e robusta na conexão de múltiplas redes heterogêneas, um importante aspecto do MAP e do TOP.

A camada de transporte do MAP e do TOP é a ISO 8072/8073, usando o serviço classe 4. Essa classe assume que o serviço da camada de rede não é totalmente

confiável e lida ela própria com o controle de erros e de fluxo. Esta escolha permite ao MAP e ao TOP interligarem-se a praticamente a qualquer tipo de rede, não importando o quanto ela seja ruim. O preço pago, evidentemente, é uma camada de transporte complexa que deve lidar com um serviço de rede não confiável. A ARPANET, que funciona da mesma maneira, tomou essa decisão há anos, após observar que o serviço de rede não era confiável, e portanto o protocolo de transporte tinha que tratar disso, quer ele quisesse ou não. Ao contrário, o serviço X.25 é considerado altamente confiável, permitindo que se use com ele uma classe mais simples de serviço de transporte.

Os padrões OSI orientados à conexão ISO 8326/8327 e ISO 8822/8823 são usados nas camadas de sessão e apresentação do MAP e do TOP. Os padrões OSI são também utilizados na camada de aplicações, em particular o protocolo para transferência de arquivos, o protocolo de terminal virtual e outros.

Existem cinco tipos de sistemas que podem estar acima de uma rede TOP: um sistema final, um repetidor, uma ponte, um roteador e um *gateway*. Eles diferem quanto à camada na qual a interligação é feita.

A rede MAP tem seis tipos de nós, incluindo o sistema final MAP, a ponte, o roteador e o gateway. Ela não tem o repetidor (utilizando pontes com este fim), mas possui dois tipos de nós adicionais, o nó MINIMAP e o gateway MAP/EPA. Esses nós são compatíveis com o padrão LAN's PROWAY, comum em ambientes fabris antes do aparecimento do MAP. Eles possuem apenas as camadas 1 e 2, e são importantes para trabalhos em tempo real crítico. [TAN94]

4.4.1 Mini - MAP

A arquitetura Mini-MAP é uma redução da funcionalidade do MAP, com o objetivo de satisfazer as necessidades dos processos industriais que exigem tempos de resposta rápidos e soluções mais econômicas para o nível baixo do processo de automação.

A principal diferença entre Mini-MAP e Full-MAP não é apenas na funcionalidade e realização das camadas da OSI, mas também na conexão com o meio. Mini-MAP dispõe apenas de transmissão *carrierband* e fornece uma funcionalidade limitada comparada com Full-MAP. Além disto, permite um limitado tamanho de

mensagem e não fornece controle de fluxo. Um *gateway* MAP/EPA (*Enhanced Performance Architecture*) conecta Mini-MAP e Full-MAP. [BEN93]

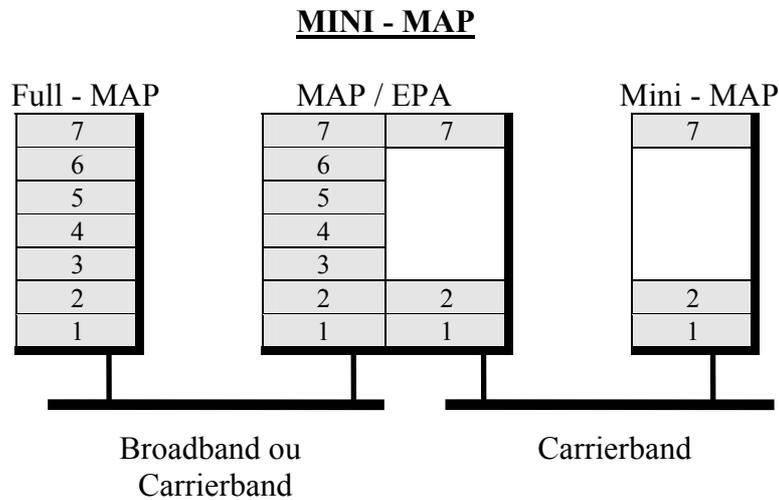


Figura 2. MAP, EPA e Mini-MAP [BEN93]

4.5 A USENET

Quando o uso do sistema UNIX se difundiu amplamente na Bell Laboratories, os pesquisadores perceberam rapidamente que precisavam de algum meio para copiar arquivos de um sistema UNIX para outro. Para resolver este problema eles escreveram o programa *uucp* (UNIX to UNIX CoPy).

À medida que os sistemas UNIX adquiriram modems e discadores de telefone automáticos, tornou-se possível copiar arquivos automaticamente entre máquinas remotas, usando o *uucp*. Surgiram diversas redes informais que cresceram muito rapidamente porque tudo que era necessário para se juntar à rede era ter um sistema UNIX e um modem, coisa que praticamente todos os departamentos de computadores das universidades possuíam. Esta rede não possui uma administração central. Cada instalação pode executar a versão de *software* que quiser.

O segmento europeu, a EUNET, e os EUA estão conectados através de uma linha arrendada, entre Amsterdã e uma instalação na Virgínia chamada **uunet**, que também está na ARPANET. Existem também segmentos no Japão, Coréia, Austrália e outros países.

O único serviço oferecido pela **uunet** é o correio eletrônico, mas uma rede associada chamada **USENET**, iniciada na Duke University e na University of North Carolina, também oferece um serviço de informações chamado *network news* ou seja, notícias da rede. Muito embora alguns computadores UNIX nos EUA estão apenas em uma ou em outra, a maioria está em ambas as redes. As notícias da rede são divididas em centenas de grupos em uma variedade de tópicos, alguns técnicos, outros não. Os usuários da USENET podem ter assinatura de qualquer grupo no qual estejam interessados. [TAN94]

Na Europa, a EUNET carrega tanto correio eletrônico quanto notícias, e portanto há apenas uma rede.

4.6 CSNET

O enorme valor da ARPANET para a comunicação entre pesquisadores já tinha se tornado óbvio por volta de 1980. O problema principal da ARPANET é que ela pertencia ao Departamento de Defesa, e as universidades que não tinham contratos com o governo federal não tinham permissão para usá-las. Para fornecer um serviço de rede para a comunidade como um todo, a NSF (*National Science Foundation*) organizou a **CSNET**, que era acessível a todos os departamentos de computação nos EUA.

Na verdade, a CSNET não é uma rede verdadeira como a ARPANET, mas uma metarrede. Ela usa as facilidades de transmissão fornecidas por outras redes e acrescenta uma camada uniforme de protocolo por cima para fazer com que o conjunto todo pareça para o usuário como uma única rede lógica. Todas as máquinas são interconectadas pela máquina CSNET-RELAY, situada em uma companhia chamada BBN em Cambridge, Massachusetts.

A primeira componente é a ARPANET. A segunda componente é uma rede pública X.25. Os departamentos que estão em uma das redes X.25, tais como a Telenet ou a Uninet, podem usar essa rede para chegar ao CSNET-RELAY.

Para aqueles departamentos que não estão na ARPANET nem se encontram em uma rede X.25, foi montada a **PHONENET**. Esta rede consiste em fazer com que os hosts PHONENET simplesmente chamem o CSNET-RELAY quando quiserem.

A quarta componente, chamada **CYPRESS**, é uma tentativa de duplicar a tecnologia da ARPANET com um orçamento menor. Ela tem nós comutadores de pacotes, chamados IMPLET's, e *hosts*. Os IMPLET's são MicroVaxes da DEC, ligados por linhas arrendadas. Para juntar-se à CYPRESS, uma organização deve comprar um MicroVax e arrendar uma linha até um outro IMPLET.

O serviço básico oferecido a todos os nós CSNET é o correio eletrônico, usando os protocolos e formatos da ARPANET. Com exceção dos *hosts* na PHONENET, é possível também fazer transferência de arquivos e *login* remoto. [TAN94]

4.7 BITNET

Uma outra rede interessante é a **BITNET** (*Because Its Time NETwork*), iniciada em 1981 pela City University de Nova Iorque, e pela Yale University. A idéia básica era criar uma rede universitária, como a CSNET, mas para todos os departamentos, não só os de computação.

Tecnicamente, a BITNET é um tanto parecida com a CYPRESS, com cada nó arrendando linhas até algum outro nó. Ao contrário da CYPRESS, os próprios *hosts* fazem a comunicação; não há sub-rede de IMPLET's. Também ao contrário da CYPRESS, a BITNET usa protocolo e *software* doados pela IBM, que não são compatíveis nem com a OSI nem com TCP/IP (ou com qualquer outra coisa), pois se baseia na idéia de transmitir imagens de cartões perfurados de oitenta colunas, uma fonte freqüente de problemas.

O financiamento da BITNET é incomum, o que explica parte de sua popularidade. Para filiar-se a ela, uma universidade deve arrendar uma linha até algum outro nó da BITNET e pagar pelo aluguel dessa linha, e também deve permitir que alguma outra universidade se ligue a ela através de uma linha arrendada (às custas da outra universidade) em algum momento no futuro. Finalmente, ela deve também concordar em passar adiante o tráfego que flui através dela, sem cobrar nada. Assim, exceto pela perda de algum poder computacional causado pelo processamento do tráfego de terceiros para passá-lo adiante, o único custo real é uma linha alugada. Ao

contrário de praticamente todas as outras redes, não há encargos baseados no volume de tráfego transmitido.

O serviço básico da BITNET é a transferência de arquivos, o que também inclui correio eletrônico e a submissão remota de tarefas. Cada arquivo que entra no sistema contém o seu destinatário final, e poderá ser armazenado e passado adiante muitas vezes antes de chegar lá. Uma capacidade mínima de login remoto é possível, mas como o tráfego interativo é armazenado e passado adiante como qualquer outro tráfego de transferência de arquivos, o tempo de resposta é muito lento e a confiabilidade é baixa. [TAN94]

Uma nova versão da BITNET, denominada BITNET-II, permite a transferência de mensagens e arquivos via Internet.

4.8 SNA

Não há discussão completa sobre redes que não contenha ao menos algumas palavras sobre a arquitetura de redes da IBM, a SNA (*System Network Architecture*). O modelo OSI foi moldado em boa parte conforme a SNA, incluindo o conceito de camadas escolhido e as suas funções aproximadas.

A SNA é uma arquitetura de redes voltada para permitir que clientes IBM construam suas próprias redes privadas, tanto os hosts quanto sub-redes.

Uma rede SNA consiste em um conjunto de máquinas chamadas **nós**, de quatro tipos caracterizados aproximadamente como se segue. Os nós do Tipo 1 são terminais. Os nós do Tipo 2 são controladoras, máquinas que supervisionam o comportamento de terminais e outros periféricos. Os nós de Tipo 4 são processadores *front-end*, dispositivos cuja função é aliviar a CPU principal do trabalho e a gerência das interrupções associados à comunicação de dados. Os nós de Tipo 5 são os *hosts* principais propriamente ditos mas, com o advento de microprocessadores de baixo custo, algumas controladoras assumiram propriedades similares a os *hosts*. Por mais estranho que pareça, não existem nós do tipo 3.

5. PROJETO INDUSTRIAL

5.1 INTRODUÇÃO

Nos últimos anos vem se produzindo uma mudança radical no projeto e fabricação de todos os produtos industriais, desde os automóveis aos sapatos e brinquedos, passando pelas peças de confecção. A automação no ambiente industrial proporciona uma melhoria nos mecanismos de produção através da aplicação do computador no controle de equipamentos e processos envolvidos.

O termo *processo* e *processo técnico* pode ser definido(veja figura 3), como: *Um conjunto de eventos de um sistema interativo que transformam, transferem ou armazenam matéria, energia ou informação. As variáveis físicas do processo técnico podem ser adquiridas e modificadas por meios técnicos.* [BEN93]



Figura 3. Processo Técnico [BEN93]

Um processo industrial se caracteriza por contar com fluxos de produtos, materiais e informação. Existem três diferentes processos envolvidos [BEN93] (veja figura 4):

- *processos técnicos* servem para o primeiro propósito de uma empresa, e.g., fabricação de produtos, produção de energia, teste de peças e armazenagem de produtos;

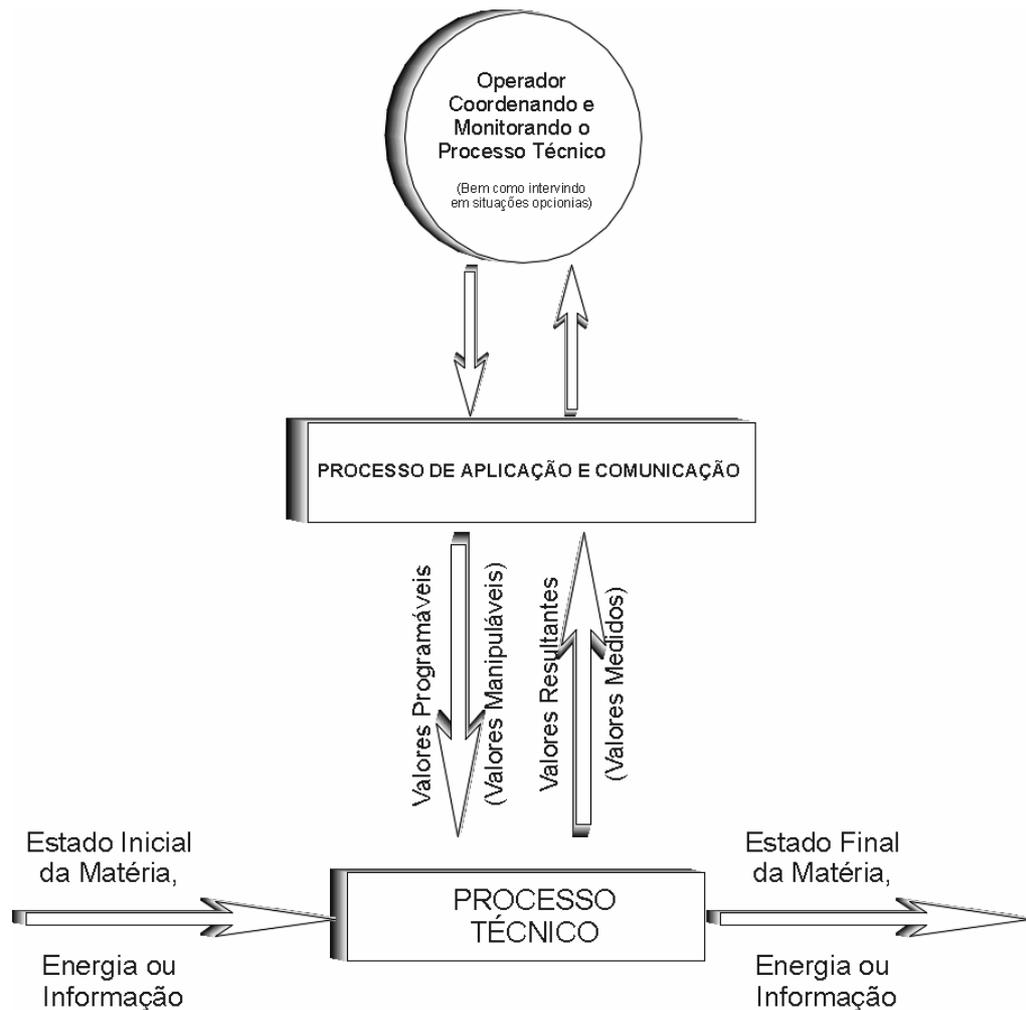


Figura 4. Tipos de Processos [BEN93]

- *processos de aplicação* constituídos por sistemas de aquisição, processamento, exibição e saída da informação. Usualmente composto de processos de controle e componentes microeletrônicos programáveis, e.g., controladores programáveis;
- *processos de comunicação* na ciência da computação são procedimentos dinâmicos que intercambiam dados entre as partes através de processamento de *programas de comunicação*. O conjunto de processos de comunicação interage via uma linha comum (cabo, fibra ótica, etc.).

Por sua vez os processos técnicos podem ser classificados considerando os tipos de procedimentos que prevalecem [BEN93]:

- *processos de fluxo* (processos contínuos), e.g., plantas de produção de energia, processos químicos, indústrias de aço. A característica principal dos

processos contínuos é que o fluxo de material não é interrompido durante a ocorrência do processo;

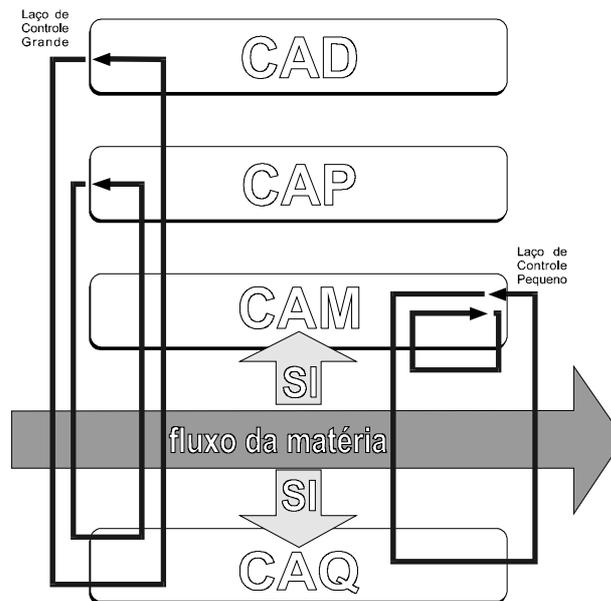
- *processos sequenciais* (processos em lote), e.g., processos de carga, processos de teste, processos de produção, processos de início e fim;
- *processos de tipo discreto* (produção de componentes discretos), e.g., processo de armazenagem, processo de transporte, processo de circulação. Nos processos discretos, o fluxo se caracteriza por ser composto de elementos unitários na saída do sistema, isto é, são peças montadas que saem de um processo de produção e servem como entrada para outro.

O controle de processos tem por objetivo automatizar os processos contínuos (normalmente indústrias químicas e de aço). Por sua vez, os processos discretos incluem os sistemas de manufatura nos quais pode haver vários equipamentos envolvidos tal como sensores, atuadores, controladores programáveis (CP's), controladores robôs, controladores numéricos computadorizados (CNC's), controladores lógicos programáveis (CLP's), sistemas de transporte de material, sistemas de visão, computadores, etc.

5.1.1 Processos Discretos

Os sistemas de produção discretos são compostos por um grande número de equipamentos diferentes, interconectados para processamento de material. Os equipamentos controlados por computador, são agrupados em conjuntos, sendo que cada um deles é responsável pela execução de parte da tarefa de produção de um produto. Cada um destes conjuntos, chamados células, é interconectado com o resto do sistema através de um sistema de transporte de materiais e peças parcialmente montadas. Além disto, várias informações devem fluir entre as células; são dados referentes à integração das diversas partes componentes do sistema. As redes de computadores são as responsáveis pela interligação das células para permitir a coordenação, monitoração, controle e gerência de todo o sistema, além da transferência de informações.

Mas uma empresa é mais do que a fabricação. A área de fabricação deve ser flexível para se adaptar a área de comercialização (*marketing*). Isto significa que em cada fase de produção, o fluxo de informação entre as partes envolvidas deve ser assegurado. A figura 5 mostra um laço pequeno e outro grande de controle de produção. Estes laços de controle precisam de um fluxo de dados da área de produção para o desenvolvimento. Isto interessa ao fluxo de produto e ao controle de qualidade, tanto quanto aos departamentos de compra e venda.



CAD: (*Computer Aided Design*)

CAP: (*Computer Aided Planning*)

CAM: (*Computer Aided Manufacturing*)

CAQ: (*Computer Aided Quality Assurance*)

SI: (*Sensory Information*)

Figura 5. Laços de Controle na Produção [BEN93]

Em esse tipo de sistemas, as possibilidades que oferece o tratamento das informação mediante computador são utilizadas, não somente para controlar o processo de fabricação, se não também para controlar o conjunto do processo de produção, considerado como um sistema fechado. Para lograr tal fim o sistema se vale de computadores situados nas diversas áreas relacionadas com o processo de produção;

desde a planificação da produção, passando pelo desenho e fabricação do produto, até as provas para assegurar a qualidade do mesmo.

A flexibilidade se consegue por meio do acoplamento e sincronização do fluxo de dados e informação com os meios automatizados de fabricação, transporte e armazenamento, através de redes locais LAN (*Local Area Network*), que permitem realizar a conexão, via computadores, entre diversos subsistemas.

A fim de organizar o constante crescimento do grau de automação nos processos, na produção e na engenharia se faz necessário o estabelecimento de uma organização hierárquica.

Uma representação hierárquica implica em uma representação em camadas onde as atividades são tratadas na forma de serviços requisitados por um nível mais alto da hierarquia ao seu nível imediatamente inferior.

Estes níveis estão associados às funções específicas do processo de produção e apresentam requisitos diferentes, no meio de transmissão (cabo, fibra ótica, rádio), no método de transferência utilizado (protegido/não protegido, banda base/banda larga) e no desempenho.

Um conceito de rede que inclui os sistemas de comunicação envolvendo todos os níveis de automação está representado pelo projeto de Manufatura Integrada por Computador - CIM (*Computer Integrated Manufacturing*).

5.2 CIM - FABRICAÇÃO INTEGRADA POR COMPUTADOR

O objetivo do CIM é a automação do fluxo de informação na organização global da produção, desde a entrada de uma ordem, ao longo de todas as etapas do processo, até o embarque e serviço pós-venda do produto fabricado.

Como se pode observar, se trata de uma nova concepção da fabricação que envolve conceitos tão novos como: controle adaptativo de processos, robótica, controle numérico, inteligência artificial, sistemas inteligentes, sistemas flexíveis de fabricação, tecnologia de bases de dados e transmissão da informação, gestão de dados técnicos, planejamento de processos, estoque e manipulação automática de materiais, programação e controle de manutenção, etc.

Os benefícios potenciais da adoção das técnicas CAD/CAM/CIM/CAE se pode simplificar da seguinte forma:

- Notável aumento da produtividade.
- Diminuição do custo da produção.
- Adaptabilidade rápida da produção às flutuações e exigências do mercado.
- Melhora considerável da qualidade e confiabilidade da produção.
- Promoção profissional do técnico e operário, e utilização mais eficiente da maquinaria.

A estrutura funcional dos Sistemas de Produção Industrial segue geralmente uma organização hierárquica em níveis associados a funções específicas do processo de produção, com requisitos distintos de comunicação, tais como: volume e tamanho da informação e tempo de resposta, na integração dos elementos dos diversos níveis.

Dependendo do nível considerado, os requisitos de comunicação são diferentes conforme pode se verificar na figura 6. As mensagens trocadas são dos mais diversos tipos: arquivos gráficos, arquivos de dados, programas, comandos, respostas a comandos, alarmes, sinais de sincronização, dados de medidas de atuadores e sensores, dados de alarmes, valores de referência, etc. Os tempos de respostas do sistema de comunicação variam, consoante nível e tipo de mensagem, desde algumas dezenas de micro-segundos para dados de tempo real (alarmes, comandos, etc.), passando por dezenas de segundos para mensagens em sistemas de controle e supervisão, até centenas de segundos para arquivos de dados. A frequência de troca de mensagens variará muito, sendo muito alta nos níveis baixos e rara ou pouco frequente em outros níveis. O comprimento das mensagens irá de alguns bytes até dezenas de K-bits.

A associação destes requisitos de comunicação às soluções de redes locais correspondentes é geralmente representada por uma estrutura em níveis, conforme indicada na figura 2.2, caracterizando o seguinte âmbito de atuação:

<div style="display: flex; justify-content: space-between;"> Característica Tipo de mensagem </div>	Arquivos Gráficos	Arquivo de Dados	Programas de Controle Numérico	Sinais de Sincronismo	Sinais com Valores Nominais e Atuais	Mensagens de Eventos
Atraso Permitido	 1-100s	 1-100s	 1-100s	 1-100ms	 20-100ms	 0.1-80ms
Tamanho da Mensagem	 >10 Kbits	 1-10 Kbits	 >10 Kbits	 8-64 bits	 <10 Kbits	 8-64 bits
Frequência de Ocorrência	 Raro	 Muito Raro	 Muito Raro	 Muito Frequente	 Frequente	 Raro
Classificação	Mensagens de Tempo Não Crítico			Mensagens de Tempo Crítico		

Figura 6. Tipos de Mensagens e Características Técnicas do Dados [BEN93]

5.2.1 Nível de campo

O nível mais baixo na hierarquia de automação é o nível de campo que inclui os dispositivos de campo próprios ao processo, tal como sensores e atuadores. A função destes dispositivos é transferir dados entre o produto e o processo técnico. O produto envia dados adquiridos por meio de medidas. Os valores medidos são a base para a intervenção no processo técnico ou de controle. Através da medição e controle, as ferramentas de produção envolvidas podem ser perfeitamente monitoradas e controladas.

O tipo de medida e controle difere dependendo do tipo de processo técnico (contínuo, discreto). Pode ser digital ou analógico. Os valores medidos podem estar disponíveis por um curto período de tempo (dinâmico) ou por um longo período de tempo (quase dinâmico). Estes valores servem como entradas para algumas funções tais como:

- visualização
- armazenagem
- documentação
- conversão
- controle

5.2.2 Nível de célula

O termo célula tem origem na estrutura de produção na flexível manufatura, FMS (*Flexible Manufacturing System*) na qual tarefas independentes são executadas em subproduções simplificadas autônomas, que pode ser uma manufatura ou uma célula montada. Devido a flexibilidade de configuração das células dependendo das necessidades específicas do produto, o sistema pode ser facilmente adaptado a manuseio de produtos e mudanças de quantidade.

Num processo contínuo, a célula corresponde a uma unidade operacional. No entanto há mais conexões cruzadas devido à produção de fluxo contínuo. Processos em lote se aproximam mais do modelo pois ele permite produção de produtos de alta qualidade, mesmo em pequenas quantidades.

No nível de célula, o fluxo de informação consiste principalmente da carga de programas, parâmetros e dados. Em processos com tempos de espera de máquina pequenos isto é feito durante o processo de produção. Em controladores de pequeno porte pode ser necessário carregar subrotinas durante um ciclo de manufatura. Isto determina os requisitos de tempo e não do carácter da informação a ser transferida. Além disso, sincronização da máquina e manuseio de eventos podem precisar tempos curtos de resposta no barramento. Este requisito de tempo real não é compatível com excessivo tempo de transferência dos programas aplicativos, assim se faz necessária uma segmentação de mensagens.

5.2.3 Nível de controle de processo

O nível de controle de processo, também chamado *planta de supervisão*, consiste de células combinadas em grupos que monitoram os processos técnicos. Células são subredes projetadas com uma funcionalidade orientada-à-aplicação.

O processo técnico é uma subunidade da planta de produção que pode consistir em vários processos técnicos similares ou não. Um processo técnico pode ser dividido em *on-sites* e *off-sites*. *Off-sites*, por exemplo, pode ser o suprimento de energia em algum processo técnico.

A descrição do fluxo de informação será limitada a um processo técnico e o exemplo de intercâmbio de dados com o processo operador será utilizado. O termo *processo operador* é derivado dos processos de engenharia onde o operador (humano) era o agente central assegurando as propriedades das funções das plantas pelo controle e intervenção quando necessário no processo técnico. Hoje, esta intervenção está limitada à programação de parâmetros, ligar e desligar máquinas e a atividades de emergência. O seguinte exemplo ilustra como influi a tradicional semelhança entre os diferentes setores da automação moderna até a tecnologia de comunicação digital.

Quando os processos de automação ainda não eram conhecidos, os sinais de processamento eram transmitidos por meio de dispositivos de automação diretamente ao nível de controle, apesar de que o sinal não era mostrado em quadros mas em visores. Desta forma, a informação de entrada podia ser digitalizada. O propósito de tal apresentação é para limitar a extensa informação a uma informação global e

condensada. No futuro, sistemas de otimização e diagnóstico terão considerável influência no fluxo da informação no nível do processo de controle.

Tal fluxo de informação no nível de controle pode ser realizado através de transferência acíclica de dados, a qual é atualmente prática comum nas redes MAP. No nível de controle, a escolha entre cíclica e/ou acíclica transferência de dados será menos importante no futuro que a preparação do tipo de dado.

5.2.4 Nível de Gerenciamento

Inclui gerência de produção e de informação que são interconectadas com as funções administrativas.

No Nível de Gerenciamento ocorrem uma série de ações de transmissão de dados, em particular no acompanhamento de todas as seqüências de produção no CAM (Comandos de Controle e Programas NC, aquisição de dados operacionais, fluxo de material) realimentando o sistema CAP. No caso de sistema complexos poderão existir diversos computadores executando essas funções, alimentando, em particular, a base de dados comum de planejamento de produção.

5.2.5 Nível de Planejamento

No nível de planejamento são elaboradas as especificações do produto a ser fabricado partindo da planificação geral da produção da fábrica e das possibilidades da mesma. As operações de definição e preparação, prévias a todo o processo de fabricação, se efetuam nos modernos sistemas automatizados, utilizando técnicas de engenharia assistidas por computador, CAE (*Computer Aided Engineering*). As funções a realizar são, basicamente duas: o desenvolvimento e projeto do novo produto, por meio de técnicas de desenho assistido por computador, CAD (*Computer Aided Design*), e a preparação de toda a documentação necessária para efetuar o processo de fabricação, que se realiza utilizando técnicas de planificação assistidas por computador, CAP (*Computer Aided Planning*).

Mediante as técnicas CAD é desenhado, através de diversos esquemas e cálculos, o novo produto, gerando-se planos do mesmo, listas de componentes para cada uma de suas partes, assim como uma base de dados onde ficam incluídos os dados geométricos e tecnológicos necessários dos diversos componentes, para sua eventual

utilização na tarefa de projetar o programa de montagem de novos elementos. Pode-se obter também automaticamente representações gráficas tridimensionais, cortes, etc.

Na planificação da fabricação, CAP, é elaborado, por meio do computador, todos os dados e documentação necessários para realizar um determinado tipo de fabricação: planos de trabalho, planos de montagem, planos de controle de qualidade, programas de controle numérico para as máquinas e robôs, etc. Como ponto de partida se utiliza a base de dados gerada pelo sistema CAD, onde se dispõe de todos os dados e informação gráfica necessários. A mesma base se utiliza também para gerar os dados que tem servido para executar os programas de controle e segurança de qualidade, CAQ (*Computer Aided Quality*).

O controle do programa de produção, PPC (*Production Program Control*) tem como missão planificar as necessidades de material e peças, realizar a gestão do tempo para os pedidos da fabricação e lograr o ajuste entre a capacidade de produção disponível e a requerida para atender à produção solicitada. Os dados necessários para implementar o PPC, tais como listas de componentes, planos de trabalho, etc., são facilitados pelos sistemas CAD e CAP, realizando-se o intercâmbio de dados tanto em forma direta como através de uma base de dados comum.

O sistema de controle e segurança da qualidade (CAQ) está coordenado com os sistemas CAD, CAP e a própria fabricação, formando um verdadeiro sistema de inspeção de qualidade controlado por computador. O sistema gera planes de fabricação, planes de provas ou verificação e programas de provas, para ser realizados em estações automáticas de medida. Os dados obtidos são utilizados nos processos de desenvolvimento e desenho para introduzir as modificações adequadas no caso de ser detectados defeitos de fabricação.

Finalmente, resta considerar o sistema de fabricação propriamente dito. Em um sistema moderno, a automação abarca o processo de transporte, armazenagem, mecanizado o conformado, montagem e expedição do produto. Entre todos eles deve-se estabelecer um intercâmbio de informação que possibilite a execução do processo de fabricação de forma logística fluída.

O controle e supervisão das tarefas que constituem o processo de fabricação se realizam por meio de técnicas CAM (*Control Aided Manufacturing*). Para isto se

utiliza um computador de controle da fabricação, que trabalha em linha (*on-line*) unido ao processo através de estações de dados.

O sistema de CAM serve, entre outras coisas, para elaborar os programas de produção diários ou para cada turno. Se organiza o trabalho de máquinas e instalações, se comprova a disponibilidade de materiais e meios de produção, gestionando os pedidos de armazenamento e transporte em função da programação estabelecida, se controla e supervisa a fabricação e a montagem nas diversas áreas de produção, etc.; tudo isto, a partir dos dados recebidos pelo computador de fabricação, desde o sistema PPC, para atender aos pedidos planificados.[MON86]

6. O PROTOCOLO PROFIBUS

PROFIBUS é uma entre as diversas redes proprietárias existentes na área de *fieldbus*. A utilização de uma destas redes resulta normalmente em soluções incompatíveis entre si, isolando a mesma de outros provedores. Para satisfazer a necessidade de um sistema de comunicação aberto, independente do vendedor é que surge a especificação e padronização do PROFIBUS.

Para descrever o protocolo completamente e de forma clara algumas definições se fazem necessárias. O resultado é um padrão complexo que a primeira vista parece difícil de entender. Neste capítulo serão descritas sucintamente as propriedades características do PROFIBUS. O leitor pode obter informações técnicas detalhadas na Norma Alemã PROFIBUS, DIN 19 245.

6.1 PROPRIEDADES BÁSICAS

O protocolo PROFIBUS (*PROcess FIEldBUS*) é o resultado de um projeto *Fieldbus* financiado pelo *German Federal Ministry of Research and Technology* (BMFT). Junto com a iniciativa de 13 fabricantes e 5 institutos técnico/científicos o protocolo PROFIBUS foi especificado, testado em campo e normalizado [PRO92].

O PROFIBUS é um barramento de chão-de-fábrica (*fieldbus*). É um barramento, pois neste protocolo todas as mensagens que circulam pelo meio físico são ouvidas por todas as estações presentes. É de chão-de-fábrica, pois o objetivo é interconectar dispositivos de campo na faixa de desempenho baixa (nível sensor/atuador) até a média (nível de célula). Com PROFIBUS, todos os tipos de componentes de automação podem intercambiar dados. PLC's, PC's, painéis operadores e monitores, assim como sensores e atuadores podem comunicar-se, através da mesma interface, numa larga faixa de aplicações conforme a figura abaixo.

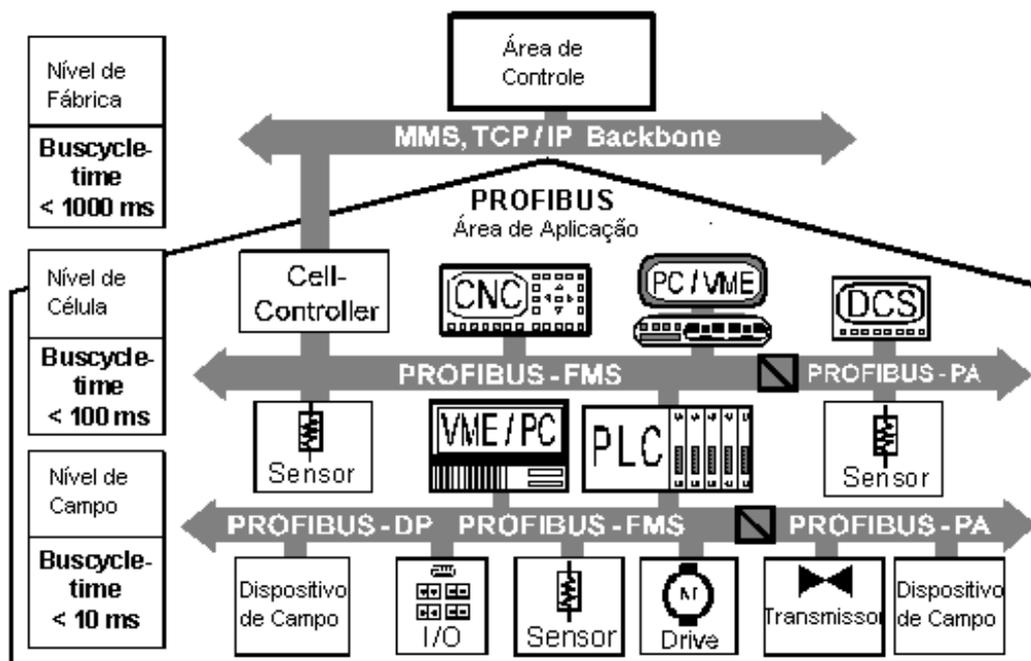


Figura 7. PROFIBUS: Faixa de Aplicações

O PROFIBUS apresenta as seguintes características, próprias de uma rede chão-de-fábrica:

- Possibilidade de alcançar distâncias elevadas (até alguns quilômetros);
- Mensagens curtas (até algumas centenas de *bytes*);
- Tempo real / Determinístico;
- Elevada imunidade à interferência eletromagnética;
- Possibilidade de entrada e saída de estações *on-line*.

6.2 ARQUITETURA DO PROTOCOLO PROFIBUS

O protocolo PROFIBUS está baseado numa variedade de normas internacionais existentes. A arquitetura do protocolo segue o modelo OSI (*Open Systems Interconnection*) de acordo com o padrão internacional ISO 7498 [DIN91]. A arquitetura do protocolo PROFIBUS-FMS e do protocolo PROFIBUS-DP é mostrado na figura 8.2.

As duas variações utilizam o mesmo protocolo de acesso ao meio (Camada 2) e a mesma técnica de transmissão (Camada 1).

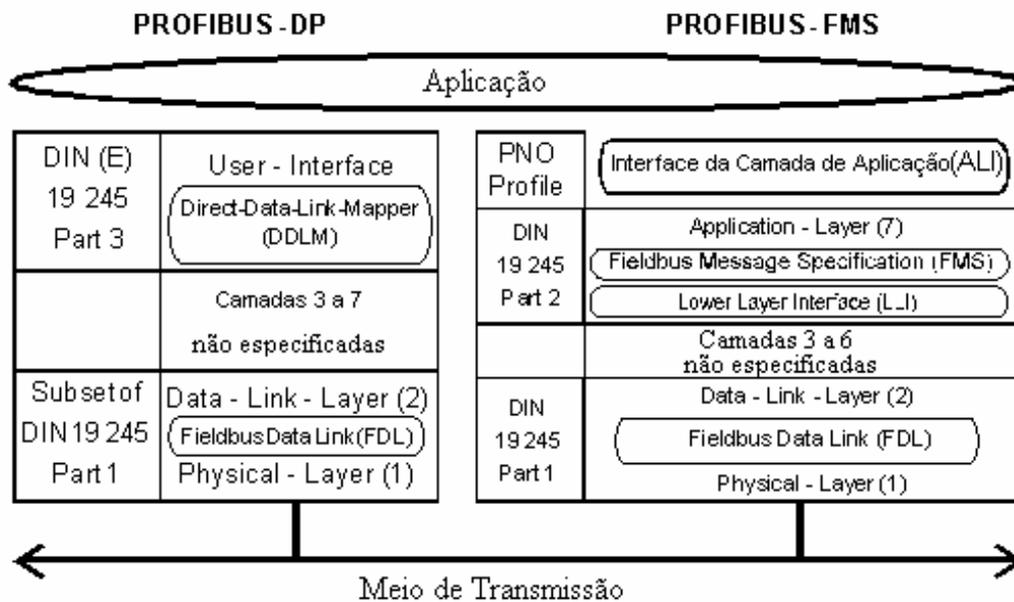


Figura 8. A arquitetura do protocolo PROFIBUS-FMS e PROFIBUS-DP

O PROFIBUS-FMS (*Fieldbus Message Specification*), é uma solução universal para tarefas de comunicação a nível de célula e a nível de campo na hierarquia da comunicação industrial, esta versão está incluída no padrão europeu EN 50710 [VOL97].

No PROFIBUS-FMS as camadas 3 a 6 não estão especificadas. A função destas camadas necessária para a aplicação se encontram na LLI (Lower Layer Interface) que faz parte da camada 7. O FMS contém o protocolo de aplicação e oferece a interface com o usuário além de fornecer poderosos serviços de comunicação.

Os serviços do FMS estão definidos como um subconjunto das funções do MMS (*Manufacturing Message Specification, ISO 9506*) do protocolo MAP. As funções do MMS são otimizadas conforme os requisitos de cada rede. Entre os serviços também estão definidas as funções para a administração da comunicação de objetos e gerenciamento de rede.

O PROFIBUS-DP (*for Decentral Periphery*) é a versão otimizada, especialmente dedicada a comunicações de tempo crítico entre sistemas de automação e periféricos distribuídos. Esta versão também está incluída no padrão europeu EN 50710 [VOL97].

No PROFIBUS-DP as camadas 3 a 7 não são utilizadas. A camada de aplicação (7) é omitida para alcançar um melhor desempenho. O DDLM (*Direct Data Link Mapper*) mapeia as funções da camada 2 para a Interface-Usuário. As funções de aplicação acessíveis ao usuário estão disponíveis na Interface-Usuário que especifica o sistema e as características dos vários tipos de dispositivos PROFIBUS-DP.

O PROFIBUS-PA (*for Process Automation*) é a terceira versão da família PROFIBUS, utilizada para conectar processos de automação e dispositivos de campo descentralizados. Está baseado nos resultados do ISP (*Inter-Operable Systems Project*) e tem sido adotado pela PNO (*PROFIBUS Nutzerorganisation - PROFIBUS Users Organization*) [VOL97].

O PROFIBUS-PA permite a alimentação das estações via barramento e o intercâmbio de dados num mesmo par de fios. PA pode ser utilizado em áreas com alto risco de explosão utilizando tecnologia de transmissão com segurança intrínseca de acordo com IEC 1158-2 [VOL97].

6.3 A CAMADA FÍSICA (1)

A área de aplicação do sistema chão-de-fábrica é afetada substancialmente pelo meio de transmissão e pela interface física do barramento. Além do requisito de integridade dos dados o custo de instalação da fiação tem um significado crítico. A norma PROFIBUS define um único protocolo de acesso ao meio para diferentes técnicas de transmissão:

Fio de cobre: Esta versão é definida como a versão base das técnicas de transmissão para aplicações em manufatura, automação predial e controle conforme o padrão americano EIA RS-488. Utiliza dois condutores par trançado, com blindagem opcional. A pinagem do conector e as correspondentes ligações são mostradas na figura abaixo.

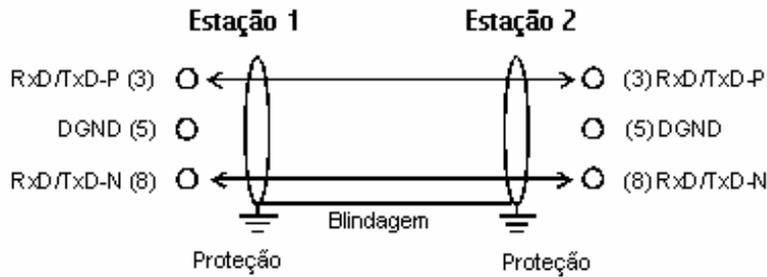


Figura 9. Conexão do PROFIBUS-DP e PROFIBUS-FMS

Fibra ótica: A PNO (*PROFIBUS Nutzerorganisation - PROFIBUS Users Organization*) está preparando a especificação para a técnica de transmissão com fibra ótica com o objetivo de estender as aplicações para altas taxas de transmissão e também para ambientes ruidosos. Esta especificação está disponível como uma proposta [VOL97].

Segurança intrínseca: O PROFIBUS-PA permite segurança intrínseca nas transmissões, adequado para ambientes com risco de explosão, e alimentação das estações via barramento. Esta técnica de transmissão está especificada na norma internacional IEC 1158-2 [VOL97].

A seguir estão listadas as características da camada física do PROFIBUS de acordo com a norma DIN 19245 parte 1 [DIN91]:

- Topologia da rede: barramento linear, terminado em ambas as extremidades, *stubs* $\leq 0,3$ m;
- Meio físico: par trançado blindado(24 AWG¹). Blindagem pode ser dispensada dependendo do ambiente;
- Padrão elétrico: EIA 485;
- Conector trapezoidal subminiatura de 9 pinos;
- Comunicação serial assíncrona *half-duplex*;
- Codificação NRZ (*non return to zero*);
- Número de estações: 32 em cada segmento sem repetidores. Com repetidores pode ser estendido até 127;

- Número máximo de repetidores: 3.

Tabela 3 - Taxas de transmissão x Comprimento máximo do barramento no PROFIBUS.

Taxa de transmissão [Kbits/s]	Comprimento máximo do barramento[m]
9,6 / 19,2 / 93,75	1200
187,5	600
500	200

6.4 A CAMADA DE ENLACE DE DADOS (2)

A segunda camada do modelo de referencia OSI fornece as funções de controle de acesso ao meio, integridade dos dados assim como executa o protocolo de transmissão e mensagens.

No protocolo PROFIBUS a camada dois é denominada FDL (*Fieldbus Data Link*). Esta camada é dividida em duas subcamadas, LLC (Logical Link Control) e MAC (*Medium Access Control*) [AGU89].

6.4.1 Transmissão de mensagens

A camada 2 do PROFIBUS opera em modo sem conexão. Oferece transmissões ponto-a-ponto e multi-ponto (*broadcast* e *multicast*)

Comunicação *broadcast* significa que a estação ativa envia uma mensagem sem confirmação para todas as outras estações (mestres e escravas).

Comunicação *multicast* significa que uma estação ativa envia uma mensagem sem reconhecimento para um grupo de estações mestres ou escravas.

A troca de mensagens é feita por ciclos. O ciclo de mensagem é composto pela fase SEND_REQUEST da estação ativa seguida de ACK_RESPONSE por uma outra estação ativa ou passiva. O ciclo só é interrompido quando da transmissão do bastão para a próxima estação ou no caso de transmissões *broadcas* ou *multicastt*. A resposta ACK de estações é feita, em geral, com serviços imediatos, na janela de tempo (*slot-time*) administrada pela estação iniciadora.

A subcamada LLC (*Logical Link Control*) fornece três serviços acíclicos e um serviço cíclico, descritos a seguir:

¹ American Wire Gauge

• **Serviços acíclicos:**

- **SDA** (*Send Data with Acknowledge*). Envio de informações com reconhecimento, permite ao usuário local enviar informações para um (único) usuário remoto, com confirmação de recebimento pelo usuário remoto. No período compreendido entre uma transmissão e a espera pelo reconhecimento (*slot-time*), nenhuma outra transmissão pode ocorrer.

- **SRD** (*Send and Request Data with Reply*). Intercâmbio de informação, permite ao usuário local enviar e ler informações para/de um usuário remoto, simultaneamente.

- **SDN** (*Send Data with No acknowledge*). Envio de informações sem reconhecimento, permite ao usuário local enviar informações para um usuário remoto ou difundir informações para todos os usuários remotos simultaneamente, sem receber confirmação de recebimento do(s) mesmo(s), transmissões multi-ponto (*broadcast e multicast*).

Tabela 4 - Serviços da Camada de Enlace de Dados do PROFIBUS[VOL97].

Serviço	Função	Utilizado por	
DAS	<i>Send Data with Acknowledge</i>	FMS	
SRD	<i>Send and Request Data with Reply</i>	FMS	DP
SDN	<i>Send Data with No acknowledge</i>	FMS	DP
CSRD	<i>Cyclic Send and Request Data with Reply</i>	FMS	

• **Serviço cíclico:**

- **CSRD** (*Cyclic Send and Request Data with Reply*). Intercâmbio periódico de informação, permite ao usuário local intercambiar periodicamente informações com um ou mais usuários remotos.

6.4.2 Controle de Acesso ao Meio

O Controle de acesso ao meio, MAC (*Medium Access Control*) define quando a estação pode transmitir dados. O MAC garante que apenas uma estação tenha o direito de transmissão de dados num determinado instante.

O MAC deve satisfazer dois requisitos. Nas comunicações entre dispositivos com capacidade similar e alguma inteligência (tal como CNC's ou controladores programáveis), as estações devem ter um intervalo de tempo suficiente para executar suas tarefas de comunicação, mas quando a comunicação é entre um dispositivo mestre e um simples periférico (escravo), deve garantido um intercâmbio de dados tão simples e rápido quanto possível. Em cada caso se aplicam técnicas de acesso ao meio diferentes.[BEN93]

Nas comunicações entre estações mestres protocolo de Acesso ao Meio utiliza a técnica descentralizada ou distribuída de passagem de bastão, mas permite a configuração de acesso do tipo mestre-escravo para comunicação entre estações complexas (mestres) e simples periféricos (escravos). Esta combinação de métodos é denominada acesso ao meio híbrido.

O método de passagem de bastão, utiliza um quadro especial conhecido como bastão ou *token* para garantir à estação mestre o direito de acesso ao barramento num intervalo de tempo.

Com o método de acesso ao meio híbrido do PROFIBUS é possível implementar:

- um sistema mestre-escravo puro;
- um sistema mestre-mestre puro (passagem de bastão);
- um sistema com a combinação dos dois métodos.

A figura 8.4 mostra uma estrutura PROFIBUS com três estações ativas (mestres) e sete estações passivas (escravos). Os três dispositivos mestres formam um **anel lógico**.

O **anel lógico** representa uma cadeia de estações ativas organizadas logicamente pelos seus endereços é formado pela passagem seqüencial do bastão de uma estação ativa para a próxima seguindo uma certa seqüência, e a última estação passa finalmente o bastão à primeira estação.

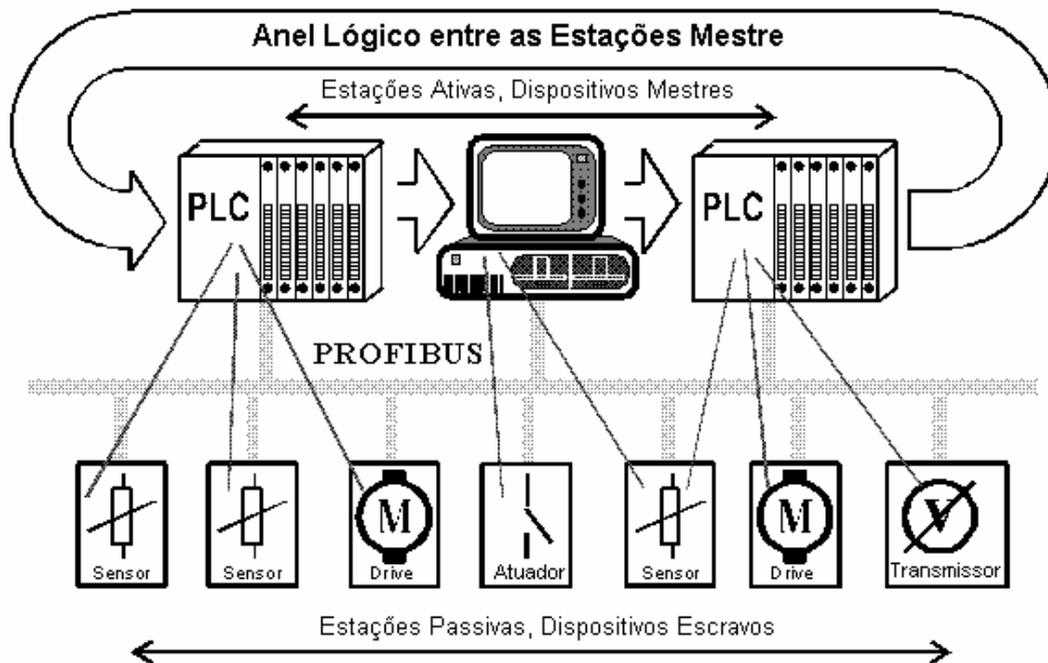


Figura 10. Acesso ao Meio do Tipo Híbrido

Para circulação do bastão cada estação conhece o endereço do seu predecessor PS (*Previous Station*) e sucessor NS (*Next Station*), além do próprio endereço TS (*This Station*).

Os endereços são determinados na fase de inicialização do sistema pelo MAC das estações ativas que detecta esta seqüência lógica e estabelece o anel.

Na fase de operação, as estações defeituosas ou desligadas devem ser eliminadas do anel e as novas estações devem ser incluídas atualizando os endereços dinamicamente, para tanto cada estação examina regularmente o espaço de endereços entre o seu próprio e o da próxima estação, o *gap*, para encontrar novas estações que estejam prontas para entrar no anel. Por este método é possível agregar novas estações à rede durante a operação ou retirar-las sem interromper a comunicação do sistema.

Existem procedimentos simples de administração do bastão para sua transmissão/recepção, inclusão e afastamento de estações, reinicialização do anel, gerenciamento dos tempos de rotação e administração de prioridades.

6.4.2.1 Características adicionais do MAC

- Reconhecer defeitos no meio de transmissão ou nos transeptores;

- Detectar erros no endereço das estações (múltipla utilização);
- Detectar erros na passagem de bastão (perda ou duplicidade de bastão).

6.4.3 Integridade dos dados

Outra importante tarefa da camada 2 é a integridade dos dados. A camada 2 do PROFIBUS assegura alta integridade dos dados. Todos os quadros tem distância de Hamming = 4. Isto é conseguido aplicando a norma internacional IEC (*International Electrotechnical Commission*) 870-5-1 através da escolha de delimitadores específicos de início e fim do quadro, sincronização *slip-free* e *bit* de paridade para cada octeto. [PRO92].

6.5 ESTRUTURA DOS QUADROS

6.5.1 Caractere de quadros (UART-Character)

Cada quadro consiste de um numero de caracteres de quadro, os caracteres UART. O caractere UART (UC) é um caracter *start-stop* para transmissão assíncrona e tem a seguinte estrutura :

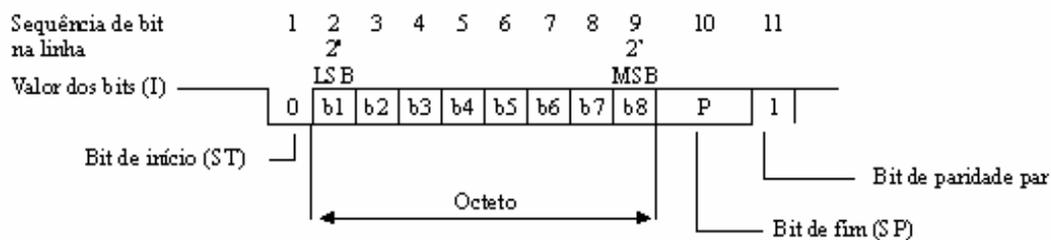


Figura 11. UART - Character

A representação do UART-Character baseia-se nas seguintes Normas: ISO 1177:1985, ISO 2022:1986 correspondente à DIN 66022 [DIN91].

6.5.2 Regra de transmissão

Cada quadro é formado por vários caracteres UART, constituídos por 11 *bits*: um *bit* de início (ST) com sinal binário “0”, 8 *bits* de informação (I) com sinal binário “0” ou “1”, um *bit* de paridade (P) com sinal binário “0” ou “1” e um *bit* de fim (SP) com binário sinal “1”. [DIN91]

Existem duas classes de quadros, um para transmissão entre estações sofisticadas e outro para os dispositivos de campo mais simples (mensagens curtas). Os quadros serão usados de forma especificada pelo protocolo, de acordo com os serviços solicitados.

6.5.3 Sincronização de bit

A sincronização de *bit* do receptor sempre inicia com o flanco negativo do *bit* de início, isto é, na transição de binário “1” para “0”. O *bit* de início e os *bits* restantes são inspecionados no centro temporário do *bit*. O *bit* de início deve ser binário “0” no centro de *bit*, caso contrário é interrompida a sincronização. O *bit* de fim com binário “1” finaliza a correta sincronização do UART-Character. Se em lugar do *bit* de fim ocorrer um *bit* binário “0”, isto será interpretado como sincronização falha ou erro de UART-Character, será marcado e aguardado o próximo flanco de inicialização.

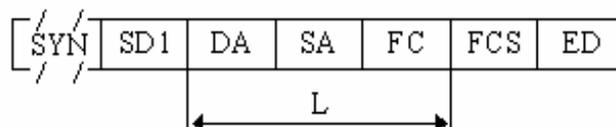
O desvio do relógio de recepção e transmissão de *bit* pode ser, no máximo, +/- 0,3% do valor nominal (veja características da camada física, velocidade de transmissão). [DIN91]

6.6 FORMATOS DOS QUADROS

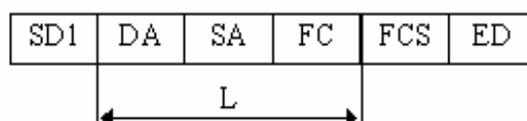
As figuras contidas nos itens a seguir mostram formatos de quadros de mesma categoria (Hd=4, tamanho fixo sem/com campo de dados e tamanho variável) e não seqüências (chamada-confirmação ou resposta), isto é, aos quadros de chamada podem suceder-se diferentes quadros de confirmação ou resposta. [DIN91]

6.6.1 Quadros de tamanho fixo sem campo de dados

A) Formato do quadro de chamada (Request-Frame):



B) Formato do quadro de confirmação (Acknowledgement-Frame):



C) Formato da confirmação breve (short Acknowledgement-Frame):

SC

onde:

SYN: *bits* de sincronização, no mínimo 33 *bits* de estado de repouso (*Idle*)

SD1: *byte* de início 1 (*Start Delimiter*), Code: 10H

DA: *byte* de endereço de destino (*Destination Adress*)

SA: *byte* de endereço fonte (*Source Adress*)

FC: *byte* de controle (*Frame Control*)

FCS: *byte* de teste (*Frame Check Sequence*)

ED: *byte* de fim (*End Delimiter*), Code: 16H

L: tamanho de campo de informações, quantidade fixa de *bytes* L=3

SC: caractere individual (*Single Character*), Code: E5H

Regras de transmissão

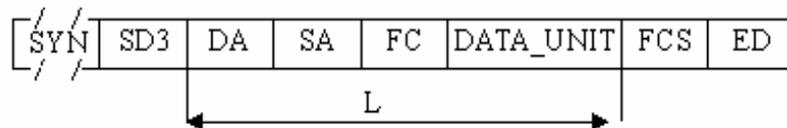
1. Estado de repouso da linha corresponde ao binário sinal “1”.
2. Antes de cada telegrama de chamada é necessário um estado de repouso de comprimento mínimo de 33 *bits* (*Syn-Time*).
3. Entre os UART-Characters de um telegrama não são permitidos estados de repouso.
4. O receptor verifica:
 - Por UART-Character: *bit* de partida, de fim e de paridade (par)
 - Por quadro: *byte* de início, DA/SA, FCS e de fim e na chamada o *Syn-Time*. Se a verificação der um resultado negativo, todo o quadro deve ser rejeitado.

Observação: SC e SD1(bem como SD2 e SD3, veja os itens. 6.6.2 e 6.6.3) tem Hd =4 entre si e tem segurança contra realocação entre si (veja a IEC 870-5-1 ou DIN 19244 parte 10 (em elaboração), isto é, o caractere individual SC é quase um telegrama Hd=4.

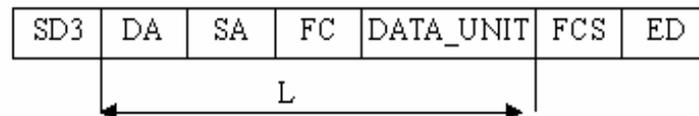
Em chamadas, que somente são confirmadas (*Send Data with Acknowledge*), o SC é permitido como confirmação positiva. Em chamadas, que são respondidas (*Send and Request Data with Reply*), o SC é permitido como confirmação negativa, se não houver disponibilidade de dados. [DIN91]

6.6.2 Quadros de tamanho fixo com campo de dados

A) Formato do quadro de chamada (*Send/Request-Frame*):



B) Formato do quadro de resposta (*Response-Frame*):



onde:

SYN: *bits* de sincronização, no mínimo 33 *bits* de estado de repouso (*Idle*)

SD3: *byte* de início 3 (*Start Delimiter*), Code: A2H

DA: *byte* de endereço de destino (*Destination Adress*)

SA: *byte* de endereço fonte (*Source Adress*)

FC: *byte* de controle (*Frame Control*)

DATA_UNIT: campo de dados, tamanho fixo $(L-3) = 8$ *bytes*

FCS: *byte* de teste (*Frame Check Sequence*)

ED: *byte* de fim (*End Delimiter*), Code: 16H

L: tamanho de campo de informações, quantidade fixa de *bytes* $L=11$

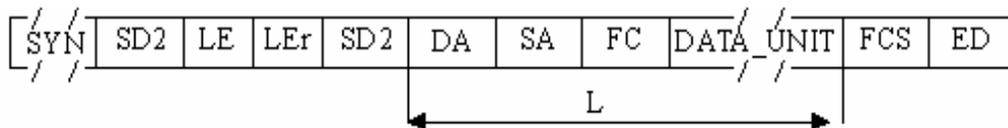
Regras de transmissão

Também aqui valem as regras de transmissão como no formato sem campo de dados do item 6.6.1.

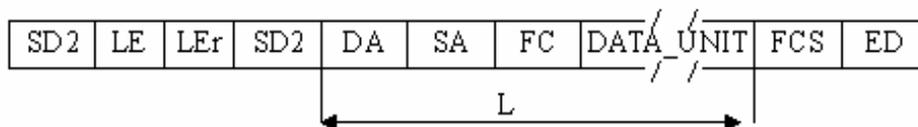
6.6.3 Quadros com campo de dados de tamanho variável

Para quantidade variável de *bytes* de dados deve ser transmitido também o tamanho atual no quadro. Esta indicação de tamanho consta em duplicidade no início do quadro, num cabeçalho fixo de quadro e, portanto, está assegurado com Hd=4 e contra escorregamento.

A) Formato do quadro de chamada (*Send/Request-Frame*):



B) Formato do quadro de resposta (*Response-Frame*):



onde:

SYN: *bits* de sincronização, no mínimo 33 *bits* de estado de repouso (*Idle*)

SD2: *byte* de início 2 (*Start Delimiter*), Code: 68H

LE: *Byte* de comprimento (*length*), valor: 4 até 249

LEr: *byte* de comprimento repetido (*repeated*)

DA: *byte* de endereço de destino (*Destination Adress*)

SA: *byte* de endereço fonte (*Source Adress*)

FC: *byte* de controle (*Frame Control*)

DATA_UNIT: campo de dados, tamanho fixo (L-3) , máximo 246 *bytes*

FCS: *byte* de teste (*Frame Check Sequence*)

ED: *byte* de fim (*End Delimiter*), Code: 16H

L: tamanho de campo de informações, quantidade variável de *bytes* L = 4 até 249.

Regras de transmissão

Também aqui valem as regras de transmissão como no formato com tamanho fixo, item 8.6.1

Adicionalmente à regra de transmissão 4 o LE e o LEr devem corresponder entre si, e os *bytes* de informações devem ser contados juntos a partir do endereço de destino(DA) até o *byte* de teste (FCS) e comparados com a indicação do tamanho (LE). [DIN91]

6.7 QUADRO DE TOKEN



onde:

SYN: *bits* de sincronização, no mínimo 33 *bits* de estado de repouso (*Idle*)

SD4: *byte* de início 4 (*Start Delimiter*), código: DCH

DA: *byte* de endereço de destino (*Destination Adress*)

SA: *byte* de endereço fonte (*Source Adress*)

Regras de transmissão

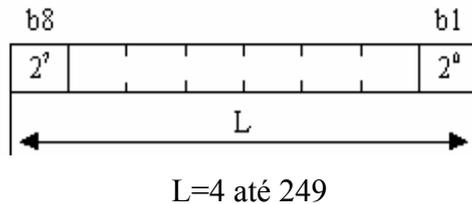
1. Estado de repouso da linha corresponde ao sinal binário “1”
2. Antes de cada quadro de Token é necessário um estado de repouso de comprimento mínimo de 33 *bits* (*Syn-Time*).
3. Entre os caracteres UART de um quadro não são permitidos estados de repouso.
4. O receptor verifica por cada caractere UART: o *bit* de início, de fim e a paridade (par), e por cada quadro: o *Syn-Time* e o *byte* de início DA/SA. Se a verificação der um resultado negativo, todo o quador deve ser rejeitado. [DIN91]

6.8 COMPRIMENTO, ENDEREÇO, CONTROLE E TESTE

6.8.1 Byte de Comprimento (LE)

Os dois *bytes* de comprimento de mesmo valor no cabeçalho do quadro de formato variável contém o número de *bytes* de informação no corpo do quadro. A estes

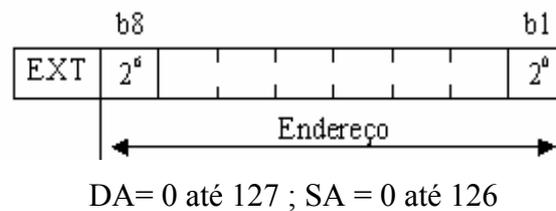
pertencem: DA, SA, FC e a DATA_UNIT. O valor pode variar entre 4 a 249, de forma que podem ser transmitidos, no máximo, 246 *bytes* na DATA_UNIT de um quadro. Um valor menor que 4 não é permitido, uma vez que um quadro engloba, no mínimo, um *byte* de DA, SA, FC e DATA. O quadro mais longo engloba um total de 255 *bytes*.



Ocupação de *bytes* de comprimento

6.8.2 Byte de endereço (DA/SA)

Os dois *bytes* de endereço no cabeçalho dos quadros (quadros de chamada, confirmação e resposta) contém o endereço de estação de destino (DA) e da fonte (SA). O quadro de Token consiste somente de dois *bytes* de endereço após o *byte* de início.



Ocupação de *bytes* de endereço

O endereço 127 (de b1 até b7 =1) é reservado como endereço global para mensagens *broadcast* ou *multicast* (quadro encaminhado a todas as estações ou grupo de estações selecionado através de ponto de acesso de serviço; somente permitido no *Send_Data_with_No_Acknowledge*, SDN). Desta forma estão disponíveis 127 endereços de estações (de 0 a 126) para estações passivas e ativas. Destas, devem ser ocupadas preferencialmente, no máximo, 32 estações ativas. Para aplicações não críticas de tempo são permitidas opcionalmente até 127 estações ativas. Uma vez que é necessária, no mínimo, uma estação ativa, são possíveis, no máximo, 126 endereços para estações passivas.

Os *bytes* de endereço do telegrama de chamada devem ser refletidos de forma espelhada no telegrama de confirmação ou de resposta, isto é, o *byte* SA do

telegrama de confirmação ou de resposta contém o endereço da estação de destino e o *byte* DA o endereço da estação fonte da chamada. [DIN91]

6.9 A CAMADA DE APLICAÇÃO (7)

A camada 7 do modelo de referência ISO/OSI fornece os serviços de aplicação para o usuário. Estes serviços servem para o intercâmbio de dados entre aplicações de processos diferentes.

A camada de aplicação do PROFIBUS inspira-se fortemente no MMS (*Manufacturing Message Specification*) adotando, sempre que possível, as mesmas primitivas de serviços sem deixar de considerar os aspectos específicos do barramento de campo [MEN91]. A camada de aplicação do PROFIBUS-FMS está especificada na DIN 19245 parte 2 e consiste de:

- Fieldbus Message Specification (FMS)
- Lower Layer Interface (LLI).
- Application Layer Interface (ALI).

O protocolo de aplicação propriamente dito FMS (*Fieldbus Message Specification*) descreve a comunicação orientada a objetos, os serviços de aplicação e os modelos resultantes do ponto de vista da comunicação de entidades pares.

A interface de mapeamento com a camada de enlace LLI (*Lower Layer Interface*) serve para adaptar as funções de aplicação as características da camada 2.

A interface de mapeamento com os AP's (*Application Processes*) ALI (*Application Layer Interface*) exerce a função de cliente (solicitador de serviços).

6.9.1 Modelo de Comunicação PROFIBUS

O processo de uma aplicação inclui todos os programas, fontes e tarefas que não estão especificadas nas camadas de comunicação. O modelo de comunicação PROFIBUS permite a distribuição dos processos ou a união em um processo comum, utilizando as chamadas relações de comunicação. Num processo de aplicação um dispositivo de campo virtual VFD (*Virtual Field Device*) é a parte do dispositivo com a qual é possível comunicar-se. Por tanto o VFD é uma abstração de vários tipos de dispositivos reais.

Num dispositivo real os diversos objetos que podem ser comunicados (variáveis, domínios, alarmes, listas de variáveis e invocações de programas) são chamados de objetos de comunicação.

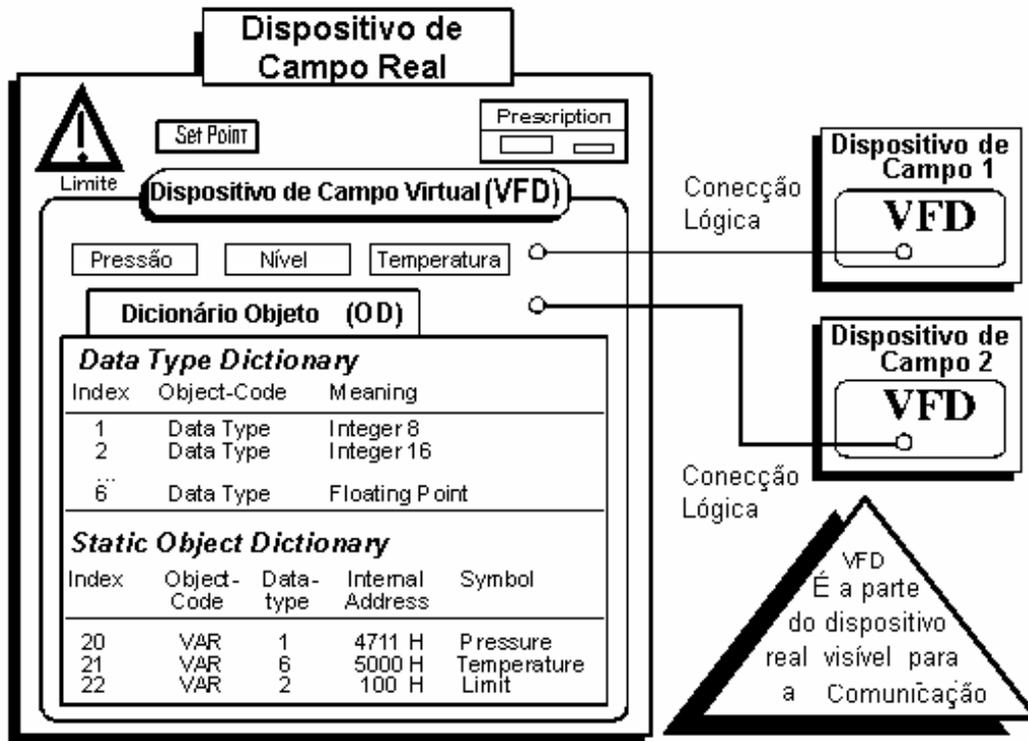


Figura 12. Virtual Field device (VFD) and Object Dictionary (OD)

VFD e OD

No modelo de comunicação PROFIBUS, o mapeamento das funções do VFD no dispositivo real é fornecido pela ALI (*Application Layer Interface*). A figura 6 mostra a relação entre o dispositivo de campo real e o VDF. Neste exemplo, apenas as variáveis pressão, nível e temperatura, que são parte do VDF, podem ser lidos ou escritos pelas duas relações de comunicação.

6.9.2 Objetos da Comunicação

Todos os objetos utilizados na comunicação das estações PROFIBUS estão descritos num dicionário objeto OD (*Object Dictionary*).

O OD pode ser predefinido num dispositivo simples. Num dispositivo complexo o OD é configurado e carregado, local ou remotamente, no dispositivo. O OD

contém descrições, estrutura e tipo de dados. Também contém o endereço interno dos objetos de comunicação dos dispositivos. O OD consiste de:

- Cabeçalho (contém a informação sobre a estrutura do OD).
- Listas de tipos estáticos (contém a lista dos tipos de dados estáticos suportados).
- Dicionário objeto estático (contém todos os objetos estáticos de comunicação).
- Lista dinâmica da lista de variáveis (contém a lista atualizada de todas as listas de variáveis conhecidas).
- Lista dinâmica de invocação de programas (contém a lista de todos os programas conhecidos).

As partes individuais do OD são implementadas apenas se o dispositivo é capaz de suportar a função correspondente.

Os objetos de comunicação estática devem ser definidos no dicionário objeto estático. Pode ser predefinido pelo fabricante do dispositivo, ou definido durante a configuração do sistema no barramento. Os objetos de comunicação estática são normalmente utilizados para comunicação entre dispositivos de campo. PROFIBUS se distingue pelos seguintes objetos de comunicação estática:

- Variável simples
- Array (linha de variáveis simples com tipos diferentes)
- Registro (seqüência de variáveis simples com tipos diferentes)
- Domínio (uma faixa de dados)
- Eventos

Os objetos de comunicação dinâmica estão contidos na parte dinâmica do OD. Pode ser predefinido em tempo de configuração ou definido dinamicamente, excluído ou mudado pelos serviços da aplicação na fase de operação. PROFIBUS suporta os seguintes objetos de comunicação dinâmica:

- Invocação de programas

- Lista de variáveis (seqüência de variáveis simples, arrais ou registros)

Endereçamento lógico é o método preferido para endereçamento na comunicação de objetos. Eles são acessados com um endereçamento curto, chamado índice. O índice um número do tipo Unsigned 16. Este faz possível mensagens eficientes e minimiza o *overhead* do protocolo. Para cada objeto de comunicação um índice é definido no OD. Todos os dispositivos PROFIBUS devem suportar o método de endereçamento lógico.

O PROFIBUS-FMS permite ainda um método de endereçamento opcional para aplicações especiais:

6.9.3 Tipos de endereçamentos

Endereçamento pelo nome O nome simbólico do objeto de comunicação é transferido pelo barramento.

Endereçamento físico Um endereço físico da memória local no dispositivo pode ser acessada com o serviço de PhysRead e PhysWrite.

Para prevenir acesso não autorizados pode ser definido uma **proteção de acesso** para cada objeto de comunicação. O acesso a um dispositivo ou a um grupo de dispositivos é possível apenas com uma certa senha. Senha e grupo de dispositivos pode ser definida individualmente para cada objeto em um OD. Adicionalmente, pela definição no OD, é possível restringir o serviço que tem acesso permitido (por ex.: acesso permitido para apenas leitura).

7. A REDE DE MICROTERMINAIS

7.1 INTRODUÇÃO

O protocolo PROFIBUS, descrito no capítulo anterior, foi utilizado para implementar a comunicação da rede monomestre analisada nesta dissertação.

As camadas Física e de Acesso ao Meio foram desenvolvidas e implementadas no laboratório de Instrumentação Eletro-Eletrônica por mestrandos integrantes de grupos de pesquisa.

Às camadas já existentes foi agregada uma camada de aplicação com a função específica de realizar o intercâmbio de dados necessário para o uso com os microterminais de consulta.

Para dar maior flexibilidade no uso das funções do protocolo foi criada uma biblioteca em linguagem C que contém funções básicas de acesso aos serviços das camadas inferiores.

7.2 DESCRIÇÃO DE UM APLICATIVO

O sistema utilizado está composto por um computador e vários terminais conectados através de um barramento PROFIBUS. O computador realiza as tarefas de controle da rede de comunicação e também roda um programa para intercâmbio de dados com os terminais.

As informações trocadas pela rede podem ser das mais diversas apenas como exemplo foi desenvolvido um aplicativo para consulta numa livraria na qual estariam espalhados os terminais que ligados em rede permitem ao usuário consultar os dados dos clientes cadastrados assim como os dados referentes aos livros existentes.

O código do livro, fornecido via teclado, é a chave para a consulta das informações que constam no respectivo registro, como por exemplo título, nome do autor quantidade em estoque, etc.

Tabela 5 - Registro do Livro

Código	Título do Livro	Nome do Autor	Estoque
001	Martín Fierro	José Hernández	20

O código do cliente, por outro lado, fornece informações tais como: nome, endereço, telefone, saldo disponível para compras, etc.

Tabela 6 - Registro do Cliente

Código	Nome	Endereço	Telefone	Crédito
001	Samuel Diaz	Andradas,1020 /203	(051) 335 21 12	100,00

O programa no computador recebe do microterminal os códigos, pesquisa no banco de dados correspondente e devolve a resposta. O nível de complexidade das aplicações que rodam tanto no computador quanto nos terminais não é relevante para as análises da rede e sim as características das mensagens que trafegam no barramento.

Cabe destacar que o programa dos terminais tem capacidade de executar, além das funções de comunicação, rotinas desenvolvidas especialmente para acessar o teclado, o *display* e a impressora matricial Epson LX-810.

7.3 O MESTRE

7.3.1 Hardware

A estação mestre tem como plataforma um computador IBM-PC ou compatível. Desta forma, um único processador é utilizado para executar o software de comunicação (protocolo PROFIBUS) e o programa aplicativo do usuário.

A porta serial do computador (RS-232) é utilizada para a transmissão e recepção das informações. Por meio de um circuito adaptador que utiliza o transceptor SN75176A, o padrão elétrico RS-232 é convertido para o EIA-485 conforme descrição da camada física do protocolo PROFIBUS. O circuito adaptador provê também isolamento galvânica.

7.3.2 Software

Às camadas física e de enlace de dados do protocolo PROFIBUS, foi agregada uma camada de aplicação com a função específica de realizar o intercâmbio de dados necessário para o uso com os microterminais de consulta.

Para dar maior flexibilidade no uso das funções do protocolo foi criada uma biblioteca em linguagem C, a qual contém funções básicas que administram o tráfego da informação. As funções estão descritas a seguir:

- inicio()
- liste_status()
- leia_estacao()
- responda_estacao()
- fim()

A função *inicio()* tem como objetivo conectar o usuário à rede, para tanto faz a chamada da função do protocolo *profi_start()* que por sua vez chama rotinas de inicialização de parâmetros das camadas física e de enlace de dados. Entre os parâmetros inicializados por esta função se encontram a velocidade de transmissão, o endereço e o tipo da estação, se mestre ou escravo.

A função *liste_status()* fornece ao usuário uma lista, conhecida como *Live_list*, que contém o *status* das estações conectadas à rede. O *status* indica se a estação está efetivamente ligada e pronta para fazer pedidos e a posição no vetor indica o endereço da mesma, por meio desta informação o mestre consulta seqüencialmente cada uma das estações e atende, caso existam, às suas solicitações.

Conhecendo os endereços dos escravos ligados à rede o mestre utilizará a função *leia_estacao()* para consultar as estações remotas, utilizando as primitivas do serviço SRD da camada 2: *FDL_DATA_REPLY.request()* e *FDL_DATA_REPLY.confirm()*. Esta função retorna um vetor que contém um código, que será igual a zero quando a estação escrava não tiver nenhuma solicitação.

Quando o código recebido pelo mestre for diferente de zero isto indica que houve uma solicitação. A resposta é enviada de imediato utilizando a função *responda_estacao()*. Esta função envia os dados solicitados pela estação remota através da chamada das primitivas *FDL_DATA_ACK.request()* e *FDL_DATA_ACK.confirm()* do nível 2, ou seja, enviando dados pelo serviço SDA (*Send Data with Acknowledge*).

Todas as estações são consultadas pelo menos uma vez antes de realizar uma nova pesquisa para atualizar a *Liv_list*, e assim com a nova lista verificar

novamente os pedidos dos terminais. Esta atualização periódica da lista de estações ligadas é necessária devido à característica da rede que permite a conexão e desconexão das estações *on line* sem alterar o funcionamento da mesma.

A função *fm()* chama a função *profi_quit()* do protocolo que desconecta o usuário da rede. Quando esta função é executada pelo mestre finaliza também o funcionamento da aplicação que roda no PC.

Estas funções da biblioteca foram consideradas suficientes para a estação mestre poder gerenciar o funcionamento da rede. A partir destas funções podem ser criados uma grande variedade de aplicativos, dentro da proposta inicial do trabalho, que atenda as necessidades específicas para cada caso.

7.4 OS ESCRAVOS

7.4.1 Hardware

Como plataforma para os escravos foram utilizados sistemas de desenvolvimento baseados no microcontrolador 8098 da Intel, cujo *hardware* foi projetado no Laboratório de Instrumentação Eletro-Eletrônica [ZUC95]. Entre as características destes sistemas, as seguintes são de relevância para os microterminais:

- Memória: até 56 *Kbytes* configuráveis entre memória de programa (EPROM) e memória de dados (RAM);
- Uma porta serial padrão EIA-485, com isolamento galvânica, para ligação direta à rede.
- Uma porta paralela para ligar uma impressora.
- Uma saída para teclado matricial de no máximo 16 teclas (4 linhas x 4 colunas).
- Uma saída para *display* de cristal líquido de 2 linhas por 16 caracteres.

7.4.2 Software

O protocolo PROFIBUS permite a ligação de estações mestres e escravos num barramento de comunicação. As estações mestres podem enviar mensagens e fazer perguntas às outras estações, mas não a qualquer momento, e sim apenas quando

estiverem com o direito de acesso ao barramento. Este direito é determinado pela posse do bastão que vai sendo passado entre os mestres criando desta forma um anel lógico de comunicação. Esta técnica de passagem de bastão é o método utilizado para disciplinar o acesso ao meio. As estações escravas não recebem o bastão e, portanto, não tem direito de ocupar o barramento, apenas de responder quando solicitadas.

Sendo que a rede descrita neste trabalho é monomestre, os terminais sendo todos escravos não terão nunca acesso ao barramento. Considerando que nesta aplicação é fundamental os terminais poderem realizar pedidos, algumas rotinas foram adaptadas permitindo que, quando consultadas, as estações escravas enviassem códigos. Esses códigos representam os pedidos que um usuário pode realizar via teclado numa estação escrava.

Os terminais escravos utilizam as seguintes funções da biblioteca criada para este trabalho:

- `inicio()`
- `requisicao()`

A função `inicio()` descrita no item 7.3.2, realiza a conexão do terminal à rede.

A função `requisicao()` é executada pelo terminal escravo sempre que precise fazer uma solicitação ao mestre. Para tanto a função atualiza um *buffer* que é acessado pelo mestre utilizando o serviço SRD (*Send and Request Data with Reply*). Este *buffer* contém o código que corresponde à solicitação. Este artifício foi criado, pois os terminais não recebem nunca o bastão e portanto não tem direito de acesso ao barramento. Como não podem realizar perguntas fazem as solicitações atualizando o *buffer* com um código. O mestre lê este *buffer* quando executa a rotina `leia_stacao()` e assim fica sabendo qual o pedido feito e imediatamente o responde através da rotina `responda_stacao()`.

8. DESCRIÇÃO DOS TEMPOS

8.1 INTRODUÇÃO

Para avaliar a Rede de Consulta Monomestre proposta neste trabalho será feito um estudo comparativo entre os valores determinados teoricamente e os valores obtidos através de medidas práticas.

Nos primeiros itens deste capítulo são descritos os tempos relevantes aos testes realizados com o objetivo de fornecer uma visão geral sobre as diferentes etapas no intercâmbio dos quadros de comunicação.

8.2 TEMPOS DE SUPERVISÃO

Os tempos de supervisão da rede serão sempre expressos em bits e representados pela letra T. O tempo T é obtido dividindo-se o tempo t em segundos pelo tempo de bit t_{BIT} .

$$T = t / t_{\text{BIT}} \quad (1)$$

O tempo de bit t_{BIT} (*Bit-Time*) é o tempo gasto para a transmissão de um bit. Ele corresponde ao valor recíproco da velocidade de transmissão:

$$t_{\text{BIT}} = 1/\text{velocidade de transmissão em bit/s} \quad (2)$$

8.2.1 Tempo de Sincronismo - T_{SYN}

O tempo de sincronismo T_{SYN} (*Synchronization Time*) é o período mínimo de tempo durante o qual o meio de transmissão deve permanecer em estado de linha muda (*idle state*), antes de qualquer estação poder aceitar o início de um quadro de requisição (*Request ou Send/Request*) ou quadro de bastão (*token*).

O tempo de sincronismo é igual a:

$$T_{\text{SYN}} = 33 \text{ bits} \quad (3)$$

8.2.2 Tempo de Espera - T_{QUI}

Depois da queda do modulador no desligamento do transmissor ou na comutação do repetidor, as transmissões e recepções devem ser desabilitadas. Este tempo de espera é conhecido por T_{QUI} (*Quit Time*).

8.2.3 Tempo de Atraso na Transmissão - T_{TD}

O tempo de atraso na transmissão do quadro, T_{TD} (*Transmission-Delay-Time*), é definido como o tempo máximo gasto no meio de transmissão, entre um transmissor e um receptor, sendo considerados também os tempos de atraso causados por repetidores na linha .

$$T_{TD} = t_{TD} / t_{BIT} \quad (4)$$

Por exemplo, numa linha de 1 Km de comprimento sem repetidor o t_{TD} é de aproximadamente $5\mu s$. Assim, para uma velocidade de transmissão de 500 Kbit/s, resulta um $T_{TD} = 2,5$ bits.

8.2.4 Margem de Segurança - T_{SM}

A margem de segurança T_{SM} (*Safety Margin*) é um acréscimo de tempo definido pelo seguinte intervalo:

$$T_{SM} = 2 \text{ bits} + 2 \cdot T_{SET} + T_{QUI} \quad (5)$$

Onde T_{SET} é o tempo de liberação (*Setup-Time*) gasto desde o início de uma ocorrência (p.ex. Interrupt: último caractere de quadro transmitido ou T_{SYN} expirado) até a execução da reação necessária (p.ex. iniciar T_{SYN} ou liberar receptor).

$$T_{SET} = t_{SET} / t_{BIT} \quad (6)$$

8.2.5 Tempo de Reação da Estação - T_{SDx}

Entre o recebimento do último bit de um quadro e a transmissão do primeiro bit do quadro seguinte pela estação, decorre um período definido como tempo de reação da estação T_{SDx} (*Station-Delay-Time*). Este tempo é necessário para a interpretação do quadro recebido e a composição do quadro seguinte e depende da realização do protocolo na estação.

São definidos os seguintes tempos de reação:

T_{SDI} (*Station Delay of Initiator*), tempo de reação da estação que transmite uma requisição ou um *token*:

$$T_{SDI} = t_{SDI} / t_{BIT} \quad (7)$$

T_{SDR} (*Station Delay of Responder*), tempo de reação da estação que transmite uma confirmação ou resposta:

$$\text{mín. } T_{SDR} = \text{mín. } t_{SDR} / t_{BIT} \quad (8)$$

$$\text{máx. } T_{SDR} = \text{máx. } t_{SDR} / t_{BIT} \quad (9)$$

É importante considerar que o mínimo tempo de reação da estação que responde deve ser maior que o tempo de espera T_{QUI} , definido no item 8.2.2. A fim de manter esta condição o T_{SDR} deve, eventualmente, ser prolongado.

$$T_{QUI} < \text{mín. } T_{SDR} \quad (10)$$

8.2.6 Tempo de Disponibilidade - T_{RDY}

O tempo de disponibilidade T_{RDY} (*Ready-Time*) é o tempo após o qual uma estação mestre deve estar preparada para a recepção de uma confirmação ou resposta após a transmissão de uma requisição. O tempo de disponibilidade é definido da seguinte forma:

$$T_{RDY} < \text{mín. } T_{SDR} \quad (11)$$

Para manter a condição anterior deve ser prolongado eventualmente o T_{SDR} .

Também aqui deve ser levado em consideração o T_{QUI} , conforme definido no item 8.2.2. A recepção só pode se habilitada após este tempo:

$$T_{QUI} < T_{RDY} \quad (12)$$

Para manter a condição anterior deve ser prolongado eventualmente o T_{RDY} e, se necessário, também o T_{SDR} .

8.2.7 Tempo de Estado de Repouso - T_{ID}

O tempo de estado de repouso T_{ID} (*Idle-Time*) é o tempo que transcorre na estação que envia a requisição desde a recepção do último bit de um quadro, até que seja transmitido o primeiro bit do quadro seguinte no meio de transmissão.

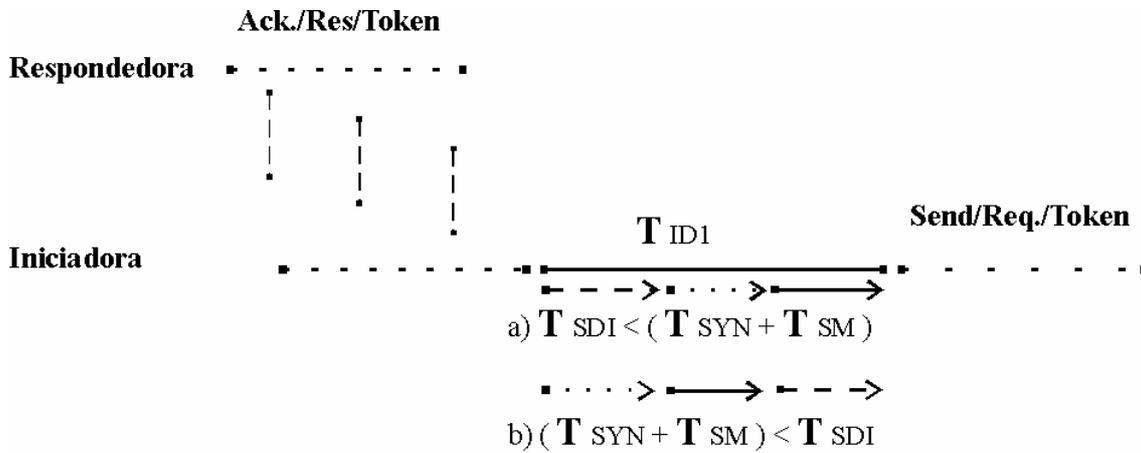


Figura 13. Idle-Time T_{ID1}

$$T_{ID1} = \text{máx} (T_{SYN} + T_{SM} , T_{SDI})^2 \quad (13)$$

O tempo de estado de repouso é também o tempo entre o último bit de um quadro sem resposta (SDN) transmitido e o primeiro bit do próximo quadro.

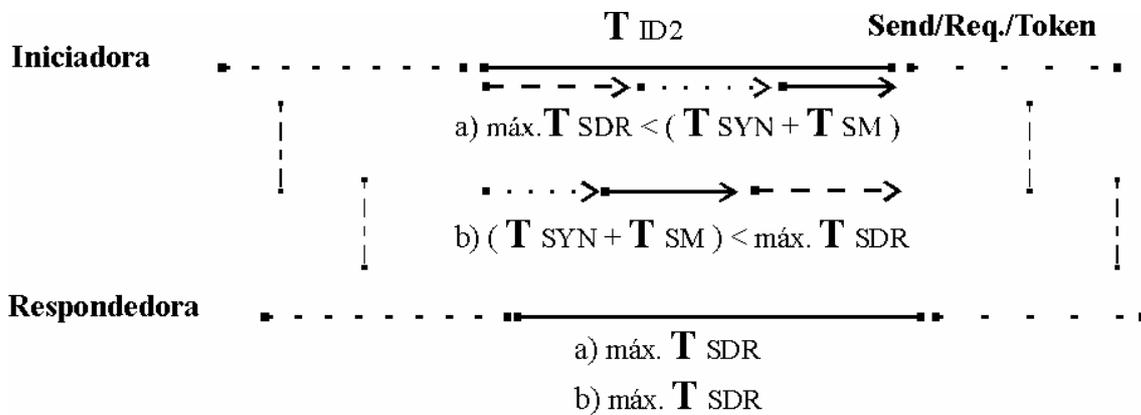


Figura 14. Idle-Time T_{ID2}

$$T_{ID2} = \text{máx} (T_{SYN} + T_{SM} , \text{máx. } T_{SDR})^2 \quad (14)$$

² Deve ser sempre utilizado o maior valor.

O tempo de estado de repouso deve corresponder, no mínimo, ao tempo de sincronismo (T_{SYN}) mais o acréscimo de segurança T_{SM} (caso (a) da figura 13 e da figura 14. Em altas velocidades de transmissão (caso (b) da figura 13 e da figura 14), o T_{SYN} é muito pequeno, de forma que os tempos de reação respectivos (T_{SDI} em (a) e T_{SDR} em (b)) devem ser levados em consideração.

8.2.8 Tempo de Espera pela Recepção - T_{SL}

O tempo de espera pela recepção T_{SL} (*Slot-Time*) é o tempo máximo que a estação que envia um quadro de requisição espera, desde a transmissão do último bit deste quadro, até a recepção completa do primeiro caractere do quadro de resposta correspondente. Ou seja, após um quadro de requisição (*Request ou Send/Request*) vale o seguinte tempo de espera:

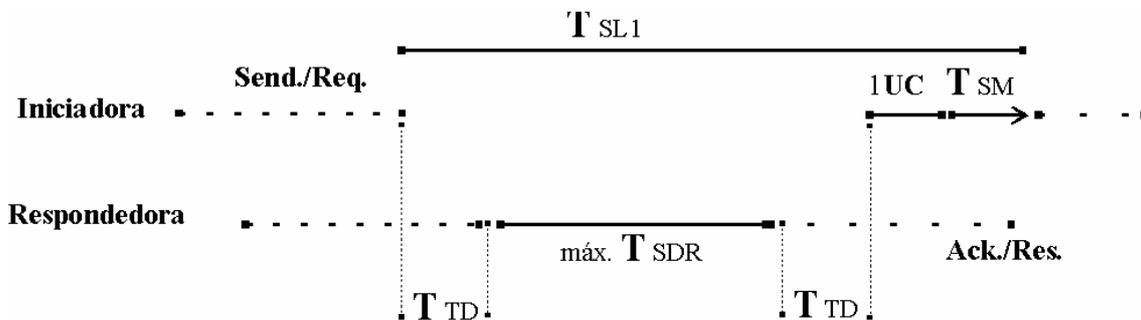


Figura 15. Slot-Time T_{SL1}

$$T_{SL1} = 2 \cdot T_{TD} + \text{máx. } T_{SDR} + 11 \text{ bits} + T_{SM} \quad (15)$$

Após o envio de um quadro de *token*, o T_{SL} corresponde ao máximo tempo de espera até a recepção completa do primeiro caractere do quadro enviado pela estação que recebeu o *token*.

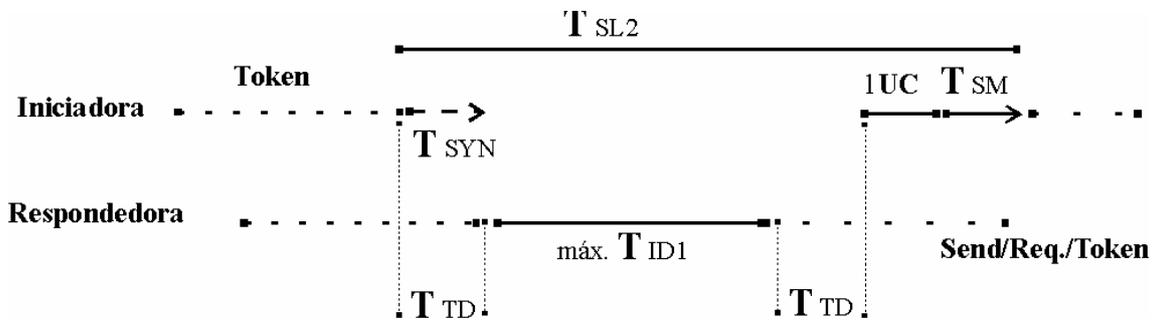


Figura 16. Slot-Time T_{SL2}

$$T_{SL2} = 2 \cdot T_{TD} + \text{máx. } T_{ID1} + 11 \text{ bits} + T_{SM} \quad (16)$$

Observação: Para a simplificação da implementação somente é utilizado o maior T_{SL} do sistema. Em se tratando puramente de um tempo de supervisão, isso não influi negativamente no tempo de reação do sistema.

$$T_{SL} = \text{máx. } (T_{SL1}, T_{SL2})^3 \quad (17)$$

8.2.9 Tempos de Transmissão de Quadro - $T_{S/R}$, $T_{A/R}$

Os tempos de quadro ($T_{S/R}$, $T_{A/R}$) são determinados pela quantidade de caracteres da UART (UC). Disto resulta:

$$T_{S/R} = a \cdot 11 \text{ bits} \quad (18)$$

onde (a) é a quantidade de UC no quadro de requisição (*Send/Request-Frame*).

$$T_{A/R} = b \cdot 11 \text{ bits} \quad (19)$$

onde (b) é a quantidade de UC no quadro de resposta (*Ack./Response-Frame*).

³ Deve ser sempre utilizado o maior valor.

8.2.10 Time-out - T_{TO}

O T_{TO} (*Time-out*) serve para a supervisão da atividade de barramento ou do tempo de estado de repouso T_{ID} (*Idle-Time*) na estação ativa e passiva. Se o T_{ID} alcança o valor de *Time-out*, o barramento é considerado inativo (caso de erro, p.ex., através de perda de *token*).

A supervisão do barramento inicia-se imediatamente após o PON (*Power ON*) no estado "*Listen-Token*" ou "*Passive-Idle*" ou mais tarde, após a recepção do último bit de um quadro, e finaliza com a recepção do primeiro bit de um quadro seguinte.

O *Time-out* é definido da seguinte forma:

$$T_{TO} = 6.T_{SL} + 2.n.T_{SL} \quad (20)$$

Para estações ativas n representa o próprio endereço e pode variar de 0 a 126.

Para estações passivas n vale 130, independente do endereço da estação.

O primeiro termo assegura que existe espaço de tempo maior que o máximo T_{ID} possível entre dois quadros. O segundo termo assegura que após um caso de erro nem todas as estações ativas aceitem simultaneamente o *token*.

8.2.11 Tempo de Atualização - T_{GUD}

O tempo de atualização GAP T_{GUD} (*GAP-Update-Time*) serve à estação ativa para a inicialização da atualização da GAP. Sempre é introduzida uma renovação da representação GAP, após a primeira criação da GAPL ciclicamente após o tempo, isto na próxima recepção do Token possível, se ainda houver disponibilidade de tempo de transmissão após o ciclo regular de mensagens, ou então em fases posteriores de retenção do *token*. O tempo é um múltiplo do *Target-Rotation-Time* do *token* T_{TR} e é definido da seguinte forma:

$$T_{GUD} = G.T_{TR} \quad (21)$$

onde, $1 \leq G \leq 100$

T_{TR} Target-Rotation-Time de *token* parametrizado.

8.3 TEMPO DE CICLO DE MENSAGENS - MC

Um ciclo de mensagens MC (*Message-Cycle*) consiste de um quadro de requisição (*Request-Frame, Send/Request-Frame*) e do respectivo quadro de confirmação ou resposta (*Acknowledgement-Frame, Response-Frame*). O tempo de um ciclo de mensagem consiste da soma dos tempos de transmissão dos quadro, dos tempos de retardo e dos tempos de processamento das estações.

Entre o envio/requisição e a confirmação/resposta decorre o tempo de processamento da estação (T_{SDR}). Este tempo é necessário para a interpretação da requisição e composição do quadro de confirmação ou de resposta (processamento do protocolo).

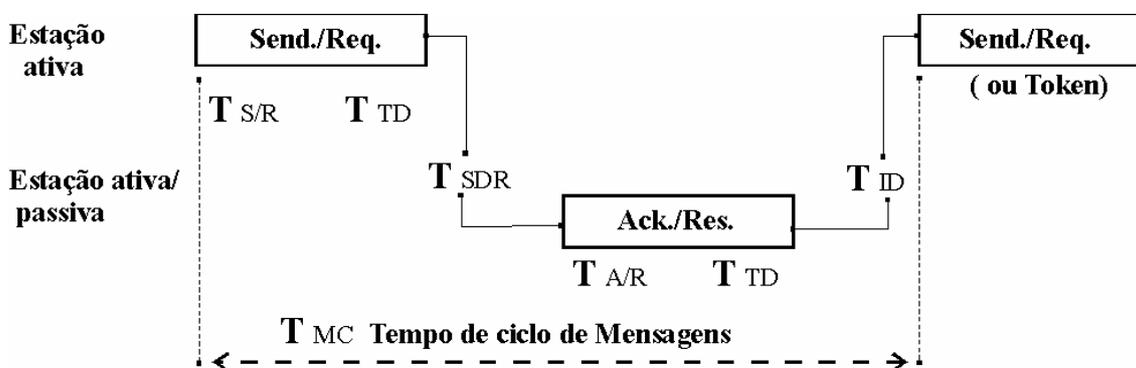


Figura 17. Ciclo de Mensagem

O T_{ID} (*Idle-time*), que transcorre entre a confirmação ou resposta e a nova requisição, também considera o tempo de processamento do protocolo (T_{SDI}). Deverá ser considerada também a soma $T_{SYN} + T_{SM}$, que deve ser utilizada caso a mesma seja maior que T_{SDI} .

$$T_{MC} = T_{S/R} + T_{SDR} + T_{A/R} + T_{ID} + 2.T_{TD} \quad (22)$$

onde:

T_{MC} : Tempo de ciclo de mensagem

$T_{S/R}$: Tempo de transmissão de um quadro de envio/requisição

T_{SDR} : Tempo de reação da estação que responde

$T_{A/R}$: Tempo de transmissão de um quadro de confirmação/resposta

T_{ID} : Tempo de estado de repouso

T_{TD} : Tempo de atraso na transmissão

Observação: Nos casos em que a taxa de transmissão é menor que 100 Kbit/s a interpretação do quadro pode ocorrer durante a recepção, conseqüentemente os tempos de reação diminuem sensivelmente.

8.4 TAXA DE MENSAGENS R_{SYS}

A taxa de mensagens R_{SYS} no sistema corresponde à quantidade de ciclos de mensagens por segundo:

$$R_{SYS} = 1/t_{MC} \quad (23)$$

$$t_{MC} = T_{MC} \cdot t_{BIT} \quad (24)$$

8.5 TEMPO DE REAÇÃO DO SISTEMA - T_{SR}

Num sistema formado por uma estação mestre e n estações escravas (monomestre) onde há varredura cíclica das estações (*polling*), o tempo máximo de reação do sistema T_{SR} é calculado a partir do tempo de um ciclo de mensagem e do número de estações escravas. Nos casos em que são permitidas retransmissões, T_{SR} é calculado conforme a fórmula abaixo:

$$T_{SR} = np \cdot T_{MC} + mp \cdot RET \cdot T_{MC} \quad (25)$$

onde:

np : número de estações escravas

mp : número de retransmissões de mensagens por ciclo de varredura

$RET \cdot T_{MC}$: tempo do ciclo de retransmissão de mensagem

8.6 DESENVOLVIMENTO DAS EQUAÇÕES

8.6.1 Cálculo de t_{MC}

Partindo da equação (22):

$$T_{MC} = T_{S/R} + T_{SDR} + T_{A/R} + T_{ID} + 2.T_{TD} \quad (22)$$

$$T_{S/R} = a.11 \text{ bits} \quad (18)$$

; onde a é o número de caracteres de um quadro Envio/Requisição.

$$T_{A/R} = b.11 \text{ bits} \quad (19)$$

; onde b é o número de caracteres de um quadro Reconhecimento/Resposta.

Substituindo (18) e (19) na equação (22)

$$T_{MC} = a.11 + T_{SDR} + b.11 + T_{ID} + 2.T_{TD}$$

$$T_{MC} = a.11 + (t_{SDR} / t_{BIT}) + b.11 + (t_{ID} / t_{BIT})$$

$$T_{MC} = a.11 + t_{SDR} . T_X + b.11 + t_{ID} . T_X$$

$$T_{MC} = (a + b) 11 + (t_{SDR} + t_{ID}) T_X$$

$$(t_{MC} / t_{BIT}) = (a + b) 11 + (t_{SDR} + t_{ID}) T_X$$

$$t_{MC} = [(a + b)11 + (t_{SDR} + t_{ID}) T_X] / T_X \quad (26)$$

8.6.2 Cálculo de R_{SYS}

$$R_{SYS} = 1/t_{MC} \quad (23)$$

8.6.3 Cálculo de t_{SR}

Partindo da equação (25):

$$T_{SR} = np . T_{MC} + mp . RET T_{MC} \quad (25)$$

$$RET = 0$$

$$T_{SR} = np . T_{MC}$$

$$(t_{SR} / t_{BIT}) = np . (t_{MC} / t_{BIT})$$

$$t_{SR} = np . t_{MC} \quad (27)$$

8.7 VALORES DA TABELA A.1 DA NORMA PROFIBUS

8.7.1 Considerações

Na tabela A.1 da Norma PROFIBUS DIN 19245 - Primeira Parte , página 127, são apresentados os valores para o tempo de ciclo de mensagens, a taxa de

mensagens e o tempo de reação de sistema em dependência da velocidade de transmissão, da DATA_UNIT e da quantidade de estações passivas.

Estes valores foram calculados, segundo a Norma, a partir das equações (22), (23) e (25). Na Tabela 7 estão listados os valores para uma taxa de 19200 bps.

Tabela 7. Tabela A.1 da Norma Profibus para 19200 bps

DATA_UNIT	t_{MC} (ms)	R_{SYS}	T_{SR} (ms)
2	12	82	364
10	17	60	502
50	40	25	1200

Como base para o cálculo teórico serão utilizadas as equações desenvolvidas no item 8.6 junto com os valores e considerações sugeridos no Apêndice 6 da primeira parte da Norma Profibus conforme listados a seguir:

- Os tempos T medidos em bits, foram convertidos em t (segundos) e os valores estabelecidos conforme a seguir:

$$t_{SDR} = 0,5 \text{ ms}; t_{ID} = 1 \text{ ms}$$

- As repetições das mensagens e o tempo T_{TD} (Transmission Delay Time) não são considerados.

- O Tempo de Transferência de Token é considerado irrelevante, já que há apenas uma estação mestre no sistema.

- DATA_UNIT = 2, 10, 50 octetos.

- Trinta estações escravas, $n_p = 30$.

- Os quadros de requisição são sem unidades de dados enquanto os quadros de resposta possuem unidade de dados variáveis. Ou seja $a = 6$ e $b = 9 + \text{DATA_UNIT}$.

Pelas considerações acima temos:

$$a = 6$$

$$b = 9 + \text{DATA_UNIT}$$

$$np = 30$$

$$t_{SDR} = 0,5 \text{ ms}$$

$$t_{ID} = 1 \text{ ms}$$

Os cálculos serão feitos para uma taxa $T_X = 19200 \text{ bps}$

Substituindo na equação (26):

$$t_{MC} = [(a + b)11 + (t_{SDR} + t_{ID}) T_X] / T_X \quad (26)$$

$$t_{MC} = [(6 + 9 + \text{DATA_UNIT})11 + (5 \cdot 10^{-4} + 1 \cdot 10^{-3}) 19200] / 19200$$

$$t_{MC} = [(6 \cdot 11 + 9 \cdot 11 + \text{DATA_UNIT} \cdot 11 + 1,5 \cdot 10^{-3} \cdot 19200)] / 19200$$

$$t_{MC} = [(66 + 99 + \text{DATA_UNIT} \cdot 11 + 28,8)] / 19200$$

$$t_{MC} = (193,8 + \text{DATA_UNIT} \cdot 11) / 19200$$

$$R_{SYS} = 1/t_{MC}$$

Substituindo na equação (27):

$$t_{SR} = np \cdot t_{MC} \quad (27)$$

$$t_{SR} = 30 t_{MC}$$

Tabela 8. Resultado dos Cálculos Teóricos

DATA_UNIT	t_{MC} (ms)	R_{SYS}	t_{SR} (ms)
2	11,24	88,97	337,19
10	15,82	63,20	474,69
50	38,74	25,81	1162,19

A partir da comparação dos valores da Tabela 7 e da Tabela 8 concluímos que as fórmulas desenvolvidas no item 8.6 podem também ser validadas e utilizadas para análises dos dados obtidos nos testes práticos da presente dissertação.

9. TESTES REALIZADOS

9.1 DESCRIÇÃO

Para avaliar o rendimento da rede foram realizados testes práticos utilizando várias estações escravas ligadas a um computador que desempenhava a função de mestre administrando a comunicação e um outro computador a fim de monitorar o tráfego na rede.

9.2 ESTAÇÃO MESTRE

Foi utilizado como mestre um Pentium de 166 Mhz, com 16MB de RAM e 1.3 GB de disco rígido.

9.3 SISTEMA DE MONITORAÇÃO

Para monitorar a rede foi utilizada uma placa da Technologie Management Gruppe (TMG) conectada no barramento de um computador Pentium 100 Mhz, 16M de RAM e 4GB de disco rígido.

Esta placa possui um microcontrolador 8051 e serve para conectar um PC a uma rede PROFIBUS. A placa tem um barramento de dados de 8 bits e pode operar em computadores 286, 386 e 486 em ambiente operacional MS-DOS [TMG-2].

A fim de ativar os parâmetros da placa o programa “lade.exe” deve ser executado. Este programa acompanha a placa assim como o programa “Profibus Bus Monitor”, utilizado para adquirir os dados do barramento.

O Bus Monitor é uma estação transparente para as outras estações do barramento. É uma ferramenta que recebe, analisa e mostra as mensagens no barramento. Desta forma tem duas funções: a primeira, gravar os quadros (medição) e a segunda analisar os quadros recebidos [TMG-1].

Este sistema de monitoração foi adquirido pelo laboratório para validar testes nos quais, como no caso do presente trabalho, o protocolo PROFIBUS implementado no IEE fosse utilizado.

9.4 ESTAÇÕES ESCRAVAS

As estações escravas são terminais microprocessados elaborados no projeto “Votador Eletrônico” desenvolvido no IEE.

As características de hardware são: processador da família do 8051, flash EPROM, cristal de 11.0592MHz, 128 bytes de RAM.

O sistema de Voto Eletrônico (VE) é utilizado em anfiteatros ou plenários. Durante as conferências a platéia tem a oportunidade de participar respondendo às perguntas, realizadas pelo conferencista, através do voto.

Cada terminal conhecido como VE, apresenta um teclado com cinco opções para as resposta além de uma tecla CANCELA e dois leds para indicar o estado de operação do sistema.

O led de cor vermelha fica aceso enquanto o assunto tratado na conferência é apresentado e também enquanto as perguntas a serem respondidas estão sendo realizadas. Este led em vermelho significa por tanto, que o sistema de Voto Eletrônico não está habilitado apesar de estar ligado.

Assim que a pergunta é formulada, as pessoas presentes na platéia terão um determinado tempo, que pode ser estipulado em alguns minutos, para escolher a resposta entre as cinco alternativas representadas pelas teclas marcadas com as letras A, B, C, D e E. Durante este período o led vermelho fica piscando.

Quando uma tecla é apertada o led verde acende indicando que uma escolha já foi feita. Se for necessário trocar a opção basta apertar a tecla CANCELA, o led verde desliga novamente, e é possível realizar uma nova escolha apertando as teclas alfa. A opção selecionada ainda não é enviada para o computador, ela está armazenada na memória do VE.

Quando o led vermelho pára de piscar significa que o prazo para responder está esgotado e a última escolha feita será transmitida para o computador mestre.

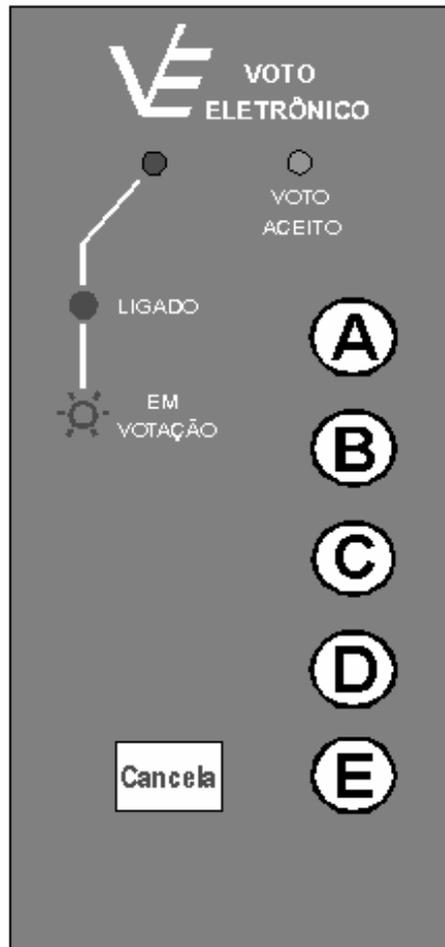


Figura 18. Terminal para Voto Eletrônico

É neste instante que o PC realiza uma pesquisa varrendo todas os VE ligados e coletando as respostas. De posse destes dados uma rotina deste programa de administração da rede realiza uma estatística dos resultados que é apresentada à platéia. Com base nesta estatística o conferencista tem condições de saber quais os pontos que foram melhor entendidos e quais os que ainda faltam ser discutidos.

Tabela 9 - Estados do VE

Estado do VE	Led Vermelho	Led Verde
Espera	Ligado	Desligado
Votação	Piscante	
Nenhuma opção foi escolhida	Piscante	Desligado
Alguma opção foi escolhida	Piscante	Ligado
Opção Cancelada	Piscante	Desligado
Coleta da votação	Ligado	Ligado

O sistema de Voto Eletrônico, que utiliza para comunicação o protocolo PROFIBUS, já foi utilizado em aplicações práticas com até 110 estações ligadas a um PC através de 3 repetidores.

9.5 PROGRAMA DE TESTES

Para realizar os testes que serão analisados foi utilizada a rede do Sistema de Voto Eletrônico, sendo que o programa foi modificado para simular uma rede de consulta conforme proposto no presente trabalho.

Seguindo a filosofia descrita no Capítulo 1, o programa que roda no mestre realiza pesquisas de forma seqüencial varrendo todas as estações presentes na rede. Quando uma estação é consultada enviará, caso exista, uma solicitação que será imediatamente atendida pelo mestre. O ciclo completo de intercâmbio de dados está constituído por quatro quadros de comunicação. O primeiro quadro é um serviço de “envio e requisição de dados com confirmação” (SRD Send and Request Data with Reply) do mestre de endereço 01 para, por exemplo, a estação de número 03:

SD1	DA	SA	FC	FCS	ED
10	03	01	7C	80	16

A estação 03 responde com um serviço de resposta de SRD, com a requisição incluída na área de dados, conforme a seguir:

SD2	LE	LEr	SD2	DA	SA	FC	DATA Unit	FCS	ED
68	08	08	68	01	03	08	01 00 00 00 00	0C	16

O mestre utiliza um quadro de “envio de dados com confirmação” (SDA Send Data with Acknowledge) para enviar os dados solicitados pela estação 03:

SD2	LE	LEr	SD2	DA	SA	FC	DATA Unit	FCS	ED
68	08	08	68	03	01	53	00 00 00 00 00	57	16

A estação 03 responde com um frame de confirmação, ACK, se o serviço foi recebido sem erros:

SD1	DA	SA	FC	FCS	ED
10	01	03	00	04	16

Terminado este ciclo de quatro quadros o mestre passa para a seguinte estação e assim sucessivamente até todas serem consultadas e volta a primeira novamente. Numa aplicação prática as estações escravas nem sempre terão solicitações a fazer por tanto o ciclo de comunicação ficaria reduzido a dois quadros, sendo desnecessário o envio do serviço de SDA por parte do mestre, nestes casos a carga de tráfego na rede será menor. Para os testes esta carga foi considerada máxima estipulando que todas as vezes que as estações fossem consultadas seria realizada uma solicitação.

9.6 TESTES A 19200 BPS

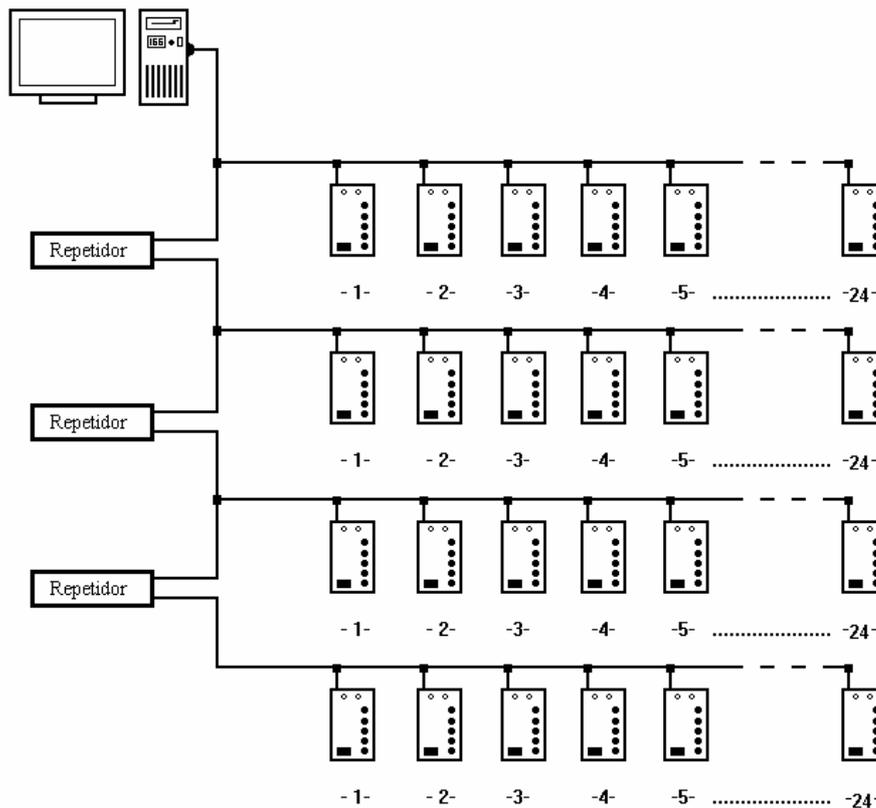


Figura 19. Rede de Testes

Foi realizada uma primeira bateria de testes na qual foram utilizadas 96 estações escravas e 3 repetidores comunicando-se a uma taxa de 19200 bps. A

configuração da rede ficou conforme figura acima. Cada um dos braços formados por 24 estações escravas foram sendo desligados sucessivamente para realizar testes com diferentes números de terminais. Assim se obteve resultados com 96, 48, 24 e 12 estações escravas. Para cada uma das situações anteriores foram feitas comunicações com quadros de tamanho variável com área de dados de 2, 10, 20, 30 e 40 bytes.

Na foto abaixo aparece a estante onde foram colocados os VE's durante os testes, distribuídos em grupos de 12 por prateleira.



Figura 20 – A estante com os VE's durante os testes

9.7 TESTES A 28800 BPS

Para melhorar o desempenho obtido foi realizado uma segunda bateria de testes a uma taxa de 28800 bps. Além do aumento da taxa, outras modificações foram feitas, como por exemplo a retirada dos repetidores que por serem muito lentos provocavam um atraso nas comunicações. Para evitar a utilização de repetidores e considerando que a Norma PROFIBUS permite ligar até 32 estações, passivas e ativas,

sem repetidores, neste segundo teste foram ligados 31 VE escravos além do PC que desempenha o papel de mestre na rede. Com estas modificações foi possível realizar análises para conjuntos com 31, 24 e 12 estações escravas. A área de dados utilizada nas comunicações foi de 5, 10, 20, 30 e 40 bytes.



Figura 21 – Os VE's nas prateleiras

10. ANÁLISE DOS TESTES

10.1 CÁLCULO DOS VALORES TEÓRICOS A 19200 BPS

10.1.1 Considerações

Neste item será realizado o cálculo teórico dos tempos envolvidos na comunicação mestre-escravos para posteriormente realizar-se uma comparação com os resultados obtidos nos testes práticos. A seguir são apresentadas as considerações feitas para obter estes valores teóricos:

- Os tempos T medidos em bits, foram convertidos em t (segundos).
- O tempo de reação da estação que transmite uma confirmação ou resposta (escravo), t_{SDR} , varia entre 2,5 ms e 5 ms.
- O tempo de reação da estação que transmite uma requisição ou *token* (mestre), t_{ID} , varia entre 12 ms e 18 ms.
- Para o cálculo teórico serão considerados os piores casos para os tempos de t_{SDR} (5 ms) e t_{ID} (18 ms).
- As repetições das mensagens e o tempo T_{TD} (Transmission Delay Time) não são considerados.
- O Tempo de Transferência de Token é considerado irrelevante, já que há apenas uma estação mestre no sistema.
- Taxa de transmissão utilizada: 19,2 Kbit/s
- $DATA_UNIT = 2, 10, 20, 30$ e 40 octetos.
- Número de estações escravas, $n_p = 12, 24, 48, 72$ e 96.
- O ciclo de comunicação é composto pela seguinte seqüência de mensagens: o mestre utiliza o serviço SRD (Send and Request Data with Reply) para ler as informações contidas no buffer do escravo. Em seguida, o mestre decodifica estes

dados, interpreta a solicitação e envia a resposta utilizando o serviço SDA (Send Data with Acknowledge).

- O ciclo de SRD é composto por um quadro de requisição sem unidade de dados e um quadro de resposta com área de dados variável.

$$a + b = 6 + (9 + \text{DATA_UNIT})$$

- O ciclo de SDA é composto por um quadro de envio de dados com área de dados variável e um ACK como resposta.

$$a + b = (9 + \text{DATA_UNIT}) + 6$$

Pelas considerações acima temos:

$$a + b = 6 + 9 + \text{DATA_UNIT}$$

$$t_{\text{SDR}} = 5 \text{ ms}$$

$$t_{\text{ID}} = 18 \text{ ms}$$

$$T_X = 19200 \text{ bit/s}$$

$$t_{\text{MC}} = [(a + b)11 + (t_{\text{SDR}} + t_{\text{ID}}) T_X] / T_X \quad (26)$$

$$t_{\text{MC}} = (606,6 + 11.\text{DATA_UNIT})/19200$$

$$\text{RSYS} = 1/t_{\text{MC}} \quad (23)$$

Na tabela abaixo são apresentados os resultados teóricos obtidos com os valores utilizados nos testes

Tabela 10. Resultados Teóricos – Tx 19200 bps

DATA_UNIT	t _{MC} (ms)	R _{SYS} Arredondado
2	32,73	31
10	37,30	27
20	43,10	23
30	48,80	20
40	54,50	18

10.2 VALORES MEDIDOS A 19200 BPS

Para cada teste prático realizado, conforme descrito no capítulo anterior, foi gerado um arquivo com uma lista de valores semelhante à apresentada no Anexo C.

Cada um destes arquivos contém aproximadamente 24.000 quadros de comunicação trocados entre a estação mestre e as escravas. A partir destes quadros foram obtidos os tempos dos ciclos de mensagens SRD e SDA e o tempo de reação do sistema. O tempo gasto para executar cada uma das partes que constituem os ciclos estão tabuladas no Anexo A, no qual se encontram o tempo máximo, mínimo, médio e desvio padrão para cada um dos casos.

10.2.1 Análise do Ciclo SRD a 19200 bps

O ciclo SRD é formado pelos tempos descritos a seguir:

1. $t_{S/R}$, tempo do quadro de consulta enviado pelo mestre para uma estação, com tamanho fixo de 6 bytes. O tempo gasto para a transmissão deste quadro é fixo independente do número de estações na rede e o seu valor medido corresponde com o valor calculado:

$$T_{S/R} = a.11 \text{ bits} \quad (28)$$

$$t_{S/R} = 6.11/19200$$

$$t_{S/R} = 3437 \mu s$$

2. $t_{SDR} + 2 t_{TD}$, representa o tempo transcorrido entre o último byte da requisição e o primeiro da resposta da estação para o mestre.

3. $t_{A/R}$, tempo do quadro de resposta da estação com área de dados variável. O tempo gasto para a transmissão deste quadro é independente do número de estações na rede e está definido pelo tamanho da área de dados mais os 9 bytes de controle de um quadro variável:

$$T_{A/R} = b.11 \text{ bits} \quad (29)$$

$$DATA_UNIT = 2$$

$$t_{A/R} = (9+2).11/19200$$

$$t_{A/R} = 6302 \mu s$$

$$DATA_UNIT = 10$$

$$t_{A/R} = (9+10).11/19200$$

$$t_{A/R} = 10885 \mu s$$

$$DATA_UNIT = 20$$

$$t_{A/R} = (9+20).11/19200$$

$$t_{A/R} = 16614 \mu s$$

$$\text{DATA_UNIT} = 30$$

$$t_{A/R} = (9+30).11/19200$$

$$t_{A/R} = 22343 \mu\text{s}$$

$$\text{DATA_UNIT} = 40$$

$$t_{A/R} = (9+40).11/19200$$

$$t_{A/R} = 28072 \mu\text{s}$$

4. $t_{ID} + t$, tempo entre a chegada da resposta da estação na estação mestre e o envio do quadro para a próxima estação presente na rede por parte do mestre. É importante destacar que este tempo envolve também os tempos de transmissão dos quadros de token e de pesquisa das estações presentes na rede.

Tabela 11. Taxa:19200 - 12 Estações - Ciclo SRD (μs)

DATA_UNI T	2	10	20	30	40
1. $t_{S/R}$	3 437	3 437	3 437	3 437	3 437
2. $t_{SDR} + 2 t_{TD}$	5 482	5 505	5 874	5 792	6 093
3. $t_{A/R}$	6 302	10 885	16 614	22 343	28 072
4. t_{ID}	12 660	13 253	13 209	13 193	12 583

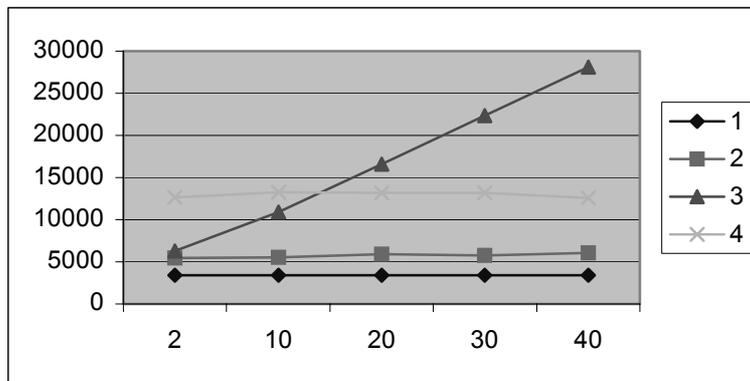


Figura 22. Gráfico referente à Tabela 11

Tabela 12. Taxa:19200 - 24 Estações - Ciclo SRD (μ s)

DATA_UNI T	2	10	20	30	40
1. $t_{S/R}$	3 437	3 437	3 437	3 437	3 437
2. $t_{SDR} + 2 t_{TD}$	5 445	5 560	5 593	5 897	5 960
3. $t_{A/R}$	6 302	10 885	16 614	22 343	28 072
4. t_{ID}	12 753	12 960	14 242	13 483	14 781

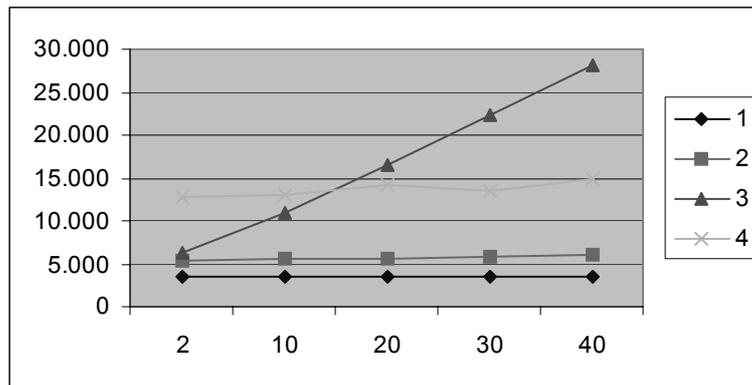


Figura 23. Gráfico referente à Tabela 12

Tabela 13. Taxa:19200 - 48 Estações - Ciclo SRD (μ s)

DATA_UNI T	2	10	20	30	40
1. $t_{S/R}$	3 437	3 437	3 437	3 437	3 437
2. $t_{SDR} + 2 t_{TD}$	5 537	5 553	5 384	5 846	6 438
3. $t_{A/R}$	6 302	10 885	16 614	22 343	28 072
4. t_{ID}	13 697	13 467	13 670	12 380	14 019

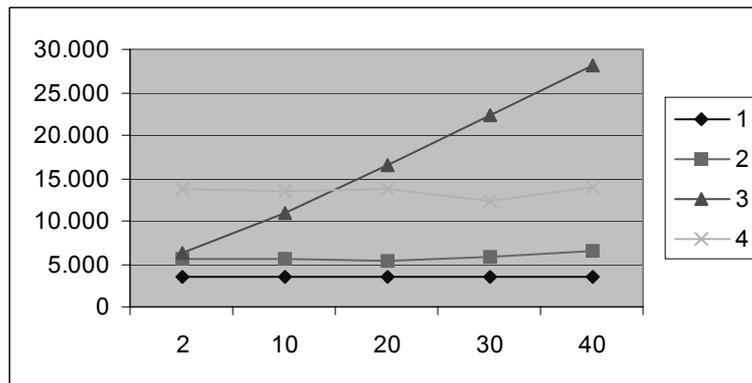


Figura 24. Gráfico referente à Tabela 13

Tabela 14. Taxa:19200 - 72 Estações - Ciclo SRD (μ s)

DATA_UNI T	2	10	20	30	40
1. $t_{S/R}$	3 437	3 437	3 437	3 437	3 437
2. $t_{SDR} + 2 t_{TD}$	5 574	5 827	5 398	6 195	6 377
3. $t_{A/R}$	6 302	10 885	16 614	22 343	28 072
4. t_{ID}	12 570	12 805	13 498	12 387	14 251

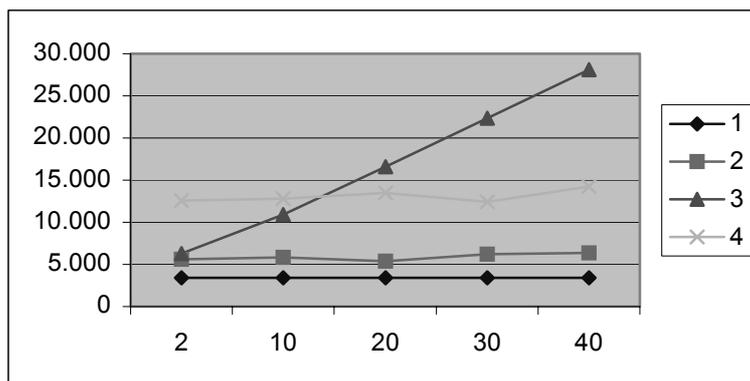


Figura 25. Gráfico referente à Tabela 14

Tabela 15. Taxa:19200 - 96 Estações - Ciclo SRD (μ s)

DATA_UNI T	2	10	20	30	40
1. $t_{S/R}$	3 437	3 437	3 437	3 437	3 437
2. $t_{SDR} + 2 t_{TD}$	5 600	5 166	5 903	6 429	5 853
3. $t_{A/R}$	6 302	10 885	16 614	22 343	28 072
4. t_{ID}	13 710	12 636	12 467	12 606	14 937

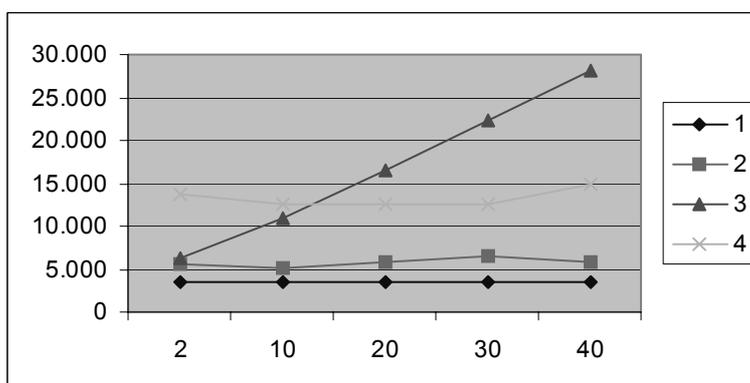


Figura 26. Gráfico referente à Tabela 15

A partir das tabelas e gráficos anteriores pode-se concluir que:

- os tempos de transmissão dos quadros ($t_{S/R}$ e $t_{A/R}$) dependem da taxa de transmissão e do tamanho do área de dados (DATA_UNIT) exclusivamente;
- os tempos entre quadros (t_{SDR} , t_{TD} e t_{ID}) dependem do hardware da rede e da forma como foi implementado o protocolo.

10.2.2 Tempo do Ciclo de Mensagem para um Ciclo SRD

O tempo de ciclo de mensagem t_{MC} (ms) representado pela soma dos tempos que compõem um ciclo SRD para cada uma das situações de teste se encontra na Tabela 16.

$$T_{MC} = T_{S/R} + T_{SDR} + T_{A/R} + T_{ID} + 2 \cdot T_{TD} \quad (22)$$

Tabela 16- Taxa: 19200 bps - Tempo do Ciclo SRD (ms)

DATA_UNIT	12 estações	24 estações	48 estações	72 estações	96 estações	Valor Teórico
2	27.88	27.94	28.97	27.88	29.05	32.73
10	33.08	33.09	33.34	32.95	32.12	37.30
20	39.13	39.89	42.19	38.95	38.42	43.10
30	44.77	45.16	44.01	44.36	44.82	48.80
40	50.19	52.25	51.97	52.14	52.30	54.50

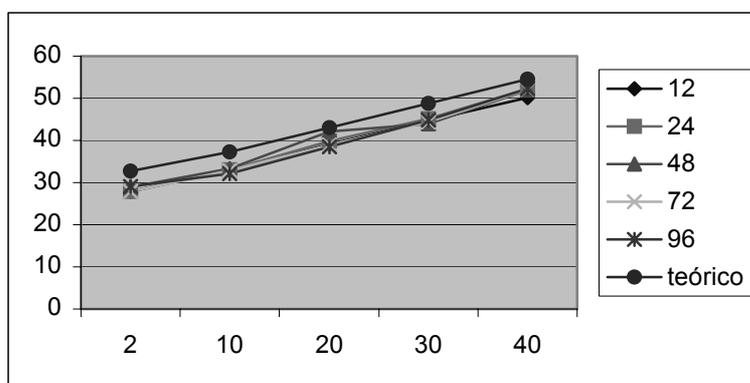


Figura 27. Gráfico referente à Tabela 16

10.2.3 Tempo do Ciclo de Mensagem para um Ciclo SDA

No Anexo B é feita uma avaliação semelhante para o ciclo de mensagem SDA, os resultados são apresentados na Tabela a seguir.

Tabela 17 – Taxa: 19200 bps - Ciclo SDA (ms)

DATA_UNIT	12 estações	24 estações	48 estações	72 estações	96 estações	Valor Teórico
2	29.98	29.93	29.90	29.99	29.90	32.73
10	34.97	34.93	34.88	35.00	34.99	37.30
20	39.35	39.99	39.99	39.88	39.99	43.10
30	46.32	46.41	46.13	46.36	46.15	48.80
40	53.86	53.54	54.23	53.98	54.21	54.50

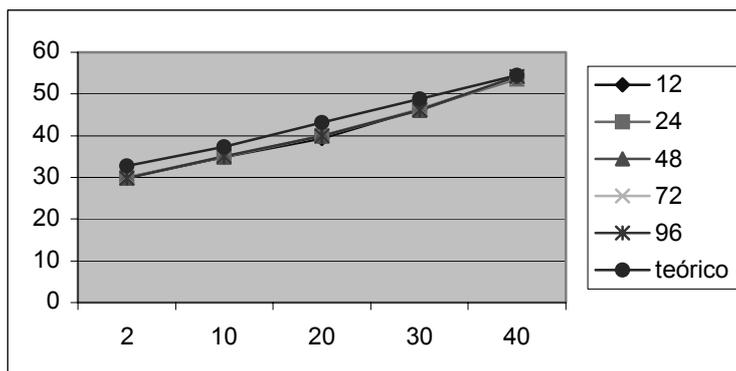


Figura 28. Gráfico referente à Tabela 17

Na Tabela 16 e na Tabela 17 agregada uma coluna com os resultados da Tabela 10, a fim de facilitar a comparação entre os resultados práticos e os resultados teóricos.

Nos gráficos gerados por estas tabelas observa-se que as curvas que representam os tempos medidos coincidem entre si o que permite concluir que o tempo de ciclo de mensagem:

- independe do número de estações envolvidas;
- aumenta em função direta do tamanho da área de dados transmitida.

Além disso, a curva que representa os valores teóricos está um pouco acima das que representam os valores medidos, o que se explica, pois essa curva foi calculada considerando os piores casos para os tempos de t_{SDR} e t_{ID} .

10.3 TEMPO DE CICLO DE MENSAGEM A 28800 BPS

Os dados obtidos nos testes realizados para uma taxa de 28800 bps, cujos resultados e análises são apresentados no Anexo B, geram a Tabela 16 e Tabela 17 a partir destas tabelas e seus gráficos comprovamos novamente que o tempo de ciclo de mensagem independe do número de estações envolvidas e aumenta em função direta do tamanho da área de dados transmitida.

10.3.1 Tempo do Ciclo de Mensagem para um Ciclo SRD

Tabela 18 – Taxa: 28800 bps - Tempo do Ciclo SRD (ms)

DATA UNIT	12 estações	24 estações	31 estações	Valor teórico
5	18.99	18.00	19.40	22.6
10	22.18	20.46	20.86	24.5
20	25.07	23.99	25.28	28.4
30	28.40	28.18	28.67	32.2

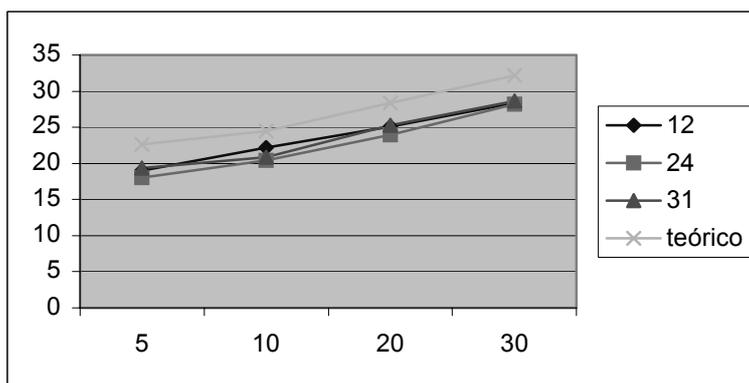


Figura 29. Gráfico referente à Tabela 18

10.3.2 Tempo do Ciclo de Mensagem para um Ciclo SDA

Tabela 19 - Taxa: 28800 bps – Tempo do Ciclo SDA (ms)

DATA UNIT	12 estações	24 estações	31 estações	Valor teórico
5	18.00	18.00	18.00	22.6
10	20.99	20.99	21.00	24.5
20	23.99	24.00	23.98	28.4
30	28.92	29.02	29.05	32.2

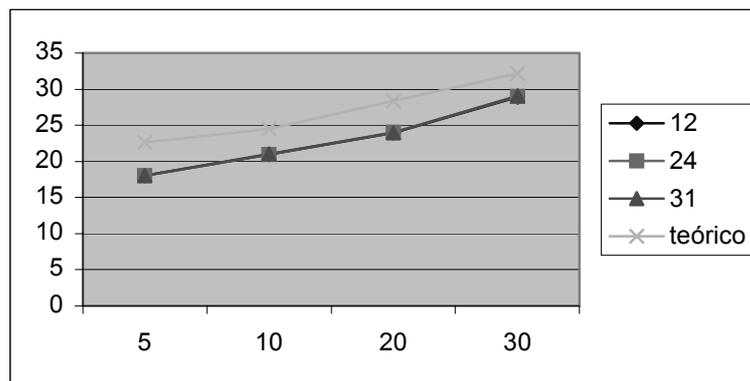


Figura 30. Gráfico referente à Tabela 19

11. CONCLUSÕES

Conforme descrito no capítulo 9 item 5, para realizar os testes foi utilizada a rede do Sistema de Voto Eletrônico, com um programa que simula uma rede de consulta. A comunicação entre o mestre e cada uma das estações escravas é composta por um ciclo SRD e um ciclo SDA. Estes dois ciclos permitem que as estações escravas recebam a resposta (através do ciclo SDA) a uma solicitação realizada (pelo ciclo SRD). O mestre pesquisa seqüencialmente as estações respondendo a cada uma das solicitações.

Numa aplicação prática as estações escravas nem sempre farão solicitações, portanto o ciclo de comunicação se reduzirá a um serviço de SRD e nestes casos a carga de tráfego na rede seria menor. Durante os testes esta carga foi considerada máxima e, para tanto, estipulado que todas as vezes que as estações fossem consultadas realizariam uma solicitação.

Outra condição estabelecida nos testes é que a área de dados (Data_Unit) tem o mesmo tamanho tanto no quadro da pergunta (ciclo SRD) quanto no de resposta (ciclo SDA) conforme descrito no item 10.2.

Os tempos apresentados na Tabela 20, para uma taxa de 19200 bps, e na Tabela 21, para 28800 bps, indicam qual o tempo médio que uma estação deve aguardar pela resposta (Tempo de Reação do Sistema). A pior situação é quando a estação realiza uma solicitação assim que foi consultada e portanto deverá esperar que o mestre pesquise primeiramente todas as estações presentes na rede até ser novamente consultada. Estes resultados foram obtidos a partir das tabelas apresentadas no Anexo A.

O tempo aceitável de espera da estação por uma resposta pode ser maior ou menor dependendo da aplicação considerada.

Tabela 20 – Tempo de Reação do Sistema a 19200 bps – (ms)

DATA UNIT / Ciclo	12 estações	24 estações	48 estações	72 estações	96 estações
2	644	1 343	2 726	4 145	5 536
10	750	1 584	3 196	4 791	6 497
20	874	1 827	3 730	5 565	7 515
30	1 011	2 100	4 274	6 485	8 723
40	1 163	2 427	4 887	7 385	10 002

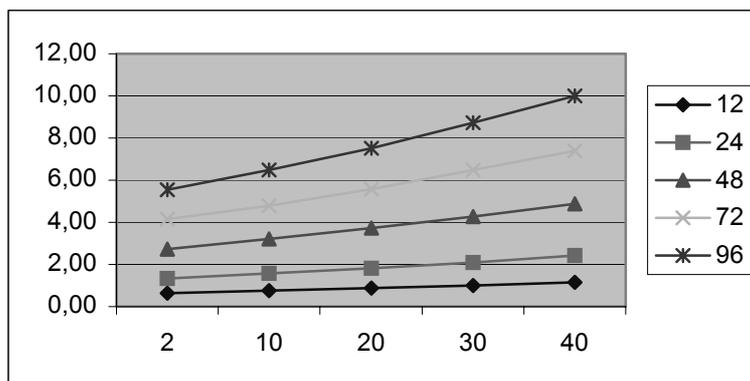


Figura 31. Gráfico referente à Tabela 20

Tabela 21 – Tempo de Reação do Sistema a 28800 bps – (ms)

DATA UNIT / Ciclo	12 estações	24 estações	31 estações
5	416	858	1 111
10	478	986	1 284
20	554	1 137	1 475
30	643	1 327	1 735

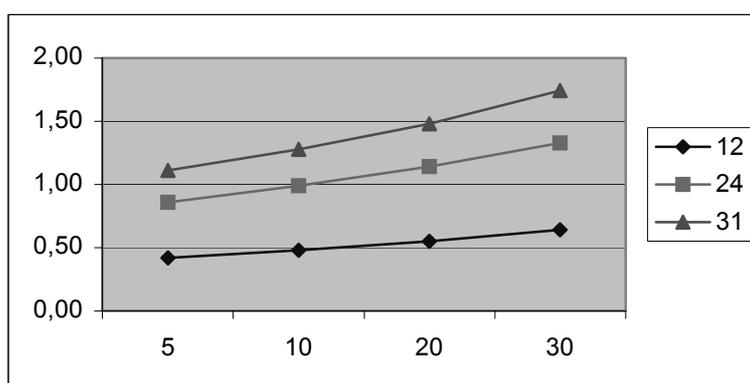


Figura 32. Gráfico referente à Tabela 21

No caso do Sistema de Votação Eletrônica as votações são coletadas em uma única seqüência e o usuário não espera uma resposta do mestre. Este tempo não é crítico e na prática o sistema funcionou sem problemas a uma taxa de 19200 bps com 110 VE's ligados a um único computador.

Em outras aplicações, no entanto, o tempo em que o usuário espera por uma resposta pode determinar ou impedir a viabilidade do sistema. Segue abaixo, a título de exemplo, uma aplicação de consulta muito conhecida na área comercial: a dos caixas de supermercado.

Este sistema funciona da seguinte maneira: o funcionário do caixa digita o código de um produto ou passa o código de barras do mesmo na frente do feixe de um leitor ótico (scanner), este código vai é enviado ao computador central cujo programa devolve para o caixa o nome e preço do produto. Estes dados são impressos numa impressora fiscal que possui um rolo de papel onde é possível imprimir até quarenta colunas. Portanto tem-se os 13 caracteres do código (EAN13) sendo enviados do caixa para o PC e aproximadamente 40 caracteres, com o nome e preço do produto, vindos do PC para o caixa. Ao todo são 53 caracteres, assim este sistema seria comparável ao sistema testado para o caso de DATA_UNIT com 30 bytes (Tabela 20 e Tabela 21), lembrando que neste caso os dados somam ao todo 60 bytes durante a comunicação. Os resultados são repetidos na Tabela 22.

Tabela 22 – Tempo de Reação do Sistema – Data_Unit = 30 bytes / Ciclo – (ms)

Taxa	12 estações	24 estações	31 estações	48 estações	72 estações	96 estações
19 200	1 011	2 100	-	4 274	6 485	8 723
28 800	643	1 327	1 735	-	-	-

Pode-se observar nesta Tabela que o tempo de Reação do Sistema, ou seja o tempo que o usuário espera por uma resposta, varia entre um 1,01 s e 8,72 s para uma taxa de 19200 bps e entre 0,64 s e 1,74 s para uma taxa de 28800 bps. Baseados nestes tempos deve-se encontrar uma forma para analisar a viabilidade neste tipo de aplicação.

Será considerada uma experiência realizada com o objetivo de determinar os benefícios resultantes da instalação de scanners de códigos de barras em filas de caixas registradoras em condições normais, feita por um grupo da área de supermercados a fim

de conferir a conveniência de possuir scanners nos seus pontos de venda. E cuja metodologia e resultados estão descritos no Anexo E.

Conforme esta experiência o tempo gasto para fazer o registro de cada item pelo teclado foi de 4,2 s e pelo scanner de 2,2 s. Observando os valores da Tabela 22 pode-se concluir que para uma taxa de 19200 bps a rede poderia ser viável se composta por até 24 estações, já que o tempo medido neste caso foi de 2,1 s. Para uma taxa de 28800 bps seria possível ter até 31 estações e ainda o tempo ficaria abaixo dos 2,2 s estabelecidos na pesquisa citada.

Em função dos resultados pode-se concluir que a rede Profibus pode ser utilizada em aplicações nas quais o número de estações envolvidas não seja elevado ou que se utilizem taxas de comunicação maiores.

Os valores listados na Tabela 20 e na Tabela 21 também podem ser utilizados para estabelecer a viabilidade do uso desta rede para qualquer outro tipo de aplicação.

12. BIBLIOGRAFIA

- [AGU89] Aguiar, M.W. Costa. Análise Comparativa de Desempenho da Camada Enlace de Dados do PROFIBUS e do FIP, Propostas Candidatas ao Padrão "Field-Bus". Florianópolis, CPG-SC, 1989.
- [BEN93] Bender, K. PROFIBUS The Fieldbus for Industrial Automation. Prentice Hall, 1993.
- [BER89] Bernard, John W., CIM The Process Industries Instrument Society of America, 1989.
- [DIN91] DIN 19245 PROFIBUS Standard. Germany, 1991.
- [ERD94] Erdei, Guillermo E., Código de Barras - Desenvolvimento, Impressão e Controle de Qualidade, São Paulo, Makron Books, 1994.
- [GAL93] Gallard, Raúl e Aguiar, Paulo, Redes de Computadoras. VI ESCUELA BRASILEÑO ARGENTINA DE INFORMÁTICA, 1993.
- [GIO86] Giozza, William Ferreira. Redes Locais de Computadores: tecnologia e aplicações. Editora McGraw-Hill Ltda, 1986.
- [KUS88] Kusiak, Andrew. Artificial Intelligence Implications for CIM Editora IFS (Publications) Ltd., 1988.
- [MEN89] Mendes, Manuel de Jesus, Comunicação Fabril e o Projeto MAP/TOP. IV ESCUELA BRASILEÑO ARGENTINA DE INFORMÁTICA Editora Kapelusz S.A., Buenos Aires, Argentina, 1989.
- [MEN91] Mendes, M.de J. Redes de Chão de Fábrica e o Projeto Fieldbus. CERTI-SC, Florianópolis, 1991.
- [MOM86] Sistemas CAD/CAM/CAE - Diseño y Fabricación por Computador (por varios autores bajo la coordinación de José Mompín Poblet, diretor dela revista "Mundo Electrónico"). Editora Marcombo, 1986.

- [PRO92] PROFIBUS - Technical Documentation , PROFIBUS User Organization, Germany,1992
- [SAI92] Saikoski, Katia Barbosa, Estudo e Especificação de Serviços para Auxílio à Gerência de Redes em Ambiente Industrial, Porto Alegre, CPGCC, 1992
- [SOA95] Luiz Fernando Gomes Soares, Guido Lemos, Sérgio Colcher, Redes de Computadores - Das LANs MANs e WANs às Redes ATM. Editora Campus, 1995.
- [TAN94] Tanenbaum, Andrew S., Redes de Computadores. Editora Campus, 1994.
- [TMG-1] PROFIBUS - Bus Monitor - User Documentation - Release 6.0, Editora Technologie Management Gruppe(TMG).
- [TMG-2] PROFIBUS - PC Card - Hardware Documentation - Release 6.0, Editora Technologie Management Gruppe (TMG).
- [VOL97] Dipl. Ing. M. Volz, PROFIBUS - Technical Description, PROFIBUS User Organization, Germany,1997
- [ZUC95] Zuccolotto, M. Sistema de Desenvolvimento para Instrumentos Inteligentes. Trabalho proposto ao Congresso da ISA SHOW BRASIL'95

A. ANEXO A - RESULTADOS DOS TESTES PRÁTICOS

Neste anexo são apresentadas as tabelas contendo os resultados obtidos nos testes práticos realizados.

Para cada tempo considerado foram calculados os tempos máximo, mínimo, média e desvio padrão. Cada tabela é o resultado do análises de um arquivo com aproximadamente 24.000 quadros de comunicação entre a estação mestre e as escravas. Um exemplo destes arquivos é listado no anexo C.

A.1. RESULTADOS A 19200 BPS – 12 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 19200 – Estações: 12 – Data_Unit: 2 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	643 659	684 857	331 567	20 662
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 482	8 070	3 028	1 475
Resposta de SRD	3. $t_{A/R}$	6 302	6 302	6 302	0
	4. $t_{ID} + t$	12 660	44 787	10 098	2 195
Ciclo de DAS	1. $t_{S/R}$	6 302	6 302	6 302	0
	2. $t_{SDR} + 2 t_{TD}$	7 008	8 326	3 298	728
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 232	14 428	11 932	707

Soma do ciclo SRD	27 882
Soma do ciclo DAS	29 979
Total (SRD + SDA)	57 861

Tabela A.2. Taxa: 19200 – Estações: 12 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	750 460	789 817	466 480	29 177
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 505	8 121	3 130	1 502
Resposta de SRD	3. $t_{A/R}$	10 885	10 885	10 885	0
	4. $t_{ID} + t$	13 253	44 729	10 061	4 522
Ciclo de DAS	1. $t_{S/R}$	10 885	10 885	10 885	0
	2. $t_{SDR} + 2 t_{TD}$	7 525	8 762	5 312	732
Resposta de DAS – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 127	14 687	10 311	737

Soma do ciclo SRD	33 080
Soma do ciclo SDA	34 974
Total (SRD + SDA)	68 054

Tabela A.3. Taxa: 19200 – Estações: 12 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	874 220	909 750	760 146	15 651
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 874	55 160	3 269	3 245
Resposta de SRD	3. $t_{A/R}$	16 614	16 614	16 614	0
	4. $t_{ID} + t$	13 209	44 607	10 143	4 683
Ciclo de DAS	1. t_{SR}	16 614	16 614	16 614	0
	2. $t_{SDR} + 2 t_{TD}$	8 036	9 317	4 294	785
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	11 863	13 153	10 110	738

Soma do ciclo SRD	39 134
Soma do ciclo SDA	39 950
Total (SRD + SDA)	79 083

Tabela A.4. Taxa: 19200 – Estações: 12 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 011 141	1 074 696	415 221	41 676
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 792	8 412	3 358	1 493
Resposta de SRD	3. $t_{A/R}$	22 343	22 343	22 343	0
	4. $t_{ID} + t$	13 193	43 386	10 117	3 969
Ciclo de DAS	1. t_{SR}	22 343	22 343	22 343	0
	2. $t_{SDR} + 2 t_{TD}$	8 502	9 886	4 838	868
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	12 043	15 061	10 047	1 739

Soma do ciclo SRD	44 765
Soma do ciclo SDA	46 324
Total (SRD + SDA)	91 089

Tabela A.5. Taxa: 19200 – Estações: 12 – Data_Unit: 40 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 162 709	1 214 688	1 134 684	16 830
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	6 093	8 509	3 489	1 460
Resposta de SRD	3. $t_{A/R}$	28 072	28 072	28 072	0
	4. $t_{ID} + t$	12 583	15 135	10 129	1 462
Ciclo de DAS	1. t_{SR}	28 072	28 072	28 072	0
	2. $t_{SDR} + 2 t_{TD}$	9 131	10 403	5 410	868
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 215	15 070	10 103	1 588

Soma do ciclo SRD	50 185
Soma do ciclo SDA	53 855
Total (SRD + SDA)	104 040

A.2. RESULTADOS A 19200 BPS – 24 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 19200 – Estações: 24 – Data_Unit: 2 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 343 530	1 419 621	316 470	73 918
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 445	7 980	3 031	1 501
Resposta de SRD	3. $t_{A/R}$	6 302	6 302	6 302	0
	4. $t_{ID} + t$	12 753	19 756	10 186	1 768
Ciclo de SDA	1. $t_{S/R}$	6 302	6 302	6 302	0
	2. $t_{SDR} + 2 t_{TD}$	7 040	8 327	5 852	746
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 214	14 403	11 929	745

Soma do ciclo SRD 27 936

Soma do ciclo SDA 29 993

Total (SRD + SDA) 57 929

Tabela A.2. Taxa: 19200 – Estações: 24 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 584 028	1 674 555	1 359 682	36 467
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 560	8 031	3 133	1 494
Resposta de SRD	3. $t_{A/R}$	10 885	10 885	10 885	0
	4. $t_{ID} + t$	12 960	19 676	10 207	2 059
Ciclo de SDA	1. $t_{S/R}$	10 885	10 885	10 885	0
	2. $t_{SDR} + 2 t_{TD}$	7 563	8 759	6 357	748
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 104	14 307	11 907	748

Soma do ciclo SRD 32 842

Soma do ciclo SDA 34 989

Total (SRD + SDA) 67 831

Tabela A.3. Taxa: 19200 – Estações: 24 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 827 368	1 889 517	1 784 556	24 855
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 593	8 200	3 262	1 422
Resposta de SRD	3. $t_{A/R}$	16 614	16 614	16 614	0
	4. $t_{ID} + t$	14 242	44 385	10 137	7 134
Ciclo de DAS	1. t_{SR}	16 614	16 614	16 614	0
	2. $t_{SDR} + 2 t_{TD}$	8 043	9 291	4 880	862
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	11 893	15 047	10 635	862

Soma do ciclo SRD	39 886
Soma do ciclo SDA	39 986
Total (SRD + SDA)	79 873

Tabela A.4. Taxa: 19200 – Estações: 24 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	2 099 540	2 214 379	111 529	183 766
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 897	8 426	3 469	1 431
Resposta de SRD	3. $t_{A/R}$	22 343	22 343	22 343	0
	4. $t_{ID} + t$	13 483	43 315	10 054	4 625
Ciclo de SDA	1. t_{SR}	22 343	22 343	22 343	0
	2. $t_{SDR} + 2 t_{TD}$	8 564	9 866	4 838	847
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	12 070	15 040	10 046	1 786

Soma do ciclo SRD	45 160
Soma do ciclo SDA	46 414
Total (SRD + SDA)	91 574

Tabela A.5. Taxa: 19200 – Estações: 24 – Data_Unit: 40 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	2 427 102	2 494 368	2 379 376	27 437
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 960	8 545	3 569	1 484
Resposta de SRD	3. $t_{A/R}$	28 072	28 072	28 072	0
	4. $t_{ID} + t$	14 781	43 867	10 284	6 890
Ciclo de SDA	1. t_{SR}	28 072	28 072	28 072	0
	2. $t_{SDR} + 2 t_{TD}$	8 991	10 343	5 383	959
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 036	15 025	10 149	1 666

Soma do ciclo SRD	52 250
Soma do ciclo SDA	53 536
Total (SRD + SDA)	105 786

A.3. RESULTADOS A 19200 BPS – 48 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 19200 – Estações: 48 – Data_Unit: 2 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	2 726 213	2 839 239	3 532	266 888
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 537	8 027	3 102	1 567
Resposta de SRD	3. $t_{A/R}$	6 302	6 302	6 302	0
	4. $t_{ID} + t$	13 697	42 899	10 063	5 834
Ciclo de Das	1. $t_{S/R}$	6 302	6 302	6 302	0
	2. $t_{SDR} + 2 t_{TD}$	7 006	8 316	3 281	822
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 153	14 736	10 114	783

Soma do ciclo SRD 28 973

Soma do ciclo DAS 29 898

Total (SRD + SDA) 58 871

Tabela A.2. Taxa: 19200 – Estações: 48 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	3 196 248	3 314 130	1 853 452	150 522
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 553	8 056	3 257	1 356
Resposta de SRD	3. $t_{A/R}$	10 885	10 885	10 885	0
	4. $t_{ID} + t$	13 467	43 458	10 079	5 379
Ciclo de SDA	1. $t_{S/R}$	10 885	10 885	10 885	0
	2. $t_{SDR} + 2 t_{TD}$	7 497	8 695	3 731	902
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 058	14 770	11 909	717

Soma do ciclo SRD 33 342

Soma do ciclo SDA 34 877

Total (SRD + SDA) 68 219

Tabela A.3. Taxa: 19200 – Estações: 48 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	3 730 496	3 819 013	3 674 025	40 155
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 384	8 211	3 294	1 485
Resposta de SRD	3. $t_{A/R}$	16 614	16 614	16 614	0
	4. $t_{ID} + t$	13 670	43 005	10 112	6 471
Ciclo de SDA	1. t_{SR}	16 614	16 614	16 614	0
	2. $t_{SDR} + 2 t_{TD}$	7 918	9 323	5 045	930
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	12 017	14 878	10 600	930

Soma do ciclo SRD	39 105
Soma do ciclo SDA	39 986
Total (SRD + SDA)	79 091

Tabela A.4. Taxa: 19200 – Estações: 48 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	4 274 221	4 428 796	1 610 626	350 369
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 846	8 202	3 379	1 540
Resposta de SRD	3. $t_{A/R}$	22 343	22 343	22 343	0
	4. $t_{ID} + t$	12 380	17 593	10 213	1 524
Ciclo de SDA	1. t_{SR}	22 343	22 343	22 343	0
	2. $t_{SDR} + 2 t_{TD}$	8 348	9 892	7 384	747
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	12 000	15 025	10 072	1 565

Soma do ciclo SRD	44 006
Soma do ciclo SDA	46 128
Total (SRD + SDA)	90 134

Tabela A.5. Taxa: 19200 – Estações: 48 – Data_Unit: 40 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	4 887 471	5 063 649	1 304 065	503 959
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	6 438	8 418	3 784	1 354
Resposta de SRD	3. $t_{A/R}$	28 072	28 072	28 072	0
	4. $t_{ID} + t$	14 019	42 551	10 118	6 286
Ciclo de SDA	1. t_{SR}	28 072	28 072	28 072	0
	2. $t_{SDR} + 2 t_{TD}$	9 277	10 414	7 949	682
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 446	15 019	10 150	1 376

Soma do ciclo SRD	51 966
Soma do ciclo SDA	54 232
Total (SRD + SDA)	106 199

A.4. RESULTADOS A 19200 BPS – 72 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 19200 – Estações: 72 – Data_Unit: 2 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	4 145 008	4 273 832	3 482 694	95 251
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 574	8 065	3 076	1 465
Resposta de SRD	3. $t_{A/R}$	6 302	6 302	6 302	0
	4. $t_{ID} + t$	12 570	15 975	10 145	1 477
Ciclo de SDA	1. $t_{S/R}$	6 302	6 302	6 302	0
	2. $t_{SDR} + 2 t_{TD}$	7 074	8 296	5 830	732
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 179	14 425	11 956	735

Soma do ciclo SRD	27 883
Soma do ciclo SDA	29 992
Total (SRD + SDA)	57 875

Tabela A.2. Taxa: 19200 – Estações: 72 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	4 790 876	4 973 657	1 165 677	483 494
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 827	7 974	3 277	1 385
Resposta de SRD	3. $t_{A/R}$	10 885	10 885	10 885	0
	4. $t_{ID} + t$	12 805	17 676	10 070	1 721
Ciclo de DAS	1. $t_{S/R}$	10 885	10 885	10 885	0
	2. $t_{SDR} + 2 t_{TD}$	7 539	8 661	6 263	707
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 127	14 449	12 006	707

Soma do ciclo SRD	32 954
Soma do ciclo SDA	34 988
Total (SRD + SDA)	67 942

Tabela A.3. Taxa: 19200 – Estações: 72 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	5 564 790	5 733 532	2 474 463	459 708
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 398	8 157	3 235	1 635
Resposta de SRD	3. $t_{A/R}$	16 614	16 614	16 614	0
	4. $t_{ID} + t$	13 498	38 838	10 120	5 539
Ciclo de DAS	1. t_{SR}	16 614	16 614	16 614	0
	2. $t_{SDR} + 2 t_{TD}$	8 250	9 269	4 314	876
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	11 578	13 093	10 623	666

Soma do ciclo SRD	38 947
Soma do ciclo SDA	39 879
Total (SRD + SDA)	78 826

Tabela A.4. Taxa: 19200 – Estações: 72 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	6 485 426	6 648 270	4 928 950	260 416
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	6 195	8 413	3 367	1 534
Resposta de SRD	3. $t_{A/R}$	22 343	22 343	22 343	0
	4. $t_{ID} + t$	12 387	18 727	10 046	1 801
Ciclo de DAS	1. t_{SR}	22 343	22 343	22 343	0
	2. $t_{SDR} + 2 t_{TD}$	8 548	9 828	7 465	717
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	12 028	15 013	10 117	1 741

Soma do ciclo SRD	44 362
Soma do ciclo SDA	46 356
Total (SRD + SDA)	90 718

Tabela A.5. Taxa: 19200 – Estações: 72 – Data_Unit: 40 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	7 384 612	7 608 027	4 163 574	565 255
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	6 377	8 488	3 554	1 465
Resposta de SRD	3. $t_{A/R}$	28 072	28 072	28 072	0
	4. $t_{ID} + t$	14 251	37 122	10 237	6 042
Ciclo de SDA	1. t_{SR}	28 072	28 072	28 072	0
	2. $t_{SDR} + 2 t_{TD}$	9 141	10 334	8 019	668
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 326	14 971	10 058	1 688

Soma do ciclo SRD	52 137
Soma do ciclo DAS	53 976
Total (SRD + SDA)	106 113

A.5. RESULTADOS A 19200 BPS – 96 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 19200 – Estações: 96 – Data_Unit: 2 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	5 536 783	5 703 483	4 561 009	147 192
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 600	8 074	3 114	1 502
Resposta de SRD	3. $t_{A/R}$	6 302	6 302	6 302	0
	4. $t_{ID} + t$	13 710	36 852	10 184	4 781
Ciclo de SDA	1. $t_{S/R}$	6 302	6 302	6 302	0
	2. $t_{SDR} + 2 t_{TD}$	7 001	8 293	5 114	847
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 160	14 434	10 150	908

Soma do ciclo SRD 29 050

Soma do ciclo SDA 29 900

Total (SRD + SDA) 58 950

Tabela A.2. Taxa: 19200 – Estações: 96 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	6 497 581	6 643 268	6 388 283	76 354
Ciclo de SRD	1. $t_{S/R}$	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 166	7 962	3 133	1 452
Resposta de SRD	3. $t_{A/R}$	10 885	10 885	10 885	0
	4. $t_{ID} + t$	12 636	17 501	10 357	1 546
Ciclo de SDA	1. $t_{S/R}$	10 885	10 885	10 885	0
	2. $t_{SDR} + 2 t_{TD}$	7 405	8 743	6 354	686
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 262	14 313	11 921	689

Soma do ciclo SRD 32 123

Soma do ciclo SDA 34 989

Total (SRD + SDA) 67 112

Tabela A.3. Taxa: 19200 – Estações: 96 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	7 515 214	7 672 994	6 975 427	118 400
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 903	8 252	3 316	1 783
Resposta de SRD	3. $t_{A/R}$	16 614	16 614	16 614	0
	4. $t_{ID} + t$	12 467	18 817	10 532	1 791
Ciclo de SDA	1. t_{SR}	16 614	16 614	16 614	0
	2. $t_{SDR} + 2 t_{TD}$	8 287	9 300	6 993	697
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	11 649	12 931	10 627	699

Soma do ciclo SRD	38 420
Soma do ciclo SDA	39 987
Total (SRD + SDA)	78 408

Tabela A.4. Taxa: 19200 – Estações: 96 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	8 723 092	8 867 677	8 602 709	85 635
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	6 429	8 156	3 477	1 318
Resposta de SRD	3. $t_{A/R}$	22 343	22 343	22 343	0
	4. $t_{ID} + t$	12 606	15 025	10 718	1 244
Ciclo de SDA	1. t_{SR}	22 343	22 343	22 343	0
	2. $t_{SDR} + 2 t_{TD}$	8 385	9 815	7 457	776
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	11 981	14 994	10 298	1 592

Soma do ciclo SRD	44 815
Soma do ciclo SDA	46 146
Total (SRD + SDA)	90 961

Tabela A.5. Taxa: 19200 – Estações: 96 – Data_Unit: 40 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	10 001 971	10 127 338	9 822 600	89 060
Ciclo de SRD	1. t_{SR}	3 437	3 437	3 437	0
	2. $t_{SDR} + 2 t_{TD}$	5 853	8 389	3 490	1 456
Resposta de SRD	3. $t_{A/R}$	28 072	28 072	28 072	0
	4. $t_{ID} + t$	14 937	40 017	10 141	7 399
Ciclo de SDA	1. t_{SR}	28 072	28 072	28 072	0
	2. $t_{SDR} + 2 t_{TD}$	9 069	9 931	6 728	767
Resposta de SDA – OK	3. $t_{A/R}$	3 437	3 437	3 437	0
	4. t_{ID}	13 631	15 019	10 213	1 394

Soma do ciclo SRD	52 299
Soma do ciclo SDA	54 209
Total (SRD + SDA)	106 509

A.6. RESULTADOS A 28800 BPS – 12 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 28800 – Estações: 12– Data_Unit: 5 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	416 639	440 911	125 943	19 129
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 670	3 460	1 956	452
Resposta de SRD	3. $t_{A/R}$	5 347	5 347	5 347	0
	4. $t_{ID} + t$	8 672	23 384	6 885	3 570
Ciclo de DAS	1. $t_{S/R}$	5 347	5 347	5 347	0
	2. $t_{SDR} + 2 t_{TD}$	3 010	3 747	2 246	450
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 348	8 134	6 601	451

Soma do ciclo SRD	18 980
Soma do ciclo DAS	17 996
Total (SRD + SDA)	36 976

Tabela A.2. Taxa: 28800 – Estações: 12– Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	477 888	506 903	123 267	23 315
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 770	3 477	1 980	430
Resposta de SRD	3. $t_{A/R}$	7 256	7 256	7 256	0
	4. $t_{ID} + t$	9 866	24 083	6 083	4 654
Ciclo de SDA	1. $t_{S/R}$	7 256	7 256	7 256	0
	2. $t_{SDR} + 2 t_{TD}$	3 211	3 904	2 436	429
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	8 235	9 020	7 519	432

Soma do ciclo SRD	22 183
Soma do ciclo SDA	20 993
Total (SRD + SDA)	43 176

Tabela A.3. Taxa: 28800 – Estações: 12– Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	553 729	572 892	220 065	24 844
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 720	3 492	2 022	456
Resposta de SRD	3. $t_{A/R}$	11 076	11 076	11 076	0
	4. $t_{ID} + t$	8 983	23 582	7 126	3 815
Ciclo de DAS	1. $t_{S/R}$	11 076	11 076	11 076	0
	2. $t_{SDR} + 2 t_{TD}$	3 553	4 293	2 820	430
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 071	7 829	6 323	429

Soma do ciclo SRD	25 070
Soma do ciclo DAS	23 991
Total (SRD + SDA)	49 061

Tabela A.4. Taxa: 28800 – Estações: 12– Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	643 291	671 835	286 963	27 739
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 819	3 559	2 067	427
Resposta de SRD	3. $t_{A/R}$	14 895	14 895	14 895	0
	4. $t_{ID} + t$	8 392	24 704	6 252	4 287
Ciclo de SDA	1. $t_{S/R}$	14 895	14 895	14 895	0
	2. $t_{SDR} + 2 t_{TD}$	3 966	4 668	3 201	458
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 763	9 009	6 060	1 092

Soma do ciclo SRD	28 397
Soma do ciclo SDA	28 915
Total (SRD + SDA)	57 312

A.7. RESULTADOS A 28800 BPS – 24 ESTAÇÕES ESCRAVAS

A.8.

Tabela A.1. Taxa: 28800 – Estações: 24– Data_Unit: 5 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	858 468	877 134	309 093	43 026
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 691	3 443	1 960	445
Resposta de SRD	3. $t_{A/R}$	5 347	5 347	5 347	0
	4. $t_{ID} + t$	7 669	8 421	6 907	444
Ciclo de DAS	1. $t_{S/R}$	5 347	5 347	5 347	0
	2. $t_{SDR} + 2 t_{TD}$	3 021	3 740	2 247	448
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 339	8 125	6 613	446

Soma do ciclo SRD 17 998

Soma do ciclo DAS 17 998

Total (SRD + SDA) 35 996

Tabela A.2. Taxa: 28800 – Estações: 24– Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	986 363	1 022 742	469 263	43 349
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 803	3 479	2 009	428
Resposta de SRD	3. $t_{A/R}$	7 256	7 256	7 256	0
	4. $t_{ID} + t$	8 111	9 127	6 106	889
Ciclo de SDA	1. $t_{S/R}$	7 256	7 256	7 256	0
	2. $t_{SDR} + 2 t_{TD}$	3 270	3 897	2 439	417
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	8 176	9 007	7 523	417

Soma do ciclo SRD 20 461

Soma do ciclo SDA 20 992

Total (SRD + SDA) 41 454

Tabela A.3. Taxa: 28800 – Estações: 24– Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 137 590	1 163 730	246 454	77 611
Ciclo de SRD	1. t_{SR}	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 739	3 477	2 021	417
Resposta de SRD	3. $t_{A/R}$	11 076	11 076	11 076	0
	4. $t_{ID} + t$	7 887	8 625	7 137	416
Ciclo de SDA	1. t_{SR}	11 076	11 076	11 076	0
	2. $t_{SDR} + 2 t_{TD}$	3 569	4 272	2 817	439
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 060	7 831	6 349	438

Soma do ciclo SRD	23 993
Soma do ciclo SDA	23 996
Total (SRD + SDA)	47 989

Tabela A.4. Taxa: 28800 – Estações: 24– Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1.327.579	1.361.660	391.335	88.296
Ciclo de SRD	1. t_{SR}	2.291	2.291	2.291	0
	2. $t_{SDR} + 2 t_{TD}$	2.787	3.550	2.065	461
Resposta de SRD	3. $t_{A/R}$	14.895	14.895	14.895	0
	4. $t_{ID} + t$	8.210	22.538	6.261	3.929
Ciclo de SDA	1. t_{SR}	14.895	14.895	14.895	0
	2. $t_{SDR} + 2 t_{TD}$	4.005	4.661	3.208	455
Resposta de SDA – OK	3. $t_{A/R}$	2.291	2.291	2.291	0
	4. t_{ID}	7.832	9.082	6.053	1.060

Soma do ciclo SRD	28 182
Soma do ciclo SDA	29 023
Total (SRD + SDA)	57 205

A.9. RESULTADOS A 28800 BPS – 31 ESTAÇÕES ESCRAVAS

Tabela A.1. Taxa: 28800 – Estações: 31 – Data_Unit: 5 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 110 740	1 139 747	128 982	83 265
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 694	3 471	1 954	462
Resposta de SRD	3. $t_{A/R}$	5 347	5 347	5 347	0
	4. $t_{ID} + t$	9 063	23 335	6 917	4 236
Ciclo de DAS	1. $t_{S/R}$	5 347	5 347	5 347	0
	2. $t_{SDR} + 2 t_{TD}$	3 015	3 746	2 264	455
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 338	8 116	6 601	460

Soma do ciclo SRD	19 395
Soma do ciclo DAS	17 991
Total (SRD + SDA)	37 385

Tabela A.2. Taxa: 28800 – Estações: 31 – Data_Unit: 10 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 284 492	1 310 678	1 241 672	19 424
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 659	3 478	1 995	444
Resposta de SRD	3. $t_{A/R}$	7 256	7 256	7 256	0
	4. $t_{ID} + t$	8 650	24 004	6 095	3 263
Ciclo de SDA	1. $t_{S/R}$	7 256	7 256	7 256	0
	2. $t_{SDR} + 2 t_{TD}$	3 198	3 900	2 434	460
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	8 252	9 022	7 523	459

Soma do ciclo SRD	20 856
Soma do ciclo SDA	20 997
Total (SRD + SDA)	41 853

Tabela A.3. Taxa: 28800 – Estações: 31 – Data_Unit: 20 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 475 112	1 499 632	258 541	118 952
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	2 796	3 482	2 037	421
Resposta de SRD	3. $t_{A/R}$	11 076	11 076	11 076	0
	4. $t_{ID} + t$	9 112	57 005	7 143	5 684
Ciclo de SDA	1. $t_{S/R}$	11 076	11 076	11 076	0
	2. $t_{SDR} + 2 t_{TD}$	3 584	4 279	2 818	478
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 032	7 836	6 339	477

Soma do ciclo SRD	25 275
Soma do ciclo SDA	23 983
Total (SRD + SDA)	49 258

Tabela A.4. Taxa: 28800 – Estações: 31 – Data_Unit: 30 – tempo (μ s)

		Média	Máximo	Mínimo	Desvio Padrão
Reação do Sistema	1. t_{SR}	1 735 588	1 766 561	1 685 603	21 337
Ciclo de SRD	1. $t_{S/R}$	2 291	2 291	2 291	0
	2. $t_{SDR} + 2 t_{TD}$	3 608	76 662	2 067	7 670
Resposta de SRD	3. $t_{A/R}$	14 895	14 895	14 895	0
	4. $t_{ID} + t$	7 859	22 341	6 270	3 418
Ciclo de SDA	1. $t_{S/R}$	14 895	14 895	14 895	0
	2. $t_{SDR} + 2 t_{TD}$	3 979	4 659	3 204	416
Resposta de SDA – OK	3. $t_{A/R}$	2 291	2 291	2 291	0
	4. t_{ID}	7 881	9 124	6 067	1 099

Soma do ciclo SRD	28 653
Soma do ciclo SDA	29 046
Total (SRD + SDA)	57 699

B. ANEXO B - ANÁLISES DOS CICLOS DE COMUNICAÇÃO

B.1. CICLO SDA A 19200 BPS

O ciclo SDA é formado pelos tempos descritos a seguir:

1. $t_{S/R}$, tempo gasto pelo quadro com os dados de resposta enviado pelo mestre. Este tempo é independente do número de estações na rede e está definido pelo tamanho da área de dados mais os 9 bytes de controle de um quadro variável:

$$T_{S/R} = a.11 \text{ bits} \quad (1)$$

$$\text{DATA_UNIT} = 2$$

$$t_{S/R} = (9+2).11/19200 \quad t_{S/R} = 6302 \mu\text{s}$$

$$\text{DATA_UNIT} = 10$$

$$t_{S/R} = (9+10).11/19200 \quad t_{S/R} = 10885 \mu\text{s}$$

$$\text{DATA_UNIT} = 20$$

$$t_{S/R} = (9+20).11/19200 \quad t_{S/R} = 16614 \mu\text{s}$$

$$\text{DATA_UNIT} = 30$$

$$t_{S/R} = (9+30).11/19200 \quad t_{S/R} = 22343 \mu\text{s}$$

$$\text{DATA_UNIT} = 40$$

$$t_{S/R} = (9+40).11/19200 \quad t_{S/R} = 28072 \mu\text{s}$$

2. $t_{SDR} + 2 t_{TD}$, representa o tempo transcorrido entre o último byte enviado pelo mestre e o primeiro byte do quadro de confirmação da estação atual.

3. $t_{A/R}$, tempo gasto pelo quadro de confirmação da estação atual para o mestre, de tamanho fixo de 6 bytes, o tempo gasto para a transmissão deste quadro é também fixo independente do número de estações na rede e é calculado por:

$$T_{A/R} = b.11 \text{ bits} \quad (2)$$

$$t_{A/R} = 6.11/19200$$

$$t_{A/R} = 3437 \mu s$$

4. t_{ID} , tempo entre a chegada da resposta da estação na estação mestre e a transmissão do primeiro byte do primeiro quadro por parte do mestre.

Tabela 5. Taxa:19200 - 12 Estações - Ciclo de SDA (μs)

DATA UNIT	2	10	20	30	40
1. $t_{S/R}$	6 302	10 885	16 614	22 343	28 072
2. $t_{SDR} + 2 t_{ID}$	7 008	7 525	8 036	8 502	9 131
3. $t_{A/R}$	3 437	3 437	3 437	3 437	3 437
4. t_{ID}	13 232	13 127	11 863	12 043	13 215

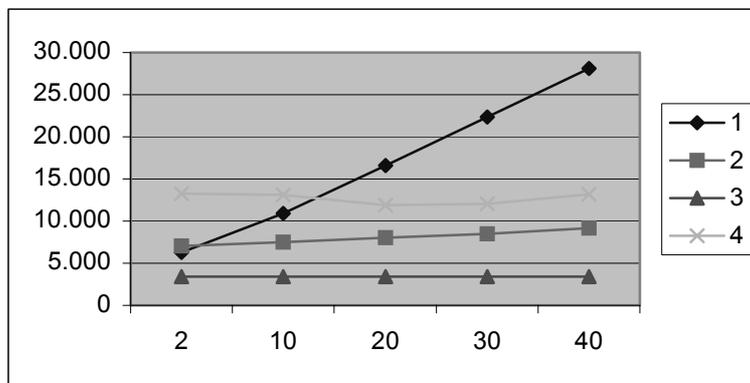


Tabela 6. Taxa:19200 - 24 Estações - Ciclo de SDA (μs)

DATA UNIT	2	10	20	30	40
1. $t_{S/R}$	6 302	10 885	16 614	22 343	28 072
2. $t_{SDR} + 2 t_{ID}$	7 040	7 563	8 043	8 564	8 991
3. $t_{A/R}$	3 437	3 437	3 437	3 437	3 437
4. t_{ID}	13 214	13 104	11 893	12 070	13 036

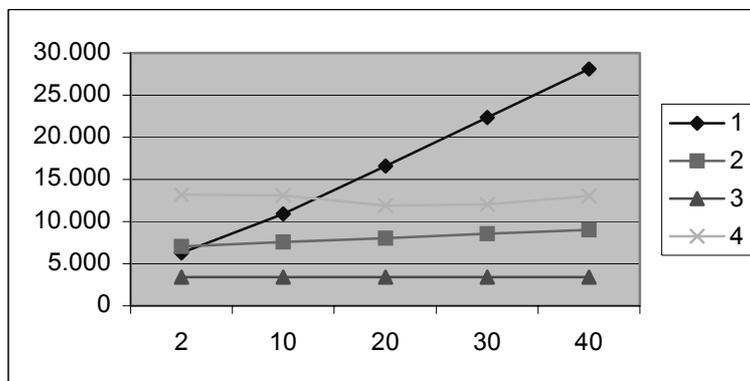


Tabela 7. Taxa:19200 - 48 Estações - Ciclo de SDA (μ s)

DATA UNIT	2	10	20	30	40
1. $t_{S/R}$	6 302	10 885	16 614	22 343	28 072
2. $t_{SDR} + 2 t_{TD}$	7 006	7 497	7 918	8 348	9 277
3. $t_{A/R}$	3 437	3 437	3 437	3 437	3 437
4. t_{ID}	13 153	13 058	12 017	12 000	13 446

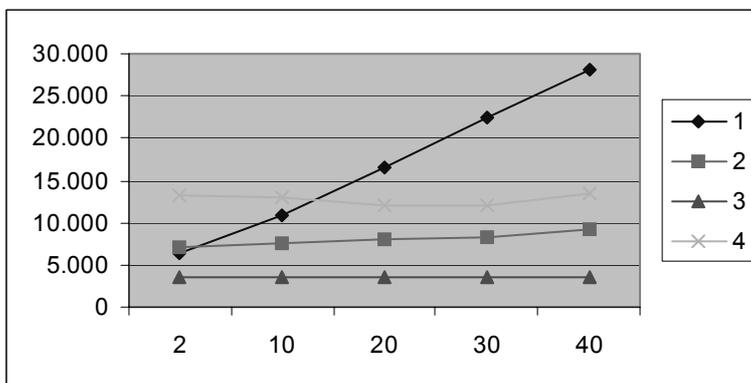


Tabela 8. Taxa:19200 - 72 Estações - Ciclo de SDA (μ s)

DATA UNIT	2	10	20	30	40
1. $t_{S/R}$	6 302	10 885	16 614	22 343	28 072
2. $t_{SDR} + 2 t_{TD}$	7 074	7 539	8 250	8 548	9 141
3. $t_{A/R}$	3 437	3 437	3 437	3 437	3 437
4. t_{ID}	13 179	13 127	11 578	12 028	13 326

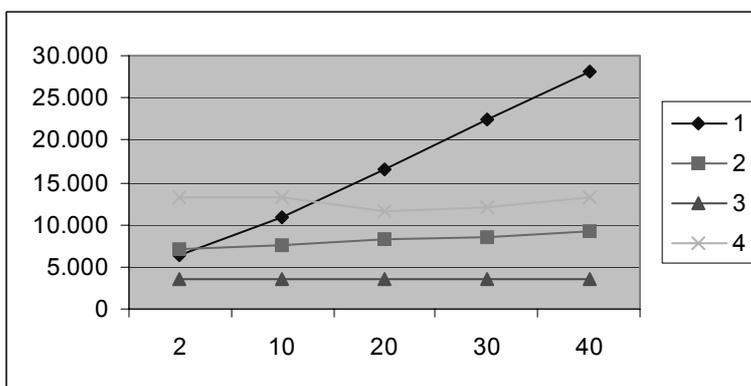
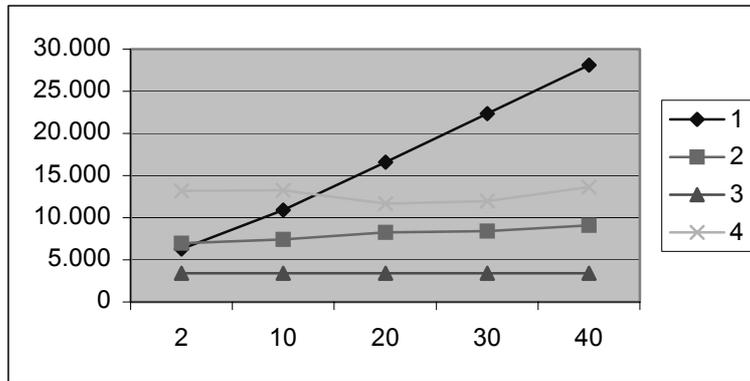


Tabela 9. Taxa:19200 - 96 Estações - Ciclo de SDA (μ s)

DATA_UNIT	2	10	20	30	40
1. $t_{S/R}$	6 302	10 885	16 614	22 343	28 072
2. $t_{SDR} + 2 t_{TD}$	7 001	7 405	8 287	8 385	9 069
3. $t_{A/R}$	3 437	3 437	3 437	3 437	3 437
4. t_{D}	13 160	13 262	11 649	11 981	13 631



B.2. CÁLCULO DOS VALORES TEÓRICOS A 28800 BPS

A seguir são apresentadas as considerações feitas para obter estes valores teóricos:

- Os tempos T medidos em bits, foram convertidos em t (segundos).
- O tempo de reação da estação que transmite uma confirmação ou resposta (escravo), t_{SDR} , varia entre 1,5 ms e 3 ms.
- O tempo de reação da estação que transmite uma requisição ou *token* (mestre), t_{ID} , varia entre 8 ms e 12 ms.
- Para o cálculo teórico serão considerados os piores casos para os tempos de t_{SDR} e t_{ID} .
- As repetições das mensagens e o tempo T_{TD} (Transmission Delay Time) não são considerados.
- O Tempo de Transferência de Token é considerado irrelevante, já que há apenas uma estação mestre no sistema.
- Taxa de transmissão utilizada: 28,8 Kbit/s
- DATA_UNIT = 5, 10, 20 e 30 octetos.
- Número de estações escravas, $n_p = 12, 24$ e 31.
- O ciclo de comunicação é composto pela seguinte seqüência de mensagens: o mestre utiliza o serviço SRD (Send and Request Data with Reply) para ler as informações contidas no buffer do escravo. Em seguida, o mestre decodifica os dados obtidos do buffer do escravo e conforme o que foi solicitado envia as informações utilizando o serviço SDA (Send Data with Acknowledge).
- O ciclo de SRD é composto por um quadro de requisição sem unidade de dados e um quadro de resposta com área de dados variável.
$$a + b = 6 + (9 + DATA_UNIT)$$
- O ciclo de SDA é composto por um quadro de envio de dados com área de dados variável e um ACK como resposta.

$$a + b = (9 + \text{DATA_UNIT}) + 6$$

Pelas considerações acima temos:

$$a + b = 6 + 9 + \text{DATA_UNIT}$$

$$t_{\text{SDR}} = 3 \text{ ms}$$

$$t_{\text{ID}} = 12 \text{ ms}$$

$$T_X = 28800 \text{ bit/s}$$

$$t_{\text{MC}} = [(a + b) 11 + (t_{\text{SDR}} + t_{\text{ID}}) T_X] / T_X$$

$$t_{\text{MC}} = (597 + 11 \cdot \text{DATA_UNIT}) / 28800$$

$$R_{\text{SYS}} = 1 / t_{\text{MC}}$$

Na tabela abaixo são apresentados os resultados teóricos obtidos com os valores utilizados nos testes

Tabela 10. Resultados Teóricos – Tx 28800 bps

DATA_UNIT	t_{MC} (ms)	R_{SYS} Arredondado
5	22,60	44
10	24,50	41
20	28,40	35
30	32,20	31

B.3. CICLO SRD A 28800 BPS

O ciclo SRD é formado pelos tempos descritos a seguir:

1. $t_{\text{S/R}}$, quadro de consulta enviado pelo mestre para uma estação, de tamanho fixo de 6 bytes, o tempo gasto para a transmissão deste quadro é também fixo independente do número de estações na rede e é calculado por:

$$T_{\text{S/R}} = a \cdot 11 \text{ bits} \quad (1)$$

$$t_{\text{S/R}} = 6 \cdot 11 / 28800$$

$$t_{\text{S/R}} = 2291 \text{ } \mu\text{s}$$

2. $t_{\text{SDR}} + 2 T_{\text{TD}}$, representa o tempo transcorrido entre o último byte da requisição e o primeiro da resposta da estação para o mestre.

3. $t_{A/R}$, tempo gasto pelo quadro de resposta da estação com área de dados variável, o tempo gasto para a transmissão deste quadro é independente do número de estações na rede e está definido pelo tamanho da área de dados mais os 9 bytes de controle de um quadro variável:

$$T_{A/R} = b.11 \text{ bits} \quad (2)$$

$$\text{DATA_UNIT} = 5$$

$$t_{A/R} = (9+5).11/28800 \quad t_{A/R} = 5347 \mu\text{s}$$

$$\text{DATA_UNIT} = 10$$

$$t_{A/R} = (9+10).11/28800 \quad t_{A/R} = 7256 \mu\text{s}$$

$$\text{DATA_UNIT} = 20$$

$$t_{A/R} = (9+20).11/28800 \quad t_{A/R} = 11076 \mu\text{s}$$

$$\text{DATA_UNIT} = 30$$

$$t_{A/R} = (9+30).11/28800 \quad t_{A/R} = 14895 \mu\text{s}$$

4. $t_{ID} + t$ Tempo entre a chegada da resposta da estação na estação mestre e o envio do quadro para a próxima estação presente na rede por parte do mestre. É importante destacar que este tempo envolve também os tempos de transmissão dos quadros de token e de pesquisa das estações presentes na rede.

Tabela 11. Taxa:28800 - 12 Estações - Ciclo de SRD (μs)

DATA_UNIT	5	10	20	30
1. $t_{S/R}$	2 291	2 291	2 291	2 291
2. $t_{SDR} + 2 t_{TD}$	2 670	2 770	2 720	2 819
3. $t_{A/R}$	5 347	7 256	11 076	14 895
4. $t_{ID} + t$	8 672	9 866	8 983	8 392

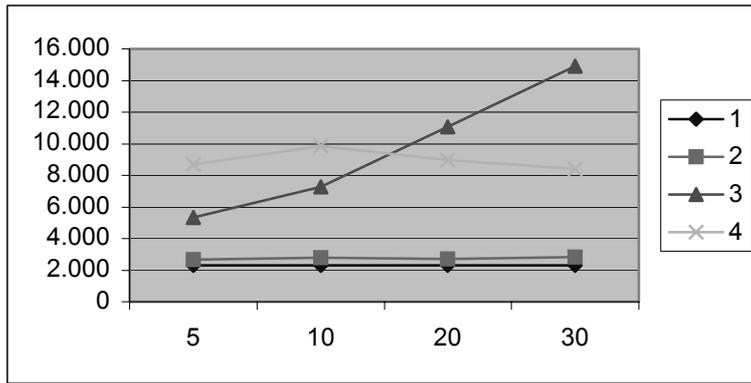


Tabela 12. Taxa:28800 - 24 Estações - Ciclo de SRD (μ s)

DATA UNIT	5	10	20	30
1. $t_{S/R}$	2 291	2 291	2 291	2 291
2. $t_{SDR} + 2 t_{TD}$	2 691	2 803	2 739	2 787
3. $t_{A/R}$	5 347	7 256	11 076	14 895
4. $t_{ID} + t$	7 669	8 111	7 887	8 210

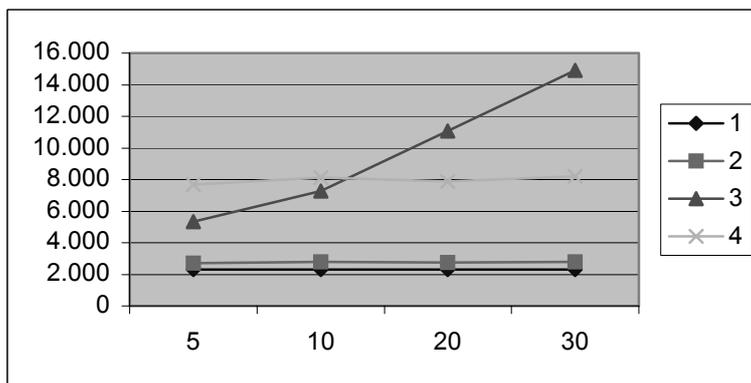
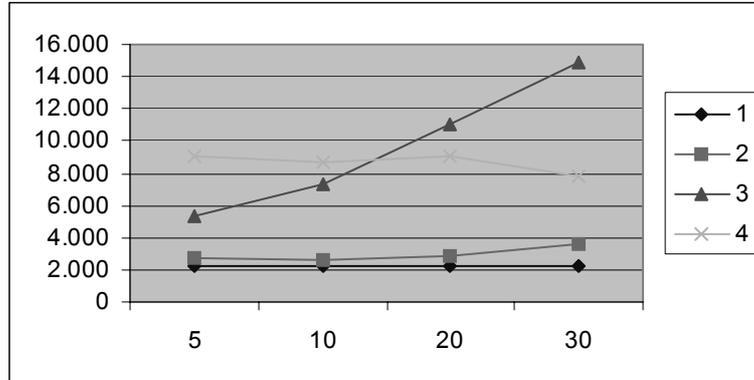


Tabela 13. Taxa:28800 – 31 Estações – Ciclo de SRD (μ s)

DATA_UNIT	5	10	20	30
1. $t_{S/R}$	2 291	2 291	2 291	2 291
2. $t_{SDR} + 2 t_{TD}$	2 694	2 659	2 796	3 608
3. $t_{A/R}$	5 347	7 256	11 076	14 895
4. $t_{ID} + t$	9 063	8 650	9 112	7 859



B.4. CICLO SDA A 28800 BPS

O ciclo SDA é formado pelos tempos descritos a seguir:

1. $T_{S/R}$, tempo gasto pelo quadro com os dados de resposta enviado pelo mestre. Este tempo é independente do número de estações na rede e está definido pelo tamanho da área de dados mais os 9 bytes de controle de um quadro variável:

$$T_{S/R} = a.11 \text{ bits} \quad (1)$$

$$\text{DATA_UNIT} = 5$$

$$t_{S/R} = (9+5).11/28800 \quad t_{S/R} = 6302 \mu\text{s}$$

$$\text{DATA_UNIT} = 10$$

$$t_{S/R} = (9+10).11/28800 \quad t_{S/R} = 10885 \mu\text{s}$$

$$\text{DATA_UNIT} = 20$$

$$t_{S/R} = (9+20).11/28800 \quad t_{S/R} = 16614 \mu\text{s}$$

$$\text{DATA_UNIT} = 30$$

$$t_{S/R} = (9+30).11/28800 \quad t_{S/R} = 22343 \mu\text{s}$$

2. $T_{SDR} + 2 T_{TD}$, representa o tempo transcorrido entre o último byte enviado pelo mestre e o primeiro byte do quadro de confirmação da estação atual.

3. $T_{A/R}$, tempo gasto pelo quadro de confirmação da estação atual para o mestre, de tamanho fixo de 6 bytes, o tempo gasto para a transmissão deste quadro é também fixo independente do número de estações na rede e é calculado por:

$$T_{A/R} = b.11 \text{ bits} \quad (2)$$

$$t_{A/R} = 6.11/28800$$

$$t_{A/R} = 2291 \mu\text{s}$$

4. t_{ID} Tempo entre a chegada da resposta da estação na estação mestre e a transmissão do primeiro byte do primeiro quadro por parte do mestre.

Tabela 14. Taxa:28800 - 12 Estações - Ciclo de SDA (μ s)

DATA UNIT	5	10	20	30
1. $t_{S/R}$	5 347	7 256	11 076	14 895
2. $t_{SDR} + 2 t_{TD}$	3 010	3 211	3 553	3 966
3. $t_{A/R}$	2 291	2 291	2 291	2 291
4. t_{ID}	7 348	8 235	7 071	7 763

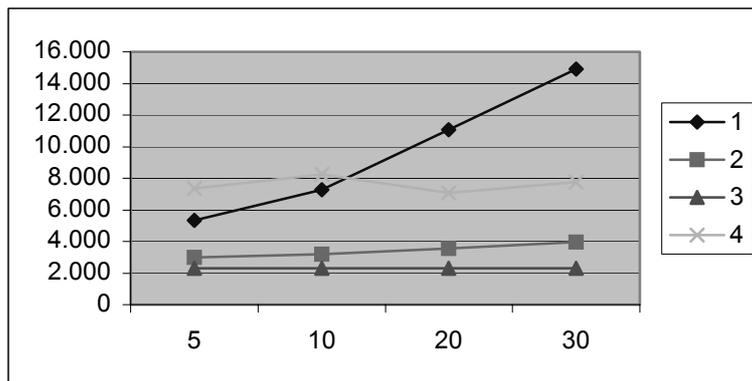


Tabela 15. Taxa:28800 - 24 Estações - Ciclo de SDA (μ s)

DATA UNIT	5	10	20	30
1. $t_{S/R}$	5 347	7 256	11 076	14 895
2. $t_{SDR} + 2 t_{TD}$	3 021	3 270	3 569	4 005
3. $t_{A/R}$	2 291	2 291	2 291	2 291
4. t_{ID}	7 339	8 176	7 060	7 832

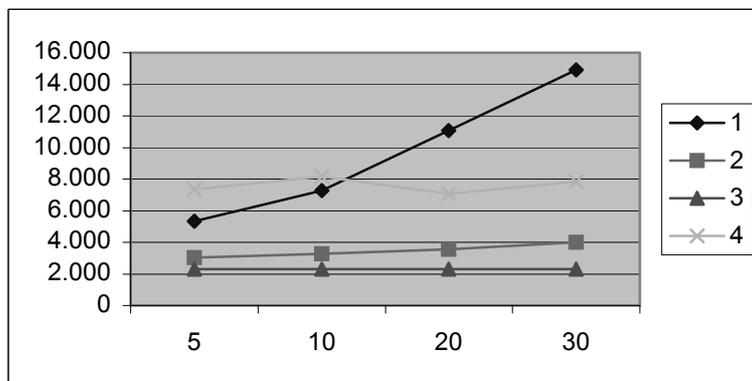
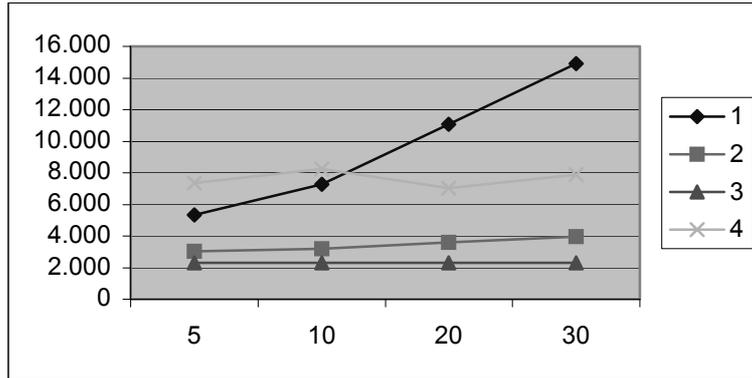


Tabela 16. Taxa:28800 - 31 Estações - Ciclo de SDA (μ s)

DATA UNIT	5	10	20	30
1. $t_{S/R}$	5 347	7 256	11 076	14 895
2. $t_{SDR} + 2 t_{TD}$	3 015	3 198	3 584	3 979
3. $t_{A/R}$	2 291	2 291	2 291	2 291
4. t_{ID}	7 338	8 252	7 032	7 881



C. ANEXO C – LISTAGEM DE UM ARQUIVO DE TESTE

C.1. INTRODUÇÃO

Neste anexo é apresentada uma listagem dos dados obtidos num dos testes realizados com as estações de voto eletrônico.

A tabela está dividida em,

- **Primeira coluna: Tempo Absoluto (μ s)** : tempo medido a partir do começo do procedimento de medida.
- **Segunda coluna: Tempo Relativo (μ s)** : tempo transcorrido desde o começo do último quadro.
- **Terceira coluna: Distância (μ s)** : tempo transcorrido desde o final do quadro anterior.
- **Quarta coluna:** quadros que circulam no barramento.

Fórmula para cálculo do tempo:

$$\text{end_to_start} = \text{start_to_start} - \text{tam_quadro_anterior} * (11 * 10^6 / 28800)$$

Taxa: 28800 – Estações: 12– Data_Unit: 5 – tempo (μ s)

Absolute Time	Start-to-Start	End-to-Start	Quadros
2977	2977	2977	10 0C 01 7C 89 16
7246	4269	1978	68 08 08 68 01 0C 08 01 00 00 00 00 16 16
20964	13718	8371	68 08 08 68 0C 01 53 01 00 00 00 00 61 16
28873	7909	2562	10 01 0C 00 0D 16
38950	10077	7786	10 0D 01 7C 8A 16
44356	5406	3115	68 08 08 68 01 0D 08 01 00 00 00 00 17 16
56940	12584	7237	68 08 08 68 0D 01 53 01 00 00 00 00 62 16
65980	9040	3693	10 01 0D 00 0E 16
74958	8978	6687	10 02 01 7C 7F 16
79930	4972	2681	68 08 08 68 01 02 08 01 00 00 00 00 0C 16
92947	13017	7670	10 0C 01 49 56 16
98329	5382	3091	10 01 0C 00 0D 16
104937	6608	4317	DC 01 01
107952	3015	1870	68 08 08 68 02 01 53 00 00 00 00 00 56 16
117037	9085	3738	10 01 02 00 03 16
125943	8906	6615	10 03 01 7C 80 16
131059	5116	2825	68 08 08 68 01 03 08 00 00 00 00 00 0C 16
143929	12870	7523	68 08 08 68 03 01 53 00 00 00 00 00 57 16
152757	8828	3481	10 01 03 00 04 16
161914	9157	6866	10 04 01 7C 81 16
166654	4740	2449	68 08 08 68 01 04 08 00 00 00 00 00 0D 16
179902	13248	7901	68 08 08 68 04 01 53 00 00 00 00 00 58 16
188313	8411	3064	10 01 04 00 05 16
197923	9610	7319	10 05 01 7C 82 16
202204	4281	1990	68 08 08 68 01 05 08 00 00 00 00 00 0E 16
215913	13709	8362	68 08 08 68 05 01 53 00 00 00 00 00 59 16
223828	7915	2568	10 01 05 00 06 16
233899	10071	7780	10 06 01 7C 83 16
239434	5535	3244	68 08 08 68 01 06 08 00 00 00 00 00 0F 16
251887	12453	7106	68 08 08 68 06 01 53 00 00 00 00 00 5A 16
259566	7679	2332	10 01 06 00 07 16
269908	10342	8051	10 07 01 7C 84 16
274596	4688	2397	68 08 08 68 01 07 08 00 00 00 00 00 10 16
287893	13297	7950	68 08 08 68 07 01 53 00 00 00 00 00 5B 16
296223	8330	2983	10 01 07 00 08 16

305884	9661	7370	10 0D 01 49 57 16
310384	4500	2209	10 01 0D 00 0E 16
317874	7490	5199	DC 01 01
320889	3015	1870	10 08 01 7C 85 16
325624	4735	2444	68 08 08 68 01 08 08 00 00 00 00 11 16
338877	13253	7906	68 08 08 68 08 01 53 00 00 00 00 5C 16
347284	8407	3060	10 01 08 00 09 16
356865	9581	7290	10 09 01 7C 86 16
361491	4626	2335	68 08 08 68 01 09 08 00 00 00 00 12 16
374889	13398	8051	68 08 08 68 09 01 53 00 00 00 00 5D 16
383116	8227	2880	10 01 09 00 0A 16
392874	9758	7467	10 0A 01 7C 87 16
398316	5442	3151	68 08 08 68 01 0A 08 00 00 00 00 13 16
410863	12547	7200	68 08 08 68 0A 01 53 00 00 00 00 5E 16
419941	9078	3731	10 01 0A 00 0B 16
428848	8907	6616	10 0B 01 7C 88 16
434307	5459	3168	68 08 08 68 01 0B 08 00 00 00 00 14 16
446872	12565	7218	68 08 08 68 0B 01 53 00 00 00 00 5F 16
455934	9062	3715	10 01 0B 00 0C 16
464856	8922	6631	10 0C 01 7C 89 16
469821	4965	2674	68 08 08 68 01 0C 08 00 00 00 00 15 16
482844	13023	7676	68 08 08 68 0C 01 53 00 00 00 00 60 16
491482	8638	3291	10 01 0C 00 0D 16
500832	9350	7059	10 0D 01 7C 8A 16
505398	4566	2275	68 08 08 68 01 0D 08 00 00 00 00 16 16
518818	13420	8073	DC 01 01
521836	3018	1873	68 08 08 68 0D 01 53 00 00 00 00 61 16
530142	8306	2959	10 01 0D 00 0E 16
539823	9681	7390	10 02 01 7C 7F 16
544125	4302	2011	68 08 08 68 01 02 08 00 00 00 00 0B 16
557811	13686	8339	68 08 08 68 02 01 53 01 00 00 00 57 16
565750	7939	2592	10 01 02 00 03 16
575830	10080	7789	10 03 01 7C 80 16
581241	5411	3120	68 08 08 68 01 03 08 01 00 00 00 0D 16
593817	12576	7229	68 08 08 68 03 01 53 01 00 00 00 58 16
602868	9051	3704	10 01 03 00 04 16
611803	8935	6644	10 04 01 7C 81 16
616843	5040	2749	68 08 08 68 01 04 08 01 00 00 00 0E 16
629791	12948	7601	68 08 08 68 04 01 53 01 00 00 00 59 16
638536	8745	3398	10 01 04 00 05 16
647814	9278	6987	10 05 01 7C 82 16
652489	4675	2384	68 08 08 68 01 05 08 01 00 00 00 0F 16
665802	13313	7966	68 08 08 68 05 01 53 01 00 00 00 5A 16
674115	8313	2966	10 01 05 00 06 16
683790	9675	7384	10 06 01 7C 83 16
688362	4572	2281	68 08 08 68 01 06 08 01 00 00 00 10 16
701778	13416	8069	68 08 08 68 06 01 53 01 00 00 00 5B 16
709984	8206	2859	10 01 06 00 07 16
719797	9813	7522	DC 01 01
722776	2979	1834	10 07 01 7C 84 16
727899	5123	2832	68 08 08 68 01 07 08 01 00 00 00 11 16
740766	12867	7520	68 08 08 68 07 01 53 01 00 00 00 5C 16
749598	8832	3485	10 01 07 00 08 16
758788	9190	6899	10 08 01 7C 85 16
763486	4698	2407	68 08 08 68 01 08 08 01 00 00 00 12 16
776773	13287	7940	68 08 08 68 08 01 53 01 00 00 00 5D 16
785112	8339	2992	10 01 08 00 09 16
794763	9651	7360	10 09 01 7C 86 16
799350	4587	2296	68 08 08 68 01 09 08 01 00 00 00 13 16
812746	13396	8049	68 08 08 68 09 01 53 01 00 00 00 5E 16
820975	8229	2882	10 01 09 00 0A 16
830769	9794	7503	10 0A 01 7C 87 16
836179	5410	3119	68 08 08 68 01 0A 08 01 00 00 00 14 16
848758	12579	7232	68 08 08 68 0A 01 53 01 00 00 00 5F 16
857802	9044	3697	10 01 0A 00 0B 16
866745	8943	6652	10 0B 01 7C 88 16

D. ANEXO D - BIBLIOTECA DAS FUNÇÕES EM C

Neste anexo são apresentadas as funções que compõem a biblioteca utilizada nos programas aplicativos dos testes. Esta biblioteca chama os serviços da camada de enlace do Protocolo Profibus conforme descrito no Capítulo 7.

```
#include "profibus.h"
```

```
/******
```

inicio

```
*****
```

OBJETIVO: Conectar o usuário com a rede.

DESCRIÇÃO: Esta função chama a função `profi_start` que é responsável pela inicialização do sistema que controla a conexão à rede através de chamadas a funções específicas, todos os detalhes de inicialização de parâmetros, inicialização de rotinas da serial e de interrupção do timer são tratados.

ENTRADAS : Nenhuma.

SAIDAS : Esta função retorna "YES" em caso de sucesso na conexão ou "NO" em caso contrário.

```
*****/
```

```
int inicio(void)
```

```
{  
    return(profi_start());  
}
```

```
/******
```

liste_status

```
*****
```

OBJETIVO : Descobrir quais as estações escravas estão ligadas à rede

DESCRIÇÃO : Esta função chama às funções `FMA12_LIVE_LIST.request()` e `confirm()`, passando para o usuário um vetor com as estações ligadas na rede e o vetor `LIVE_LIST` com o status das estações.

ENTRADAS :

Live_list - Ponteiro do vetor que contém o estado das estações ligadas à rede conforme descrito na função `FMA12_LIVE_LIST.confirm()`.

SAIDAS:

M_status - Descrito na função `FMA12_LIVE_LIST.confirm()` no `PROFIBUS.H`.

```
*****/
```

```
int liste_status(char Live_list[])
```

```
{  
    int M_status;
```

```
FMA12_LIVE_LIST.request();
```

```
FMA12_LIVE_LIST.confirm(Live_list, &M_status);
```

```
    return(M_status);  
}
```

```

/*****
leia_estacao
*****/
OBJETIVO : Consultar à estação remota para saber se está sendo solicitado serviço.
DESCRICAÇÃO : Esta função chama às funções FDL_DATA_REPLY.request( ) e
                FDL_DATA_REPLY.confirm( ) para consultar o buffer da estação remota onde está
                codificado o serviço que está sendo solicitado.
ENTRADA :
                Rem_Add - Endereço físico da estação remota que está sendo consultada.
                L_sdu - Ponteiro para a área de memória dos dados lidos da estação remota.
SAIDA :
                L_status - Descrito na função FDL_DATA_REPLY.confirm( ) no PROFIBUS.H
*****/
int leia_estacao(unsigned char Rem_Add,char L_sdu[])
{
    SSAP, DSAP, Serv_class;

    FDL_DATA_REPLY.request(0,0,Rem_Add,L_sdu,0);

    FDL_DATA_REPLY.confirm(&SSAP, &DSAP, &Rem_Add, L_sdu, &Serv_class,

                                &L_status);
    }while( !L_status

);

    return(L_status);
}

```

```

/*****
responda_estacao
*****/
OBJETIVO : Enviar a resposta à solicitação recebida na função leia_estacao().
DESCRICAÇÃO : Esta função envia os dados solicitados pela estação remota através da chamada das
                funções FDL_DATA_ACK.request( ) e FDL_DATA_ACK.confirm( ).
ENTRADA :
                Rem_Add - Endereço da estação remota.
                L_sdu - Ponteiro da área de memória de dados enviados para a estação remota.
SAIDA :
                L_status - Escrito na função FDL_DATA_REPLY.confirm( ) no PROFIBUS.H
*****/
int responda_estacao(unsigned char Rem_Add,char L_sdu[])
{
    unsigned char SSAP, DSAP, Serv_class;
    int L_status;

    FDL_DATA_ACK.request(0,0,Rem_Add,L_sdu,0);
    do{

        FDL_DATA_ACK.confirm(&SSAP, &DSAP, &Rem_Add, &Serv_class, &L_status);
    }while( !L_status);
}

```

```
return(L_status);  
}
```

```

/*****
                                requisicao
*****/
OBJETIVO : Solicitar pedidos ao mestre.
DESCRICAÇÃO : Esta função atualiza o buffer do escravo com um código que corresponde ao pedido
                solicitado.
ENTRADAS :
                L_sdu - Ponteiro da área de memória atualizado pelo escravo.
                buffer_Rx - Ponteiro da área de memória de dados enviados pelo mestre.
SAIDAS : Esta função retorna "YES" se recebeu resposta do mestre ou "NO" em caso contrário.
/*****/
int requisicao(char L_sdu[], char buffer_Rx[])
{
    unsigned char SSAP, DSAP, Rem_Add, Loc_Add;
    char Serv_class, cont=0;
    unsigned int i;
    CONTEXT sys;

    FDL_DATA_ACK.indication(&SSAP, &DSAP, &Loc_Add, &Rem_Add,
                            &Serv_class);
    de rececao */
    FDL_REPLY_UPDATE.request(0, L_sdu, 0, 0);

    while(!FDL_DATA_ACK.indication(&SSAP, &DSAP, &Loc_Add, &Rem_Add,
                                    &Serv_class))
    {
        delay(10);
        if ((cont++)==10)
        {
            L_sdu[1]='0';
            FDL_REPLY_UPD
            ATE.request(0, L_sdu, 0, 0);

            return(0);
        }
    }

    L_sdu[1]='0';
    FDL_REPLY_UPDATE.request(0, L_sdu, 0, 0);
    return(1);
}

/*****/
                                fim
*****/
OBJETIVO : Desconectar o usuário da rede.

```

DESCRICAÇÃO : Esta função chama a função PROFI_QUIT que desabilita o transmissor e as rotinas responsáveis pela comunicação serial e pela interrupção do TIMER.

ENTRADAS : Nenhuma.

SAIDAS : Nenhuma.

*****/

void fim(void)

```
{  
    profi_quit();  
}
```

E. ANEXO E – TESTES DE UTILIZAÇÃO DE SCANNERS NOS CAIXAS DE SUPERMERCADO

Este anexo relata uma experiência feita com o objetivo de determinar os benefícios resultantes da instalação de scanners de códigos de barras em filas de caixas registradoras em condições normais, feita por um grupo doméstico da área de supermercados que decidiu conferir mais uma vez a conveniência de possuir scanners nos seus pontos de venda. Devemos esta experiência aos engenheiros D. Passeron e D. Barmat, sob a direção do Eng. F. Braun [ERD94].

E.1. BENEFÍCIOS DIRETOS DOS SCANNERS NO PONTO DE VENDA

Neste estudo de caso, 80% dos itens são codificados na origem. Esta porcentagem corresponde a praticamente tudo que pode ser codificado na origem. As porcentagens consideradas são ligeiramente inferiores àquelas que prevalecem nos Estados Unidos e na Grã-Bretanha. Os parâmetros analisados são:

- Loja com 11 pontos de venda (caixas registradoras)
 - Compra média: 11 itens por cliente
 - clientes atendidos por mês
 - Itens codificados na origem: 80%
 - Leitura bem-sucedida pelo scanner no item: 1ra. vez: 85%; 2da. vez: 10%
 - Análise: economia de mão-de-obra

E.1.1. Tempo na fila do caixa

O tempo total diário de operação de uma caixa registradora consiste nos tempos produtivos e não-produtivos do operador de caixa. O tempo produtivo é dividido em **tempo de registro** que corresponde ao tempo decorrido entre o pressionamento da primeira tecla e o pressionamento da tecla de total, e **tempo de recebimento**, que vai do fim do registro até o fechamento da gaveta de dinheiro. Os **tempos intermediários**, correspondentes às pausas entre os clientes que se sucedem na fila, estão incluídos no tempo de recebimento.

E.1.2. Tempo de atenção ao cliente

Chamamos o **tempo de registro/item** o resultado da divisão do **tempo de registro** pelo número de itens inseridos na operação. Nos exemplos que se seguem, são usados parâmetros de operação que são significativamente inferiores aos limites operacionais teóricos estabelecidos para a função de operador de caixa. A economia de tempo apresentada correspondeu somente ao tempo produtivo. Portanto, considerando que o trabalho está abaixo do limite teórico, há um aumento no tempo de descanso para cada operador de caixa, além da economia de mão-de-obra.

E.2. BENEFÍCIOS DIRETOS, PRODUTIVIDADE NA FILA DO CAIXA

Caso 1: Economia de salário em uma fila do caixa, para um determinado número de clientes.

Caso 1	Teclado	Scanner	Vantagem do Scanner
Tempo:			
Tempo de registro/item (s)	4,2	2,2	- 47 %
Tempo de recebimento/cliente (s)	29,0	29,0	
Clientes/caixa/hora	20,0		
Limite teórico	38,3		
Resultados Operacionais:			
Tempo de registro/cliente (s)	46,2	24,2	
Tempo de recebimento (s)	29,0	29,0	
Tempo total/cliente	94,0	66,5	- 29 %
Horas de trabalho	4,2	3,3	

O tempo de registro por teclado (4,2 segundos) foi tirado da fitas de caixa registradora de várias lojas durante um período de cinco meses (Código de 5 dígitos). O tempo de registro por scanner (2,2 segundos), foi tirado de dados provenientes dos Estados Unidos e conferido localmente em uma fila-piloto de caixa registradora, com 85% de leituras de scanner na primeira passagem e mais 10% na segunda passagem.