FÁBIO DA SILVA CÔRTES

ESTUDO DO PROTOCOLO SNMP EM DISPOSITIVOS FOUNDATION FIELDBUS

Monografia apresentada à Faculdade de Tecnologia SENAI – Cimatec, como parte dos requisitos para obtenção do Título de Especialista em Automação, Controle e Robótica.

Orientador: Prof. Msc. Sérgio F. Brito

SALVADOR

2008

SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL – SENAI
CIMATEC – CENTRO INTEGRADO DE MANUFATURA E TECNOLOGIA
DEPARTAMENTO DE AUTOMAÇÃO INDUSTRIAL

Fábio da Silva Côrtes

Monografia apresentada em 08 Maio de 2008

Prof. MSc. Sérgio Figueiredo Brito
Mestre em Engenharia Elétrica
Orientador

Prof. Dr. Edmarcio Antonio Belati
Componente da Banca Examinadora

Prof. MSc. Milton Bastos de Souza
Componente da Banca Examinadora

Prof. MSc. Sérgio Figueiredo Brito
Componente da Banca Examinadora

Salvador, Bahia, Brasil.
Maio de 2008

AGRADECIMENTOS

Agradeço àqueles que se mostraram especiais a mim no decorrer de minha vida; a começar pela família, aos amigos mais próximos, aos colegas e professores que deixaram em mim algum tipo de aprendizado.

Agradeço especialmente a minha mãe e minha noiva que são as pessoas que mais amo na vida e que mais prezam por mim.

Agradeço ao Professor Sérgio Brito que em nossas reuniões além de orientar teve o papel de motivador para que este trabalho fosse concluído.

Agradeço ainda a Deus, que me dá forças e disposição para seguir sempre adiante.

RESUMO

Este trabalho visa realizar um estudo acerca da existência e funcionalidade do protocolo SNMP (*Simple Network Management Protocol*) nos dispositivos Foundation Fieldbus (FF). Para isto será feita explanação da tecnologia proposta pela *Fieldbus Foundation* destacando as principais características e principais elementos que compõem este sistema de controle e informação.

O Sistema FF apresenta duas tecnologias complementares e totalmente integradas denominadas de H1 e HSE. A tecnologia H1 destina-se a interconexão de dispositivos de campo com taxas de 31,25kbps, tais como sensores, atuadores e válvulas de controle. A tecnologia HSE destina-se a interconexão de dispositivos com taxa de 100Mbps como PLCs, estações de trabalho e interconexão de redes H1, Profibus, DeviceNet e outras.

No desenvolvimento deste trabalho serão levantadas diversas informações sobre o funcionamento do SNMP nos dispositivos FF, incluindo a existência das MIBs (*Management Information Base*) nos dispositivos.

PALAVRAS-CHAVES: Fieldbus Foundation, SNMP, HSE, H1, Foundation Fieldbus, integração industrial, gerenciamento, MIB.

ABSTRACT

This work seeks to accomplish a study concerning the existence and functionality of the protocol SNMP (Simple Network Management Protocol) in the devices Foundation Fieldbus (FF). For it will be made explanation of the technology proposed by Fieldbus Foundation the main characteristics and main elements that compose this control system and information highlighting.

The system FF presents two complementaries technologies and totally integrated denominated of H1 and HSE. The technology H1 the interconnection of field devices is destined with baud rate of 31,25kbps, such as sensor, actuators and control valves. The technology HSE the interconnection of devices is destined with baud rate of 100Mbps as PLCs, work stations and interconnection of nets H1, Profibus, DeviceNet and other.

In the development of this work they will be several lifted up information on the operation of SNMP in the devices FF, including the existence of MIBs (Management Information Base) in the devices.

KEY-WORDS: Fieldbus Foundation, SNMP, HSE, H1, Foundation Fieldbus, management, MIB.

LISTA DE ACRÔNIMOS

Sigla	Termo
AC	*Alternating Current*
AI	*Analog Input Function Block*
APL	*Application layer*
AO	*Analog Output Function Block*
ASN.1	*Abstract Syntax Notation 1*
AUTO	*Automatic mode*
AWG	*American Wire Gauge*
BG	*Bias/Gain Station Block*
CCITT	*International Telegraph and Telephone Consultative Committee*
CAS	*Cascade mode*
CD	*Compel Data DLPDU*
CF	*Capabilities File*
CFF	*Common File Format*
CL	*Claim LAS DLPDU*
COTS	*Commercial off the Shelf*
CPU	*Central Processing Unit*
CS	*Control Selector Function Block*
CT	*Compel Time DLPDU*
DC	*Direct Current*
DC	*Disconnect Connection DLPDU*
DC	*Device Control Block*
DCS	*Distributed Control System*
DD	*Device Description*
DDL	*Device Description Language*
DDS	*Device Description Service*
Device ID	*Device Identifier*
DI	*Discrete Input Function Block*
DLCEP	*Data Link Connection End Point*
DLL	*Data Link Layer*
DLSAP	*Data Link Service Access Point*
DT	*Data Transfer DLPDU*
DT	*Dead Time Function Block*
DO	*Discrete Output Function Block*
EC	*Establish Connection DLPDU*
EU	*Engineering Unit*
EUC	*End User Council*
ETS	*Enterprise Technology Solutions*
FAS	*Fieldbus Access Sublayer*
FB	*Fieldbus*
FB	*Function Block*
FCS	*Frame Check Sequence*
FDA	*Field Device Access*
FF	*Fieldbus Foundation*
FMS	*Fieldbus Message Specification*
Gbps (Gbit/s)	*Gigabit per second*
HMI	*Human Machine Interface*
ID	*Identifier*
IEC	*International Electro-technical Commission*

IMAN	*Initialize Manual mode*
IP	*Internet Protocol*
IS	*Intrinsic Safety*
ISA	*International Society of Measurement and Control*
ISO	*International Organization of Standard*
IT	*Information Technology*
IT	*Integrator Function Block*
Kbps (kbit/s)	*kilobits per second*
kHz	*kilohertz*
LAN	*Local Area Network*
LAS	*Link Active Scheduler*
LL	*Lead Lag Function Block*
LM	*Link Master*
MIB	*Management Information Base*
MES	*Manufacturing Execution System*
ERP	*Enterprise Resource Planning*

SUMÁRIO

1 – INTRODUÇÃO

Novos sistemas de controle automático surgiram ao longo dos últimos anos trazendo ao chão de fábrica diversos dispositivos e equipamentos capazes de auto regular processos, gerenciar informações e responder de maneira rápida, precisa e confiável às mudanças exigidas durante a produção [GOMES, 2005].

Até o início dos anos 90, existiam centenas de fabricantes de *hardware* e *software* de automação vendendo sistemas fechados e proprietários. Assim, a escolha por um fabricante ou marca era quase um caminho sem volta, pois, os custos da mudança eram muito altos e a integração com outros sistemas inviáveis.

Diversos fabricantes de equipamentos e sistemas, centros de pesquisa aplicada, universidades e outros institutos de tecnologia focaram suas ações na integração de sistemas industriais. O padrão a ser desenvolvido deveria integrar os diferentes tipos de instrumentos de controle, proporcionando uma interface para a operação de diversos dispositivos simultaneamente e um conjunto de protocolos de comunicação para todos eles.

Devido à diversidade de produtos e métodos de implementação, o processo de padronização se tornou lento, não permitindo uma solução direta e simples para ser padronizada. Por isso em Setembro de 1994 duas organizações, WorldFIP e ISP, juntaram-se formando a *Fieldbus Foundation* com o objetivo de acelerar o processo de normalização das redes industriais de automação.

O Sistema *Foundation Fieldbus* [FIELDBUS FOUNDATION, 2003] é um sistema de comunicação completamente digital, serial e bidirecional que interconecta dispositivos de campo (sensores, atuadores, válvulas e controladores) com estações de trabalho, servidores de arquivos e outros, promovendo a integração total da informação na indústria (Figura 1).

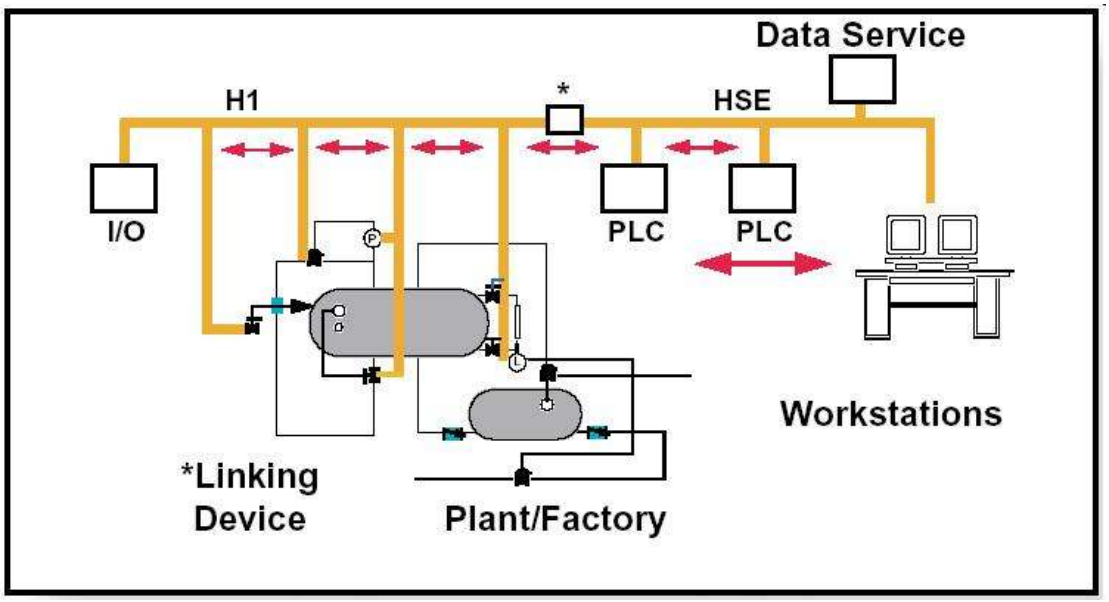

Figura 1 – Sistema integrado de comunicação Fieldbus Foundation.
Fonte: Artigo "Fieldbus, Ethernet and Reality Convergence" Jonas Berge - SMAR.

O *Foundation Fieldbus* (FF) é definido como uma rede de campo digital, bidirecional para comunicação multiponto entre dispositivos inteligentes de controle. É uma de várias redes locais dedicadas para automação industrial, tais como Profibus [SMAR PROFIBUS, 2003], DeviceNet [Allen-Bradley, 1996] e outras redes de medição e controle[OLIVEIRA, 2005].

O FF é uma rede local de instrumentação [MONTEZ, 2005] e controle [OGATA, 2003] com habilidade de distribuir o controle da planta entre os dispositivos de campo através de uma comunicação digital. Os dispositivos FF trocam informações entre si para que as tarefas de controle sejam realizadas nos processadores de cada dispositivo individualmente. Esta estratégia de controle contrasta com o sistema de controle tradicional que utiliza sinais analógicos 4-20mA para transmitir o valor das diversas variáveis do processo a um elemento central de controle que decide quais as operações a serem realizadas em cada dispositivo (Figura 2).

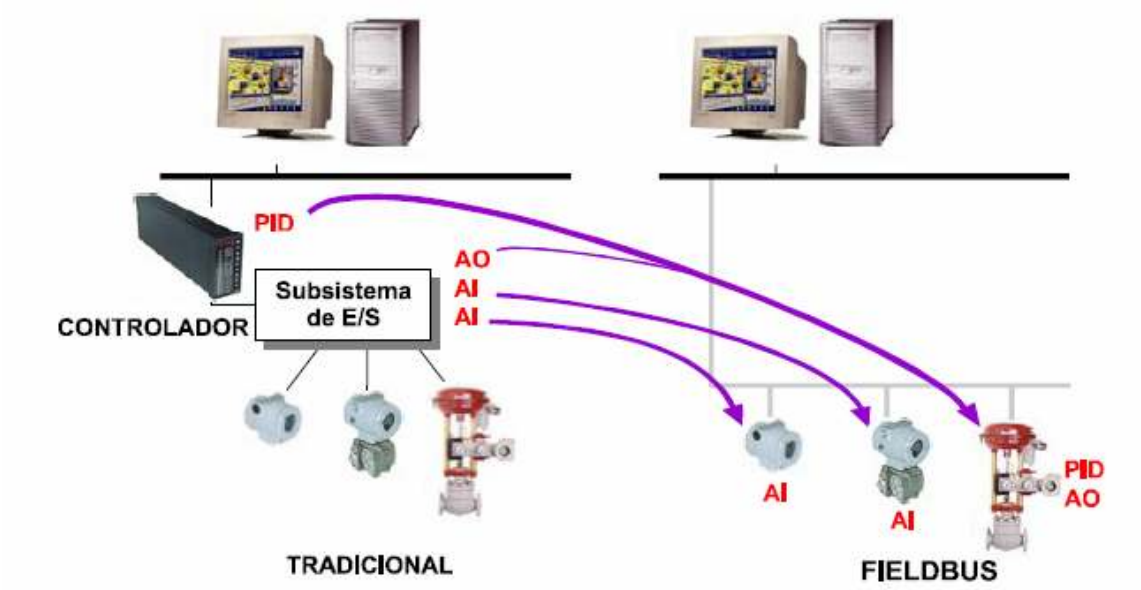

Figura 2 – Comparação das topologias dos sistemas de controle tradicional e do FF
Fonte: Fieldbus Tutorial , SMAR, 2001.

O FF se diferencia dos outros sistemas de comunicação industrial, porque além de definir o meio de transmissão de dados, foi desenvolvido para prover soluções de controle de processos. No FF a estratégia de controle é definida através de blocos funcionais implementados nos próprios dispositivos de campo. Os blocos funcionais são representações do *hardware* do dispositivo que podem ser desde sistemas de I/O a controladores PID.

No FF a estratégia de controle fica distribuída através dos dispositivos de campo, pois os mesmos possuem os blocos funcionais necessários para a regulação do processo industrial e por estarem ligados a um barramento comum conseguem trocar informações entre si.

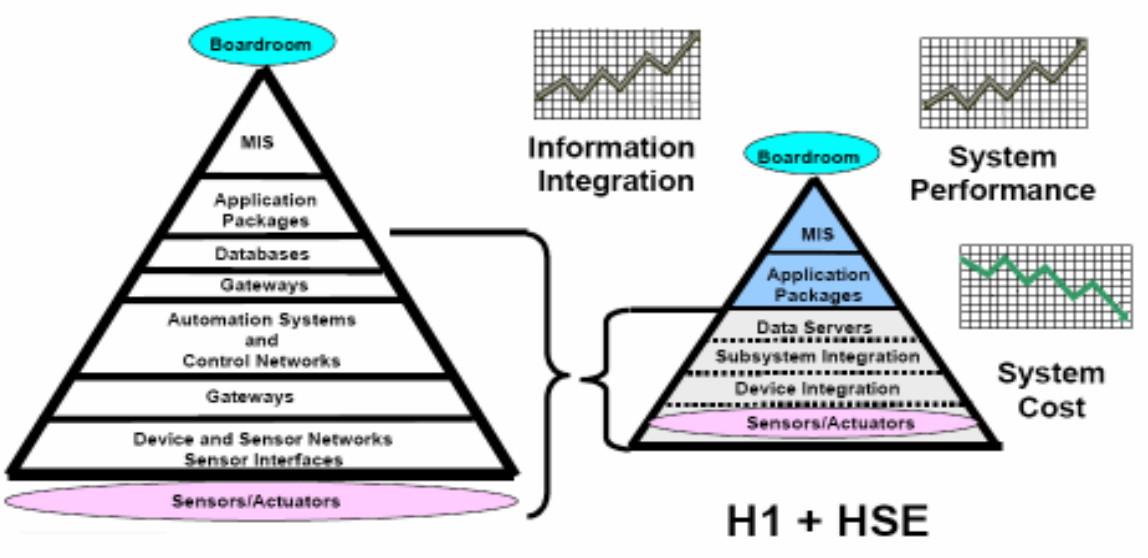

Figura 3 – FF na pirâmide de integração de informações na indústria.
Fonte: Technology Report 2003 – Fieldbus Foundation, 2003.

Com a utilização do FF busca-se a integração de informações na indústria o que favorece a redução dos custos de operação e manutenção da planta. A figura 3 ilustra a redução de camadas na pirâmide de informações na indústria obtida através da utilização do FF. Esta integração possibilita economia uma vez que o fluxo de informações de um sistema para outro passa a ser realizada automaticamente sem a intervenção humana.

O sistema FF prevê ao menos duas estratégias de implementação de sistemas de comunicação denominadas de H1 e HSE, conforme descrito abaixo:

a) Rede H1, baseada na norma IEC 61158-2, opera a 31,25 Kbit/s é utilizada para interconectar dispositivos de campo como sensores, atuadores e dispositivos de I/O;
b) Rede HSE *High Speed Ethernet*, baseada na tecnologia *Ethernet*, trafega tipicamente a 100 Mbps, utilizando par trançado ou fibra óptica como meio de transmissão. Em uma topologia típica, a rede HSE interconecta os diferentes segmentos H1 de uma planta a outros dispositivos de alta velocidade. Tais como Controladores Lógico Programável (CLP), Servidores de dados, Estações de trabalho e até mesmo outros modelos de redes industriais como Profibus e DeviceNet.

Os sistemas H1 e HSE, também chamados de Rede H1 e Rede HSE, serão apresentados mais detalhadamente nos capítulos 2 e 3 deste trabalho. O capítulo 2 fará uma apresentação

do sistema FF H1 detalhando suas principais características. Enquanto que o capítulo 3 apresentará maiores detalhes do sistema FF HSE que é utilizado como *Backbone* do sistema FF e como *Gateway* de outras redes industriais.

Na busca da integração entre o ambiente corporativo e a rede de chão de fábrica, a FF utilizou inicialmente o modelo de comunicações OSI [TANENBAUM, 2007] como referência, mas como as redes TCP/IP [EVANS & WASHBURN, 1996] são mais utilizadas em escritórios e aplicações comerciais, diversos protocolos desta arquitetura foram incluídos na especificação dos dispositivos FF tais como TCP, UDP, IP, DHCP, dentre outros.

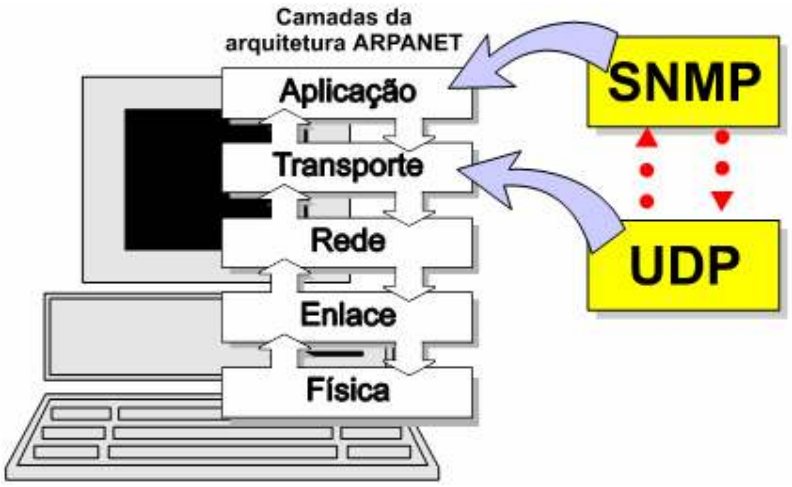

Figura 4 - Camadas da Arquitetura TCP/IP (Aplicação, Transporte, Rede, Enlace e Física).
Fonte: Apontamentos de aula do Prof Msc Sérgio F. Brito

O *Simple Network Management Protocol* (SNMP) [STALLINGS, 1999] é o protocolo da arquitetura TCP/IP utilizado para o gerenciamento de dispositivos, uma vez que é um protocolo robusto, simples e que utiliza poucos recursos dos dispositivos. Além destas características o SNMP já é um protocolo maduro, ou seja, consegue-se desenvolver aplicativos utilizando SNMP com relativa facilidade.

A figura 4 ilustra o modelo de rede simplificado em 5 camadas, destacando a existência do protocolo SNMP na camada de aplicação e o protocolo UDP na camada de transporte. O protocolo UDP será apresentado no capítulo 3 deste trabalho e o capítulo 4 apresentará o protocolo SNMP descrevendo os seus principais componentes e seu princípio de funcionamento.

Os dispositivos FF foram especificados de forma a suportar a utilização do protocolo SNMP e por isto é possível coletar informações dos dispositivos FF através do protocolo SNMP e disponibilizar tais informações para outros sistemas tais como *Manufacturing Execution System* (MES) e *Enterprise Resource Planning* (ERP) [RODRIGUES, 2007].

No capítulo 5 deste trabalho será feito um estudo esclarecendo como os dados são armazenados e como é realizado o fluxo de dados entre os vários dispositivos FF. Também será feita será feita explicação de como se processa a utilização do Protocolo SNMP no gerenciamento dos dispositivos FF.

Ao final deste trabalho serão feitas considerações acerca da existência e funcionalidades do protocolo SNMP nos dispositivos FF. Serão propostos trabalhos posteriores que ajudarão a entender determinadas características do FF.

2 – REDE DE INSTRUMENTAÇÃO – FF H1

O modelo de comunicação do FF é explicado através de um modelo simplificado, composto de três camadas e um nível de aplicações especificas, baseado no modelo de comunicações *Open System Interconnect* (OSI) da *International Standard Organization* (ISO).

O modelo simplificado do FF-H1 é composto pelas seguintes camadas:

1) Camada Física - *Physical Layer* (PHL);

2) Camada de Enlace – *Data Link Layer* (DLL);

3) Camada de Aplicação – composta por *Fieldbus Access Sublayer* (FAS) e *Fieldbus Message Specification* (FMS)

4) Aplicações do Usuário – *User Application* (API).

Conforme ilustrado na Figura 4, apenas o equivalente das camadas Física (1), Enlace (2) e Aplicação (7) do modelo OSI são implementadas no FF. Além destas camadas, o FF define um outro nível denominado de aplicações do usuário que é justamente onde ficam os blocos de funcionais dos dispositivos FF [FIELDBUS FOUNDATION, 2003].

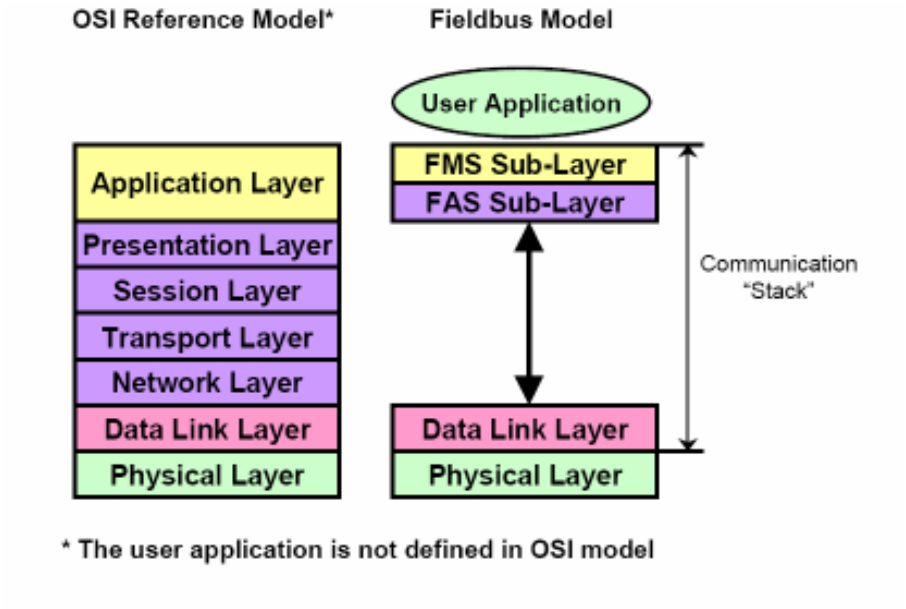

Figura 5 – Comparação entre as camadas do modelo OSI e as camadas do Fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

Observa-se também que no FF são utilizadas duas subcamadas {*Fieldbus Access Sublayer* (FAS) e *Fieldbus Message Specification* (FMS)} para realizarem as tarefas

equivalentes à camada de aplicação do modelo OSI. A subcamada FAS realiza o mapeamento de serviços FMS para a camada de enlace de dados – *Data Link Layer* (DLL).

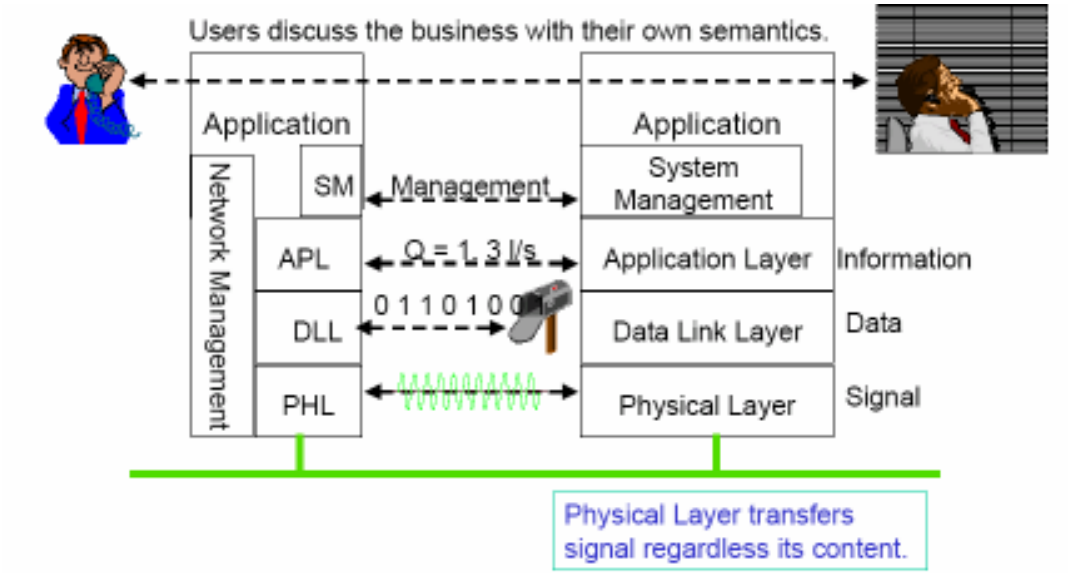

Figura 6 – A comunicação virtual entre as camadas pares no fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

As Figuras 6 e 7 ilustram como é realizada a transferência de dados através do Fieldbus Foundation. Basicamente cada camada adiciona informações de controle chamadas de *Protocol Control Information* (PCI) à mensagem recebida da camada superior [SMAR FF, 2001].

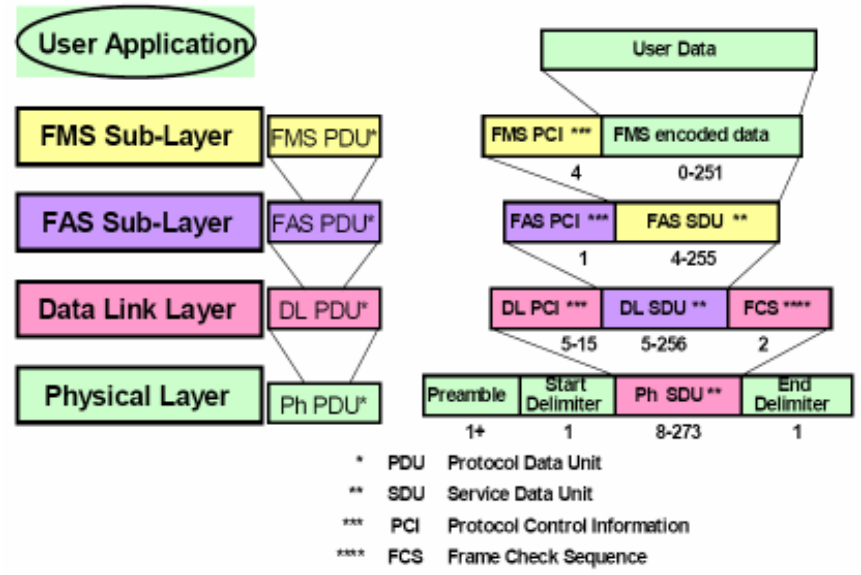

Figura 7 – Transferência de dados no fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

A unidade de dados trocada entre camadas iguais é chamada *Protocol Data Unit* (PDU). A PDU pode conter dados opcionais chamados *Service Data Unit* (SDU) que é a PDU da camada superior. Este fluxo de encapsulamento da mensagem transmitida é ilustrada na figura 7 [YOKOGAWA, 2001].

2.1 – FF H1 - Camada Física

A Camada Física do FF H1 é similar à camada física do modelo OSI e serve para definir os padrões de ligações, fios, cabos e características elétricas necessárias para a formação de uma rede FF. A norma que especifica esses padrões é a ANSI/ISA-S50.02.

A camada física define que para transmissão de dados será utilizada a técnica de codificação Manchester Bifase-L a uma taxa de 31,25kbps, com corrente de até +10mA entregues com uma impedância equivalente de 50 ohms. Além de definir os comprimentos máximos dos cabos a serem utilizados, quantidade de dispositivos que podem pertencer à mesma rede H1, tolerância a falhas, funcionamento crítico da rede e outras características de transmissão.

A Figura 8 exibe o formato típico do sinal transmitido da camada física para o meio. A camada física do receptor reconhece o *bit time* através do preâmbulo, reconhece o limite do preâmbulo através *Start-Delimiter* e através do *End-Delimiter* reconhece o fim do sinal da Camada Física. Todas as informações recebidas entre o *Start-Delimiter* e o *End-Delimiter* é colocado em um buffer de memória para ser acessado pelo *Virtual Communication Relationship* (VCR). O comprimento de preâmbulo pode ser aumentado quando o sinal necessitar ser transmitido através de repetidores [FISHER-ROSEMOUNT, 1998].

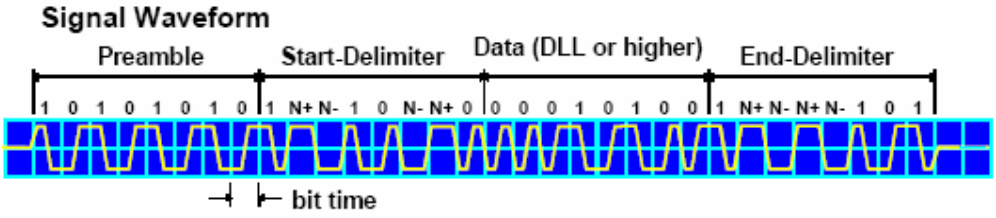

Figura 8 – Formato típico do sinal trocado entre dispositivos fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

2.2 – FF H1 - *Data Link Layer* (DLL)

A camada de enlace de dados *Data Link Layer*(DLL) [ISA s50.02 - IEC/TS 61158-4:1999 parte 4] visa garantir a integridade da mensagem e o controle de acesso ao meio. Para garantir a integridade dos dados transmitidos dois bytes são calculados através de um polinômio aplicado a todos os bytes da mensagem e é acrescentado no final da mesma um quadro chamado de *Frame Check Sequence* (FCS) para que a mensagem enviada seja validada no receptor (Figura 7). Conforme ilustrado na figura 9, a DLL gerencia o acesso dos dispositivos ao meio de transmissão através de um controle determinístico centralizado em um dispositivo chamado *Link Activer Scheduler*(LAS).

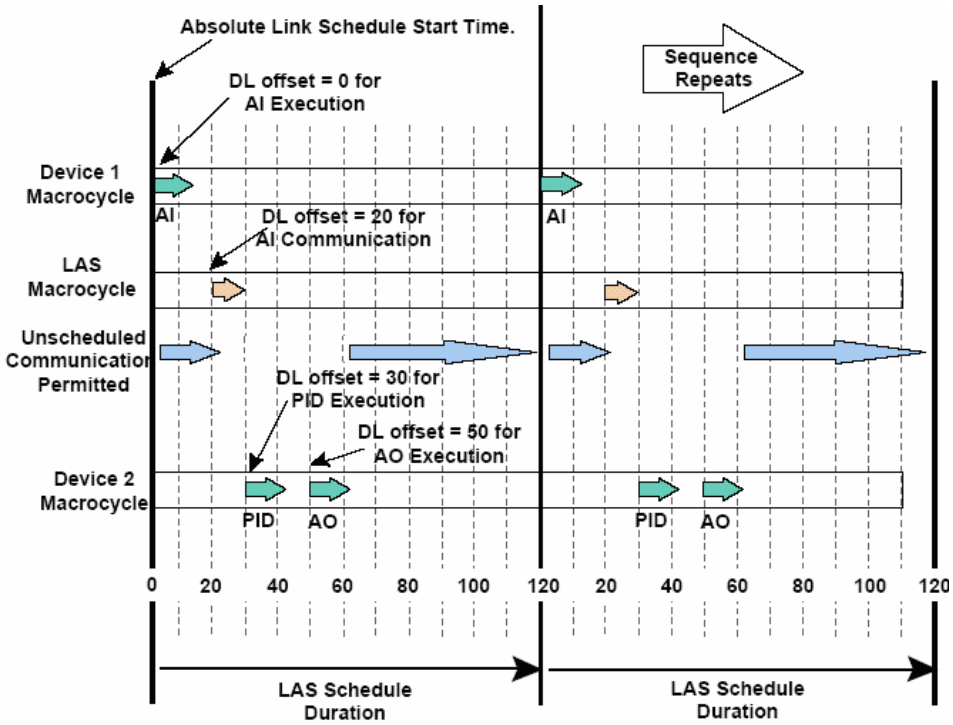

Figura 9 – LAS realizando o controle de acesso ao meio.
Fonte: Fieldbus Technical Overview, Fisher-Rosemount, 1998.

A DLL é o mecanismo para transferir dados de um dispositivo para outros dispositivos que precisam dos dados. Também administra a prioridade e ordem de tais pedidos de transferência. Os interesses de DLL são dados, endereço, prioridade, controle do meio e outros que se relacionarem a transferência de mensagem. Considerando que DLL opera na baixa velocidade Camada Física, tem mecanismos para aperfeiçoar o uso meio de transmissão.

2.2.1 – MAC – *Medium Access Control*

A funcionalidade mais importante de DLL é controle de acesso meio do FF. Como todos os dispositivos do mesmo cabo recebem o mesmo sinal de Camada Física, apenas um deles tem permissão para transmitir um sinal de cada vez. O controle de acesso ao meio é utilizado para alcançar esta meta. O domínio de dispositivos que compartilham o mesmo sinal de Camada Física é chamado *link*. Em outras palavras, somente um dispositivo de cada *link* pode usar o meio para transmitir por vez.

O *Link Active Scheduler* (LAS) tem o papel de controlar o acesso meio. Basicamente o dispositivo só pode transmitir quando recebe a permissão de transmissão através de um *token*. O LAS possui a *token* e disponibiliza para um dispositivo transmitir, quando este dispositivo terminar de transmitir ele devolve a permissão para o LAS que concede permissão a outro dispositivo da rede.

Como as mensagens da aplicação têm vários níveis de urgência, a DLL possui um mecanismo para transmitir mensagens de acordo com a urgência. Este mecanismo possui três níveis de prioridade: URGENTE, NORMAL e TIME_AVAILABLE, nesta ordem. Uma mensagem URGENTE é transmitida imediatamente mesmo que outras mensagens com prioridade NORMAL ou TIME_AVAILABLE estejam na fila de espera. A tabela 1 mostra o tamanho da mensagem que pode ser enviado comparado com a sua classificação de prioridade [EMERSON, 2002].

Priority	Maximum Data (DLSDU) Size
URGENT	64 bytes
NORMAL	128 bytes
TIME_AVAILABLE	256 bytes

Tabela 1: Comparação tamanho máximo X prioridade da mensagens fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

2.2.2 – Endereçamento (DL-address)

A camada de enlace dos dispositivos possui um endereço físico chamado *DL-address* que é composto de três componentes: *Link, Node* e *Selector* conforme ilustrado na figura 10 [YOKOGAWA, 2001]. O campo *Link* possui 16 bits e serve para identificar o barramento em que o dispositivo está ligado. Quando a comunicação for entre elementos do mesmo *link* este

campo pode ser omitido, mas o mesmo é necessário quando uma mensagem estiver passando para outros *links* através de *bridges* [PERLMAN, 2000].

Figura 10 – Endereçamento dos dispositivos fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

O campo *Node* com 8 bits dá o endereço físico do dispositivo. Os dispositivos FF podem ter um endereço físico entre 0x10 e 0xFF e são classificados nas faixas *Link Master* (LM), BASIC, DEFAULT ou temporária. Normalmente dispositivos estão na faixa LM ou BASIC de acordo com sua classe de dispositivo. Quando um dispositivo perde o seu endereço físico, ele usa um endereço DEFAULT para se comunicar e conseguir novo endereço. O LAS tem o endereço de nodo 0x04.

A figura 11 ilustra a faixa de endereços utilizados em um *link* FF. Existe uma faixa de endereços de tamanho V(NUN) que não deve ser utilizado no endereçamento de dispositivos, caso um dispositivo utilize um endereço desta faixa não conseguirá receber ou enviar mensagens com o *Link*. Os endereços V(FUN) e V(NUN) são parâmetros que podem ser acessados pelo administrador de Rede.

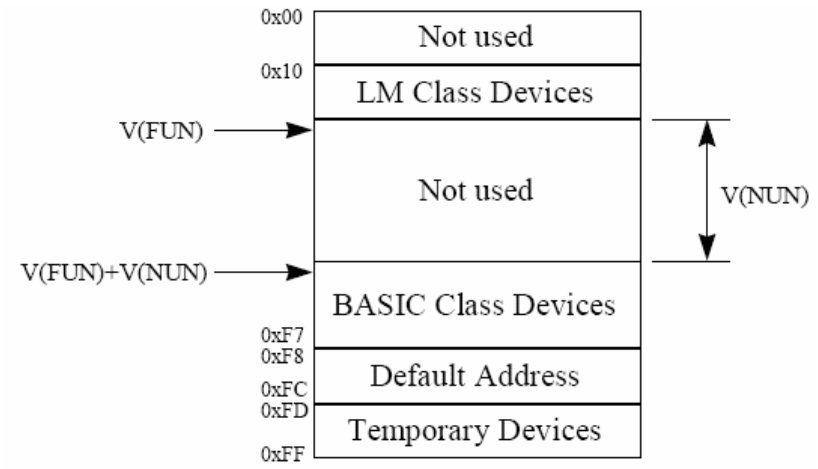

Figura 11 – Distribuição de endereços físicos dos dispositivos fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

O campo de *Selector* contém 8 bits destinados ao endereçamento interno do VCR no dispositivo. Quando um VCR é conectado a outro VCR, o mesmo é identificado com o

DLCEP (*Data Link Connection End Point*) mostrado neste campo. Quando um VCR não está conectado a outro, mas disponível para troca de mensagens a identificação com DLSAP (*Data Link Service Access Point*) é mostrado neste campo. DLCEP e DLSAP têm faixas diferentes.

Alguns endereços DL-Address são reservados para um propósito especifico e são compartilhados por diversos dispositivos. Uma aplicação deste tipo pode ser um alarme associado a diversos dispositivos.

2.2.3 – *Link Active Scheduler* (LAS)

O *Link Active Scheduler* (LAS) tem a função de controlar o acesso ao meio para transmissão.

Os dispositivos são classificados por classes: BASIC, *Link Master* (LM) e *Bridge*. Os dispositivos da classe LM tem a capacidade de operar como LAS, enquanto que os dispositivos BASIC não tem esta capacidade. Os equipamentos da classe *Bridge* além de ter a capacidade de funcionar como LAS possuem a funcionalidade de conectar *links*.

Apenas um e somente um dispositivo pode funcionar como LAS por vez. Assim apenas um dispositivo LM ou *Bridge* é necessário por link. OS Dispositivos de LM tentam adquirir papel de LAS quando nenhuma LAS existir na partida ou quando o LAS atual falha. O dispositivo de LM com o menor endereço *node* passa a ser o novo LAS. Outros dispositivos de LM observam a atividade de LAS e assumem seu papel quando LAS falhar. A Figura 12 ilustra o procedimento no qual um determinado dispositivo classe LM se torna as LAS.

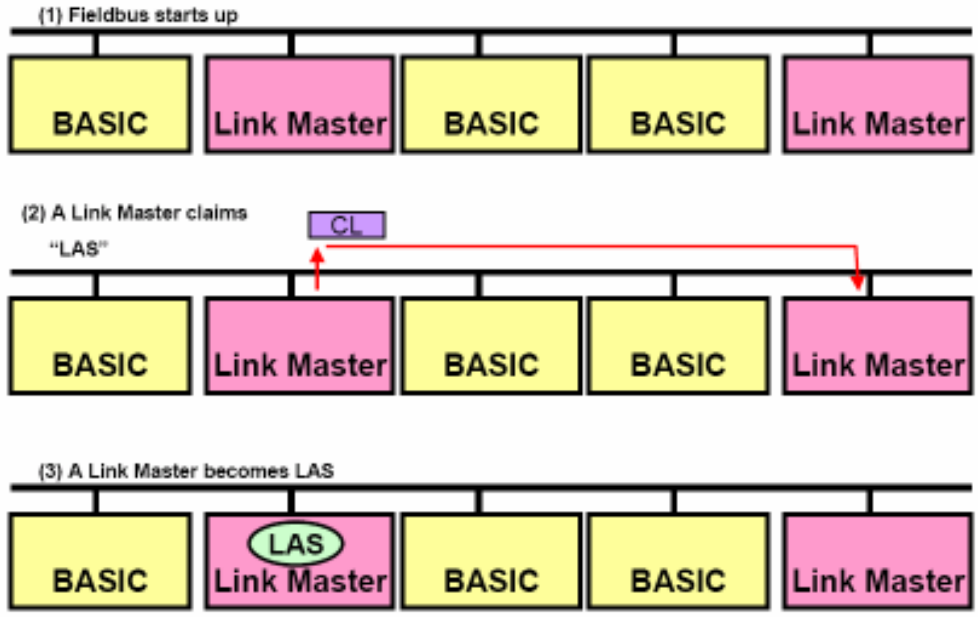

Figura 12 – Situações em que o dispositivo LM se torna LAS
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

Note que LAS é uma funcionalidade adicional à comunicação básica. Por isso possui DL-address diferente (0x04) do endereço *node.*

2.2.4 – Comunicação Agendada

O LAS é responsável pelas comunicações agendadas que são necessárias para a execução dos blocos de funções no link. Blocos de funções são aplicações distribuídas que operam de forma sincronizada. O LAS gerencia a sincronização da transferência de dados.

Os parâmetros de saída de um bloco de função são publicados no formato de dados que podem ser recebidos por outros blocos de funções chamados de *Subscribers*. O LAS controla periodicamente a transferência de dados entre os *Publishers* e os *Subscribers* usando a rede de agendamento.

Quando o tempo agendado para a transmissão do dispositivo é atingido, o LAS envia a permissão para transmitir *Compel Data* (CD) para o publicador DLCEP. Assim o dispositivo pode publicar imediatamente os dados presentes no seu buffer de saída para o publicador DLCEP que irá transmitir até que se esgote o tempo para transmissão ou todos os dados sejam transmitidos. Este processo é ilustrado na Figura 13.

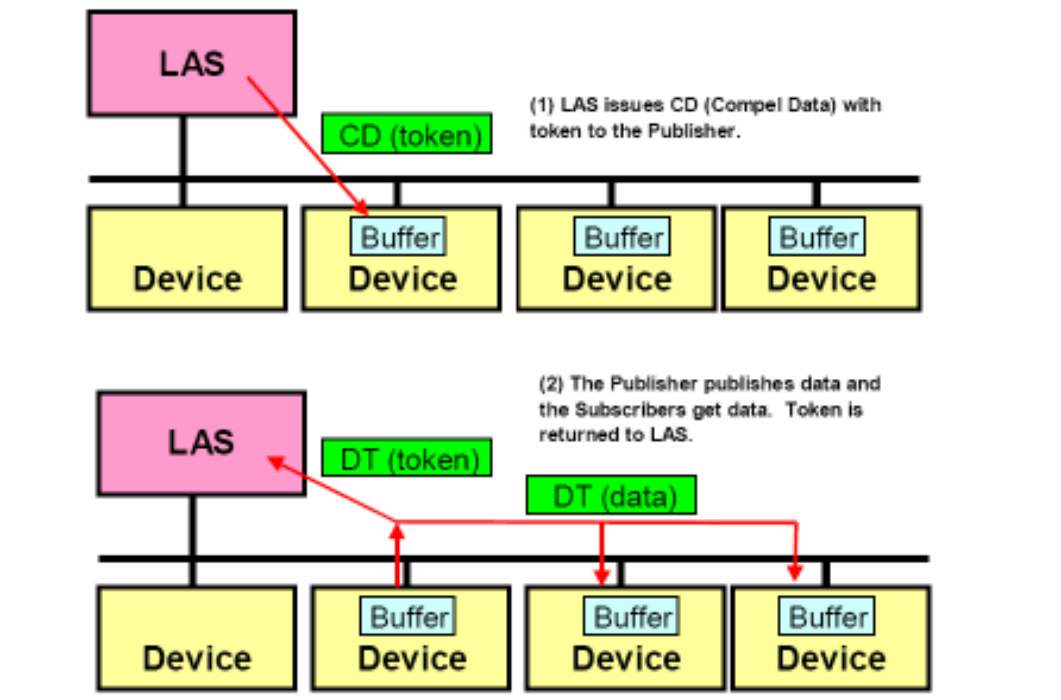

Figura 13 – O LAS controla a passagem da *token* para que todos os dispositivos possam transmitir.
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

DLL adiciona informações atualizadas ao PCI para que os receptores identifiquem que os dados foram atualizados desde a última publicação.

2.2.5 – Comunicação Não-Agendada

Outras comunicações acontecem em modo assíncrono. O LAS é responsável por dar a todos os nodos do link uma chance para enviar mensagens. O LAS dá a permissão através do *Pass Token* para um nodo. Um PT PDU contém prioridade e informação de intervalo de tempo. Quando o nodo não tem mensagens de determinada ou mais alta prioridade para ser enviado, ou o intervalo de tempo determinado para o dispositivo é expirado, o mesmo devolve a permissão *Return Token* (RT) PDU.

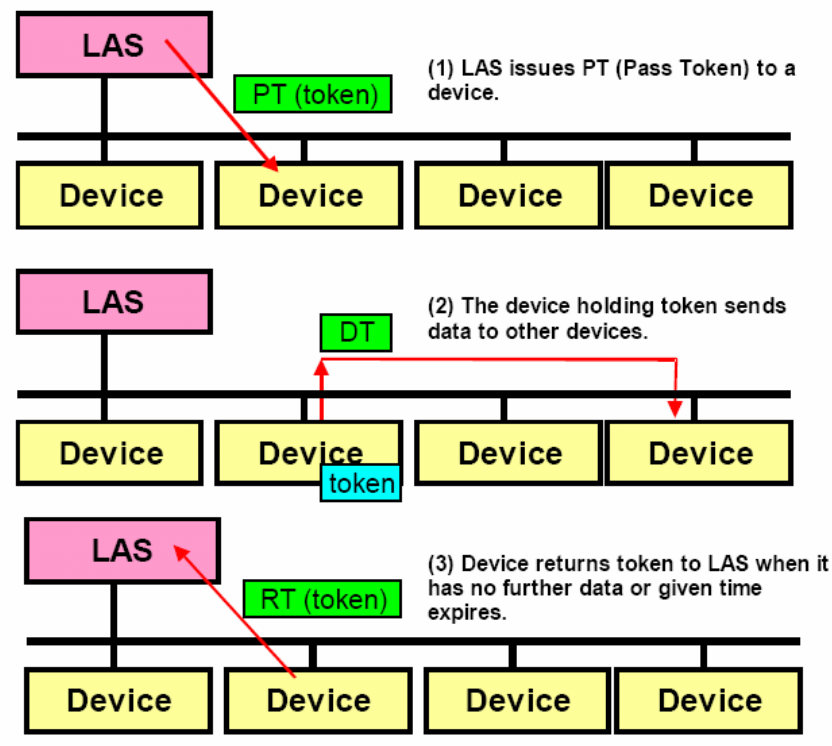

Figura 14 – LAS realizando o controle de acesso ao meio para comunicações não agendadas.
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

O LAS controla a transferência de mensagens atualizando a prioridade. Quando a permissão é dada a todos os dispositivos em um intervalo de tempo pequeno, o LAS dá mais tempo aos nodos abaixando a prioridade. Quando o *token* não vai para todos os dispositivos dentro de um tempo designado O LAS aumenta a prioridade de forma que a permissão passe por todos dispositivos em um intervalo de tempo desejado (Figura 14).

Cada dispositivo deve retornar a permissão dentro do intervalo de tempo designado na PT PDU, isto é necessário para garantir que as comunicações não-agendadas sejam concluídas antes da próxima comunicação agendada. Note que o *token* é dado ao nodo e não ao DLCEP ou DLSAP, assim o dispositivo é responsável para permitir que todos os DLCEPs e DLSAPs no dispositivo possam enviar mensagens.

2.2.6 – Manutenção da Lista de Dispositivos ativos

A terceira tarefa do LAS é realizar a manutenção do link. Como o LAS deve dar permissão a todos os dispositivos do link para publicar, quando um novo dispositivo é

adicionado a rede ele deve ser reconhecido pelo LAS e incluído na lista de rotação de permissão chamada de *Live List*. O LAS envia uma PDU *Probe Node* (PN) para o endereço no nó que não foi reconhecido anteriormente. Este novo elemento espera o PN e responde com o PDU *Probe Response* (PR) para o LAS. Assim o LAS adiciona este dispositivo ao *Live List* e este passa a receber permissão para publicar.

Quando um dispositivo é removido do link e deixa de responder ao PT por três vezes, o LAS detecta isto e remove o dispositivo do *Live List*. Quando uma mudança é detectada no *Live List* o LAS publica esta alteração para todos os dispositivos LM possuam a mesma lista e estejam prontos para entrar em operação em caso de falha do LAS.

Periodicamente o LAS também publica o *Data Link Time* (LStime) que visa realizar a sincronização na execução de blocos de funções em todos os dispositivos da rede. O *Data Link Time* também é chamado de *network time.*

2.3 – FF H1 – Camada de Aplicação

Conforme mencionado anteriormente, no FF são utilizadas as subcamadas: Fieldbus Access Sublayer (FAS) e Fieldbus Message Specification (FMS) para realizarem as tarefas equivalentes à camada de aplicação do modelo OSI.

2.3.1 – FAS – *Fieldbus Access Sublayer*

O *Fieldbus Access Sublayer* (FAS) é uma parte da comunicação segura. Como o FF não tem as camadas de 4 a 6 do modelo OSI entre as camadas DLL e APL, o FAS mapeia as solicitações da camada de aplicação diretamente em serviços da camada DLL. Esta é a parte mais importante do gerenciamento VCR.

As comunicações agendadas e não agendadas trocadas entre aplicações dos dispositivos FF são funções da DLL. A camada FAS utiliza estas funções para proporcionar serviços à camada FMS. Estes serviços são descritos por VCRs *Virtual Communication Relationship*. A VCR é como uma abreviatura de acesso, uma pequena estrutura que resume um conjunto maior de dados pré-armazenados. Nestes canais os PCI são inseridos no remetente e retirados no destinatário conforme ilustrado na figura 14.

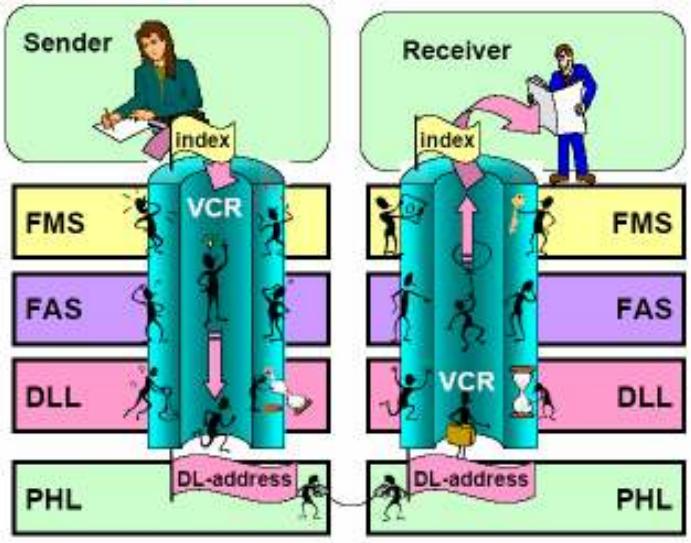

Figura 15 – Comunicação no fieldbus através de VCR
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

Um dispositivo FF tem vários VCRs com o objetivo de poder se comunicar ao mesmo tempo com vários dispositivos ou aplicações. Isto é possível porque o VCR garante que a mensagem vai para o elemento correto sem riscos de perder informações. Um VCR é identificado por um endereço local atribuído na camada de aplicação chamado *index*. Os outros dispositivos da rede entendem o *index* como o endereço da camada de enlace (DL-address) do dispositivo (Figura 15).

Os principais modelos de comunicação do FAS são:

a) Cliente - servidor: Utilizada para comunicação enfileirada, não escalonada, iniciada pelo usuário, um para um, entre dispositivos no FF. Enfileirada implica que as mensagens são enviadas na ordem fornecida para transmissão, respeitando suas prioridades, sem sobrescrita das mensagens anteriores. Quando um dispositivo recebe um *token* ele coloca uma mensagem no barramento. Ele é dito cliente da comunicação e o destino da mensagem é o servidor. Quando o servidor recebe o *token* do LAS ele responde à pergunta recebida. [SEIXAS FILHO FF, UFMG].

b) Distribuição de Relatório: Utilizada para comunicação enfileirada, não escalonada, iniciada pelo usuário, um para muitos, entre dispositivos no FF. Quando um dispositivo com um evento ou relatório de tendência recebe o *token* do LAS, ele envia a mensagem para um grupo de endereços representado pelo VCR. Dispositivos

interessados em receberem a mensagem identificada pelo VCR irão receber o evento ou relatório.

c) Produtor - Consumidor: É utilizado para comunicação *bufferizada* de um para muitos. *Bufferizado* quer dizer que apenas a última versão da informação é mantida. O dado mais recente sobrescreve o dado anterior. Quando um dispositivo recebe a mensagem CD do LAS, ele transmite uma mensagem. Este dispositivo é chamado de produtor. Todos os dispositivos interessados nestas informações irão recebê-la. Estes dispositivos são os assinantes ou consumidores. A mensagem CD pode ser escalonada no LAS ou enviada aos assinantes de forma não escalonada. Um atributo do VCR irá determinar qual dos dois mecanismos será utilizado.

2.3.2 – FMS – *Fieldbus Message Specification*

O FMS faz a interface entre as aplicações do usuário e os serviços FF. Quando serviços são solicitados pelas aplicações do usuário, o FMS codifica esta solicitação e transfere para as outras aplicações. O FMS receptor decodifica o pedido para notificar a aplicação.

Segundo a *Fieldbus Foundation* o VFD *Virtual Field Device* é uma espécie de atalho para visualizar remotamente os dados de um dispositivo local, a partir de outro equipamento, que pode ser outro dispositivo de campo ou até mesmo uma estação de trabalho.

Os dispositivos FF possuem ao menos duas VFDs: Uma VFD de gerência e outra VFD de blocos de funções. O VFD de gerência, que é usado para configurar os parâmetros de rede e do dispositivo, contém as aplicações de gerenciamento da rede e do sistema. Um dispositivo de campo pode ter mais que uma VFD de blocos de funções que realizam as tarefas do dispositivo (Figura 16).

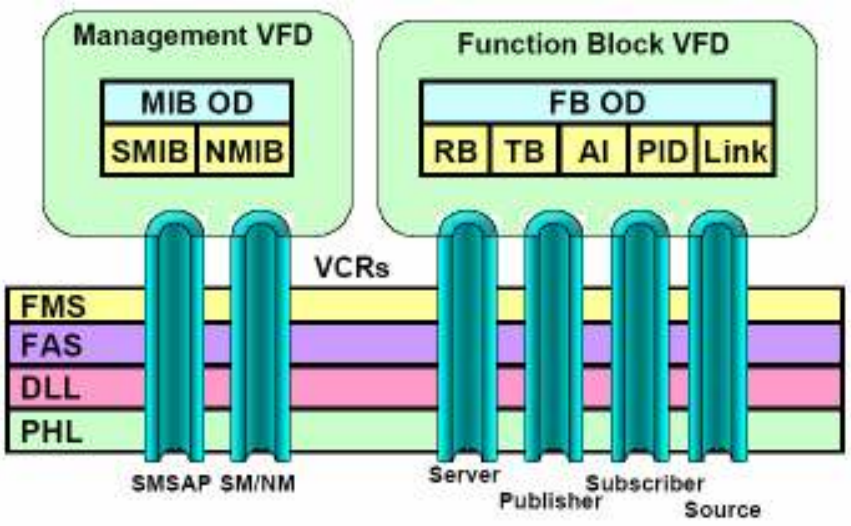

Figura 16 – Os dispositivos FF possuem ao menos as VFDs de Gerência e de Blocos de Função
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

O comportamento da rede é gerenciado através dos objetos da NMIB *Network Management Information Base* o comportamento do sistema é gerenciado através dos objetos da SMIB *System Management Information Base*. Agendamentos e VCRs também são objetos.

O mesmo VFD utilizado para administração de rede também é usado para administração do sistema. Este VFD provê acesso para as informações da NMIB e da SMIB. Os dados da NMIB incluem VCRs, variáveis dinâmicas, estatísticas, e agendamento do LAS, caso se trate de dispositivo mestre. Os dados da SMIB incluem informações de endereço e identificação do dispositivo e agendamento para execução de blocos de funções.

Aplicações em uma VFD são mostradas para as outras aplicações na rede usando um *Object Model* que consiste dos atributos, comportamentos e métodos de acesso. Blocos de funções têm objetos/parâmetros que podem ser acessados por outras aplicações. Por exemplo, alarmes são objetos que podem ser manipulados por outros aplicativos. Este comportamento é definido pela especificação da aplicação do bloco de função.

2.4 – FF H1 – *User Application* (API)

O *User Application* (API) utiliza o conceito de blocos para realizar todas as suas funções. Conforme ilustrado na figura 17, existem três tipos de blocos básicos: blocos de recursos, blocos de função e blocos de transdutores. Os blocos de recursos são utilizados para configurar os dispositivos. Blocos de função são utilizados para construir a estratégia de

controle. O bloco transdutor serve para desacoplar os blocos de função das interfaces com o dispositivos de campo.

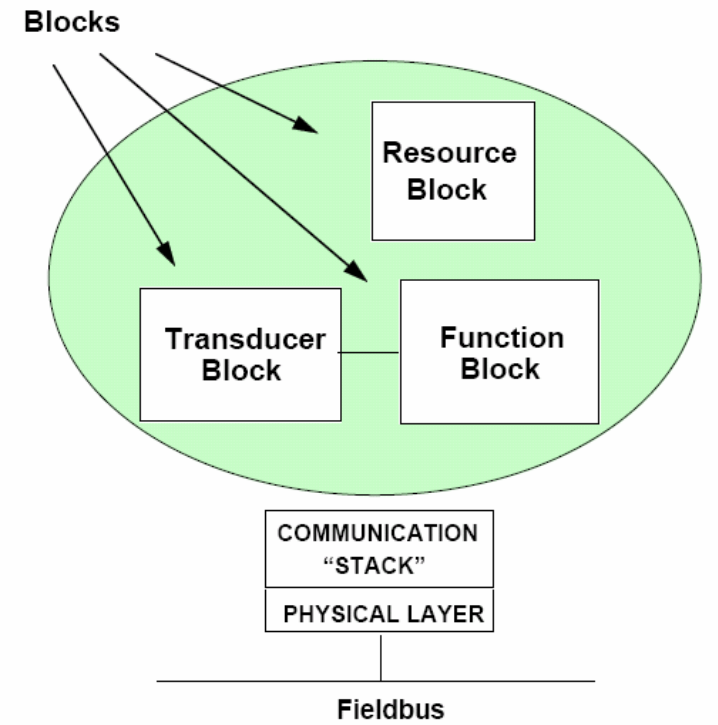

Figura 17: Tipos de blocos existente na camada User Application de dispositivos FF – H1
Fonte: Technical Overview, Fieldbus Foundation, 2003.

Os blocos de recurso descrevem as características dos dispositivos de FF como o nome de dispositivo, ID do fornecedor, capacidade de memória, versão do dispositivo e número de série dentre outras. Há apenas um bloco de recurso em cada dispositivo, contendo configurações gerais para o VFD. Embora sejam visíveis externamente não podem ser interligados nem participar do escalonamento estabelecido pelo LAS.

Os blocos transdutores desacoplam os Blocos de função das funções de entrada/saída locais requeridas para leitura do sensor e comando da saída do *hardware*. Eles contêm informações como dados de calibração e tipo de sensor. Embora os blocos transdutores possam executar tarefas a uma freqüência superior à dos blocos de função e sejam visíveis, não podem ser conectados via ferramenta de configuração e desta forma não podem ser escalonados pelo sistema de gerenciamento. Normalmente há um transdutor para cada entrada e saída do bloco.

Os Blocos de função provêm o controle comportamento de sistema. Os parâmetros de entrada e saída dos blocos de função podem ser ligados em cima do FF. A execução de cada Bloco de Função é precisamente agendada. Pode haver muitos blocos de função em uma única aplicação de usuário.

A tabela 02 ilustra os dez principais blocos de função definidos pela Fieldbus Foundation.

Function Block Name	Symbol
Analog Input	AI
Analog Output	AO
Bias	B
Control Selector	CS
Discrete Input	DI
Discrete Output	DO
Manual Loader	ML
Proportional/Derivative	PD
Proportional/integral/Derivative	PID
Ratio	RA

Tabela 2: Tipos e simbologia dos principais blocos de função do FF – H1
Fonte: Fieldbus Tutorial, SMAR, 2001.

Observe que a quantidade e tipos de blocos de função em um dispositivo dependem do tipo de dispositivo, por exemplo, um transmissor simples de temperatura/pressão/vazão possui um único bloco AI. Uma válvula de controle pode conter um bloco PID além do bloco AO.

A figura 18 ilustra um sistema de controle em malha fechada composto de um sensor de vazão e uma válvula de controle. Nesta aplicação o sensor é modelado como uma entrada analógica enquanto que a válvula de controle funciona como PID e como saída analógica.

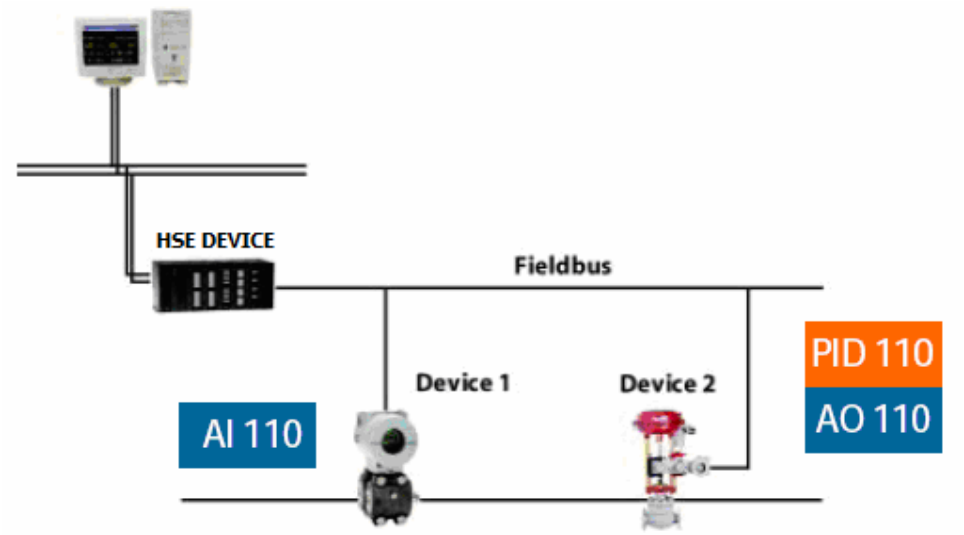

Figura 18: Os tipos de bloco função e a quantidade em cada dispositivo FF dependem do tipo de dispositivo.
Fonte: Fieldbus Tutorial, SMAR, 2001.

A função de um dispositivo fieldbus é determinada pelo arranjo e interconexão dos blocos. É justamente o arranjo dos blocos que determinará que o sensor de vazão funcione como uma entrada analógica e que a válvula de controle funcione com as funções de saída analógica e PID. A figura 19 ilustra a interconexão de blocos, destacando a possibilidade de existirem diversos blocos de função e blocos transdutores em um mesmo dispositivo, mas existe apenas um bloco de recurso por dispositivo [SMAR FF, 2001].

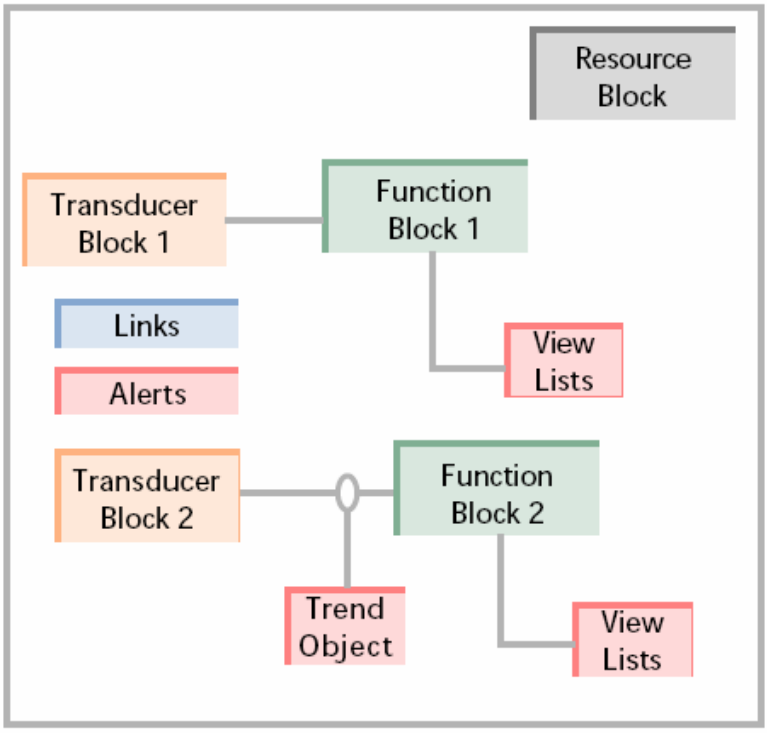

Figura 19: A função de um dispositivo fieldbus é determinada pelo arranjo e interconexão dos seus blocos.
Fonte: Fieldbus Tutorial, SMAR, 2001.

Um objeto é identificado por um número chamado *index* que é único para cada VFD. Em sistemas abertos para descrever/explicar um objeto são necessárias informações adicionais chamadas de OD *Object Dictionary*. Uma aplicação Cliente pode ler explicações com *Get OD* e ler o valor quando o objeto for uma variável.

O objeto mais fundamental é a variável para conter um valor. Pode ser uma variável simples, um registro (estruturado) ou um vetor. Parâmetros dos blocos de funções, VCR, NMIB e SMIB são exemplos de variáveis de registro. Outros objetos são eventos, domínio e programa.

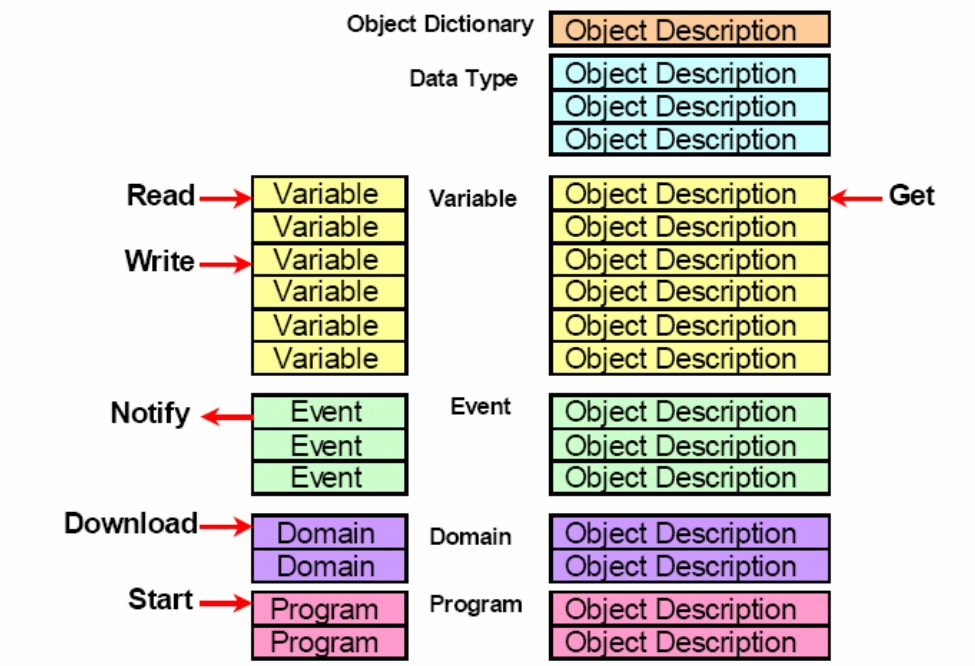

Figura 20 – Ilustração de objetos e propriedades em MIBs do fieldbus
Fonte: Fieldbus Book - A Tutorial, Yokogawa Electric Corporation, 2001.

Um objeto é acompanhado com seu OD que compartilha o mesmo índice. Existem ODs que não estão associados a objetos, eles fornecem informações como local do objeto, quantidade de objetos, tipos de dados, estrutura dos dados e assim por diante (Figura 20).

Assim fica evidenciada a utilização de MIB no gerenciamento dos dispositivos FF-H1. No capítulo 4 deste trabalho será realizado um estudo sobre o SNMP onde será explicado o

que é MIB e como esta estrutura de dados é utilizada para monitorar e controlar o funcionamento dos dispositivos.

3 – REDE DE BACKBONE – FF HSE *HIGH SPEED ETHERNET*

Na arquitetura de sistema de controle proposta pela *Fieldbus Foundation*, a Rede H1 é utilizada para interconectar dispositivos de campo tais como transmissores, atuadores e sensores, enquanto que a Rede HSE é utilizada para interconectar diferentes segmentos H1 e dispositivos com interface Ethernet com taxa de 100 Mbps, tais como estações de trabalho e CLPs (Figura 21) [FIELDBUS FOUNDATION, 2003].

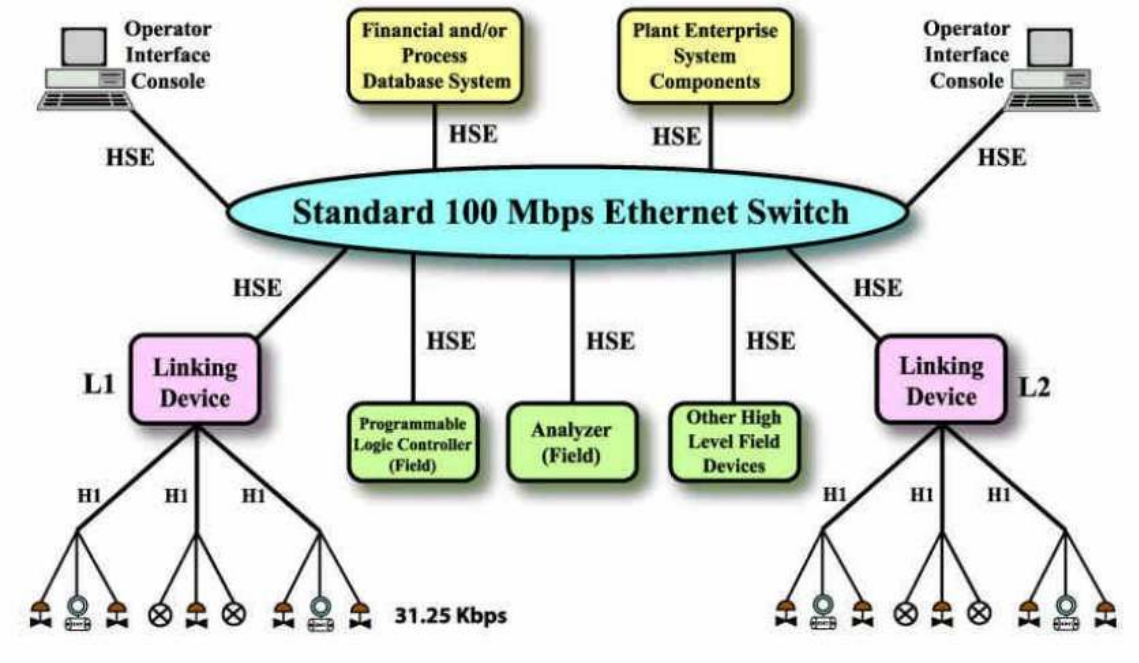

Figura 21 – O Sistema HSE FF possibilita a integração de Redes H1 com PLCs e Redes Ethernet
Fonte: Apostila Foundation Fieldbus, Constantino Seixas - UFMG.

Por outro lado, temos que pesar que FF foi desenvolvido na busca de um barramento que integrasse as tecnologias prévias mas representativas, e como tal é muito genérico admitindo múltiplos métodos de comunicação e tipos de serviços. Isto permite que HSE não somente interconecte dispositivos HSE, segmentos H1 e dispositivos de outras redes industriais como Profibus e DeviceNet, sem perda de informações ou funcionalidade.

Conforme a figura 21, o Sistema FF HSE utiliza a arquitetura TCP/IP com o protocolo Ethernet na camada física. Na especificação do FF HSE foram incluídos diversos protocolos da arquitetura TCP/IP, todos os protocolos do sistema FF H1 e outros protocolos para

viabilizar o acesso às informações dos dispositivos H1 e a disponibilização destas informações para outros dispositivos, tais como estações de trabalho.

Conforme ilustrado na figura 22, os dispositivos HSE são dispositivos FF que contem um FDA agente, um sistema de administração de kernel, *System Management Kernel* (SMK), um agente de gerenciamento de rede, *Network Management Agent* (NMA) e VFDs. O dispositivo HSE possui sua própria pilha de comunicações de forma a converter as mensagens H1 em HSE, ao invés de simplesmente empilhar pacotes H1.

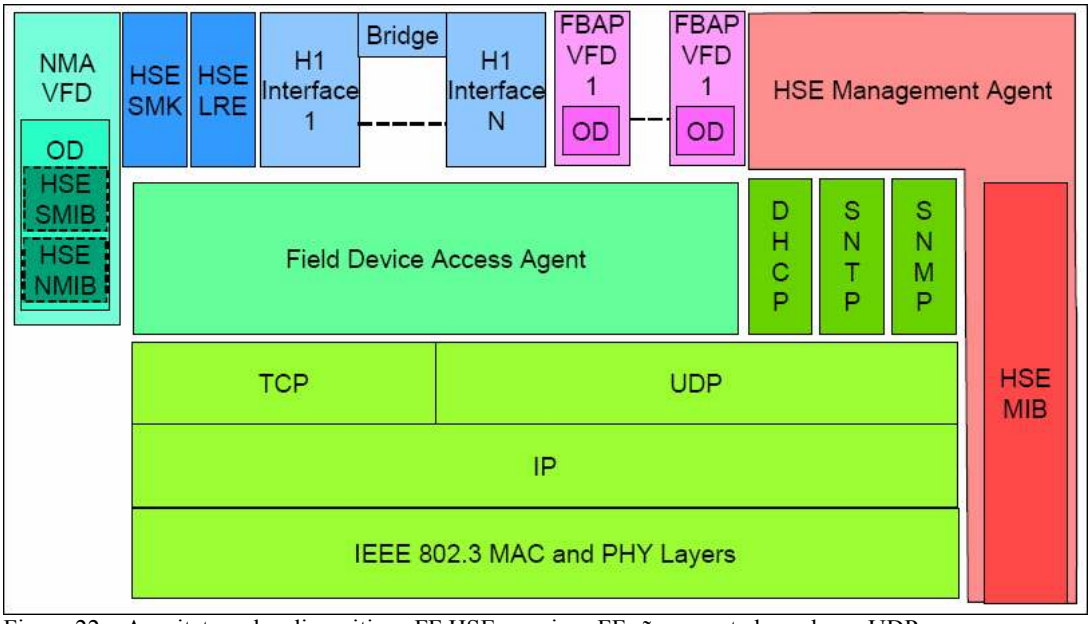

Figura 22 – Arquitetura dos dispositivos FF HSE: serviços FF são executados sobre o UDP.
Fonte: Fieldbus Foundation Comunications, Emerson Proccess Management, 2002.

Existem quatro tipos básicos de dispositivos HSE, mas tipicamente estes são combinados no mesmo dispositivo: *Linking Device* (LD), *Ethernet Device* (ED), *Host Device* (HD) e *Gateway Device* (GD). Estes elementos são descritos abaixo:

a) LD – são os dispositivos HSE que conectam os dispositivos HSE a uma ou mais redes H1. Eles provêem serviços de roteamento para a transmissão de mensagens FMS entre redes H1.
b) GD – são dispositivos HSE semelhantes aos LD que servem para interconectar um ou mais sistemas de I/O ou barramentos. Os blocos de funções Multiples Input/Output (MIO) são idéias para a comunicação de protocolos Modbus ou Profibus-DP que contem muitas informações de Input/Output do processo.

c) HD – são dispositivos com capacidade de comunicar-se com dispositivos HSE. Por exemplo, estações de trabalho.

d) ED – são dispositivos HSE que provêem serviços de inicialização da pilha de comunicação *Ethernet* tais como sincronização do tempo e gerenciamento *Ethernet*. O ED pode executar alguns blocos de funções e pode operar como um sistema de I/O.

O ED é uma adição ao modelo de administração de sistema e rede H1. O ED não substituiu o modelo H1, apenas adiciona serviços para o gerenciamento HSE. Este elemento, por exemplo, habilita a operação do protocolo SNMP para o gerenciamento dos dispositivos H1.

Conforme a figura 23, o modelo simplificado do FF-HSE é composto pelas seguintes camadas:

1) Camada Física - IEEE 802.3u – Fast Ethernet 100Mbps;
2) Camada de Enlace – IEEE 802.3 - MAC;
3) Camada de Rede – IP;
4) Camada de Transporte – TCP/UDP;
5) Camada de Aplicação – *Field Device Access* (FDA);
6) Aplicações do Usuário – *User Application* (API).

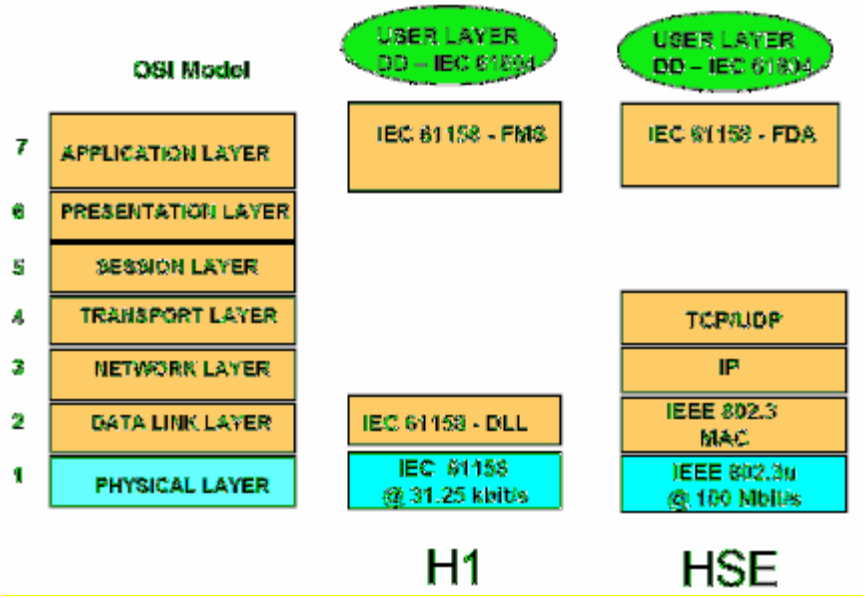

Figura 23 – Comparação entre as camadas existentes no FF-HSE com o modelo OSI.
Fonte: Technology Report to the 20013 General Assembly, Fieldbus Foundation, 2003.

A figura 23 ilustra a comparação entre o modelo de camadas OSI e os modelos FF-H1 e FF-HSE, desta figura fica evidente que as aplicações de usuário suportadas no FF-H1 também são suportadas pela FF-HSE.

3.1 – FF-HSE – Camada Física: IEEE 802.3u

A especificação de nível físico IEEE 802.3u, popularmente conhecida como 100BASE-T define que as estações são interligadas a um *hub*, por ligações ponto a ponto, segundo a topologia estrela. A especificação 100BASE-T engloba as opções de nível físico 100BASE-TX, 100BASE-T4 e 100BASE-FX, uma interface padrão denominada MII e um repetidor 100BASE-T.

As interfaces *Fast Ethernet* incluem especificações de mecanismos de Auto-Negociação de velocidade dos meios de comunicação. Isto possibilita que os fabricantes disponibilizem interfaces de dupla velocidade, ou seja, pode operar com 10Mbps ou com 100Mbps automaticamente.

Os padrões 100BASE-TX e 100BASE-FX usados em Fast Ethernet são ambos aprovados a partir de normas de camada física desenvolvida pela *American National Standards Institute* (ANSI).

A figura 24, ilustra o PDU transmitido no meio físico do protocolo ethernet.

7	1	6	6	2	46 =< n =< 1500	4bytes
Pre	SFD	DA	SA	Length Type	Data unit + pad	FCS

Figura 24: Estrutura do protocolo Fast Ehernet – 100 Mbps Ethernet (IEEE 802.3u)
Fonte: Site http://www.javvin.com/protocolFastE.html, Acesso em Maio de 2008.

- *Preamble* (PRE) - 7 bytes. A PRE está alternando um padrão de uns e zeros que diz as estações receptoras que um frame se aproxima, e que proporciona um meio para recebimento dos quadros na camada físicas;
- *Start-of-frame delimiter* (SOF) - 1 byte. A SOF é uma alternância padrão de zeros e uns, terminando com dois bits consecutivos que indica que os próximos bytes fazem parte da mensagem;
- *Destination address* (DA) - 6 bytes. O campo DA identifica qual ou quais estações devem receber os quadros;

- *Source addresses* (SA) - 6 bytes. O Campo SA identifica a estação que enviou os quadros;
- *Length/Type* - 2 bytes. Este campo indica tanto o número de bytes MAC-client dados que estão contidos no campo de dados do quadro, ou do tipo ID *frame*, se for montada usando um modelo facultativo;
- *Data* - é uma seqüência de n bytes (46 = <n = <1500) de qualquer valor. O tamanho mínimo do quadro é 64bytes;
- *Frame Check Sequence* (FCS)- 4 bytes. Este campo contém um seqüência de 32 bits *Cyclic Redundancy Check* (CRC), que é criado pela MAC de envio e recalculada pela MAC do receptor para verificar se há quadros danificados.

3.1.1 – CSMA/CD – *Carrier Sense Multiple Access/Collision Detection*

O CSMA / CD (*Carrier Sense Multiple Access / Collision Detection*) é o protocolo usado em redes Ethernet para garantir que apenas um nó de rede está transmitindo na rede fio em qualquer ocasião.

Carrier Sense significa que cada dispositivo Ethernet escuta o cabo Ethernet antes que ele tenta transmitir. Se o dispositivo Ethernet necessita transmitir o mesmo verifica se não existe outro dispositivo está transmitindo, caso exista irá aguardar a conclusão da transmissão para transmitir.

Multiple Access significa que mais de um dispositivo Ethernet pode escutar e esperar para transmitir ao mesmo tempo.

Collision Detection significa que, quando múltiplos dispositivos Ethernet acidentalmente transmitem ao mesmo tempo, os mesmos são capazes de detectar este erro e cessam a transmissão.

Para entender o funcionamento do CSMA/CD considere uma rede Ethernet muito simples, com apenas dois nodos. Cada nó, independentemente, decide enviar um frame Ethernet para o outro nó. Ambos os nodos escutam o cabo Ethernet e verifica que nenhuma portadora está presente. Assim ambos os nodos começam a transmitir simultaneamente,

causando uma colisão. Ambos os nodos detectam a colisão, cessam a transmissão e esperam um tempo aleatório antes de reiniciar a transmissão.

As colisões são normais em uma rede Ethernet. Uma pequena quantidade de colisões é esperada no protocolo. Se muitos nodos estão transmitindo em uma rede Ethernet o número de colisões pode ascender a um nível inaceitável. Isto pode reduzir a quantidade de largura de banda disponível em uma rede Ethernet, porque muito se perdeu na banda de retransmissão. Switches Ethernet conseguem reduzir fortemente as pequenas dificuldades já sentidas com o protocolo CSMA / CD.

3.2 – FF-HSE – Camada de Enlace: MAC *Medium Access Control*

A subcamada MAC definida no padrão IEEE 802.3 (método de acesso CSMA/CD) é utilizada sem nenhuma modificação no FF-HSE. Cada placa de rede existente em um dispositivo conectado à rede possui um endereço MAC único, que é gravado em *hardware* e não pode ser alterado. Esse endereço utiliza 06 bytes como por exemplo: 02608C428197.

Os três primeiros bytes representam o código do fabricante determinado *Organizationally Unique Identifier* (OUI), e os três últimos bytes é definido pelo fabricante e servem para identificar o dispositivo, os fabricantes são responsáveis em controlar estes valores pois não podem existir dois dispositivos com o mesmo endereço MAC. A finalidade dessa distinção é para que o computador seja capaz de identificar outros computadores na rede. Esse endereço é o R.G. da placa e do micro na rede.

Outra função da MAC é controlar o uso do barramento, verificando se o cabo está ocupado ou não. Se o cabo está ocupado o quadro de dados não será enviado, caso contrário os dados serão enviados pela rede. Se durante a transmissão ocorrer uma colisão (transmissões simultâneas pelo mesmo cabo) a MAC é capaz de identificar as máquinas envolvidas, fazendo com que elas esperem tempos diferentes para poderem transmitir novamente.

Quando o pacote chega à esta sub-camada, ele deve receber uma informação sobre o tipo de arquitetura definida para esta rede (Ethernet, ARCNet, FDDI, Token Ring). Cada arquitetura define uma forma de acesso ao cabo, como por exemplo, CSMA/CD para Ethernet

ou passagem de bastão para Token Ring. É de responsabilidade dessa subcamada definir essa informação para o pacote.

3.3 – FF-HSE – Camada de Rede: IP

Internet Protocol (IP) é o protocolo da arquitetura TCP/IP para encaminhamento e roteamento do pacote. O IP realiza o roteamento das informações de um computador para outro, definindo os mecanismos de expedição de pacotes sem conexão.

O *software* de IP executa a função de roteamento, escolhendo um caminho através do qual os dados serão enviados. Incluem um conjunto de regras que envolvem a idéia da expedição de pacotes não confiáveis. Estas regras indicam como os hosts ou gateways poderiam processar os pacotes; como e quando as mensagens de erros poderiam ser geradas; e as condições em que os pacotes podem ser descartados.

O datagrama IP é a unidade básica de dados no nível IP. Um datagrama está dividido em duas áreas, uma área de cabeçalho e outra de dados.

- O cabeçalho contém toda a informação necessária que identificam o conteúdo do datagrama;
- A área de dados é onde está encapsulado o pacote do nível superior, ou seja um pacote TCP ou UDP.

A figura 25 ilustra o formato do datagrama IP.

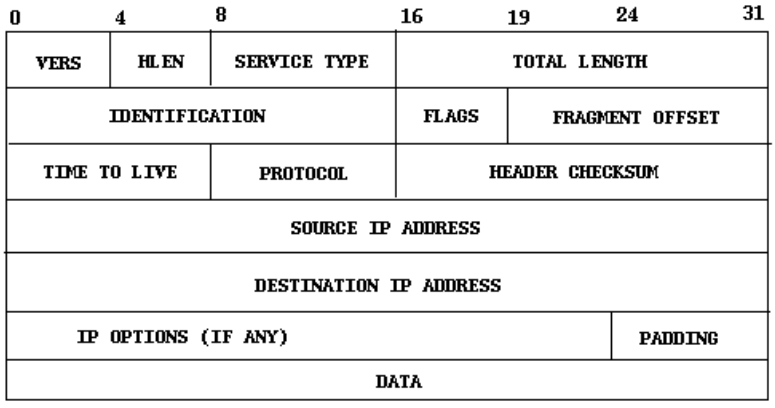

Figura 25: Datagrama IP.
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

3.3.1 – ICMP – *Internet Control Message Protocol*

O protocolo IP provê um serviço de expedição de datagramas sem conexão, não confiável, no qual os datagramas viajam de um *gateway* a outro até alcançar um *gateway* que possa encaminhá-lo ao *host* destino. Assim é necessário um mecanismo que emita informações de controle e de erros quando acontecerem problemas na rede. Alguns dos problemas típicos que podem acontecer são:

- Um *gateway* não pode expedir ou rotear um datagrama
- Um *gateway* detecta uma condição não usual tal como congestionamento.

O protocolo ICMP permite aos *gateways* enviar mensagens de erros ou de controle a outros *gateways* ou *hosts* informando a perda ou descarte de determinado pacote. O ICMP provê comunicação entre os *softwares* de IP numa máquina e o *software* de IP numa outra máquina.

ICMP somente reporta condições de erros à fonte original. A fonte deve relatar os erros aos programas de aplicação individuais e tomar ação para corrigir o problema. Uma das mensagens que o ICMP pode enviar é: *Destination Unreachable*, o qual, por sua vez pode ser dos seguintes tipos:

- *Network Unreachable* (rede não alcançável);
- *Host Unreachable* (host não alcançável);
- *Port Unreachable* (port não alcançável);
- *Destination Host Unknown* (Host destino desconhecido);
- *Destination Network Unknown* (rede destino desconhecida).

3.4 – FF-HSE – Camada de Transporte: TCP/UDP

A função básica do nível de transporte é permitir a comunicação fim-a-fim entre aplicações.

Os seguintes serviços são fornecidos pela camada de transporte:

- Controle de erro;
- Controle de fluxo;

- Seqüencialização;
- Multiplexação do acesso ao nível inter-rede.

O FF-HSE pode utilizar os protocolos TCP ou UDP para prover os serviços da camada de transporte.

- *Transmission Control Protocol (*TCP): Protocolo de Controle de Transmissão. Protocolo da suite TCP/IP que realiza a entrega garantida dos dados seqüenciais.
- *User Datagram Protocol* (UDP): Protocolo semelhante ao TCP que realiza a entrega dos dados mas sem garantia de que eles chegarão ao seu destino.

3.4.1 – TCP – *Transport Control Protocol*

Conforme ilustrado na figura 25, o *Transfer Control Protocol* (TCP) é um protocolo da camada de transporte da arquitetura TCP/IP. O TCP é um protocolo orientado a conexão, o que significa que neste nível serão solucionados todos os problemas de erros que não forem solucionados no nível IP, dado que este último é um protocolo sem conexão.

Alguns dos problemas com os que TCP deve tratar são:

- Pacotes perdidos ou destruídos por erros de transmissão;
- Expedição de pacotes fora de ordem ou duplicados.

O TCP especifica o formato dos pacotes de dados que dois computadores trocam para realizar uma transferência confiável, assim como os procedimentos que os computadores usam para assegurar que os dados cheguem corretamente. Entre estes procedimentos estão:

- Distinguir entre múltiplos destinos numa máquina determinada.
- Fazer recuperação de erros, tais como pacotes perdidos ou duplicados.

Baseando-se no modelo de referência OSI, o protocolo TCP é um protocolo que reside na camada de transporte como mostra a figura 26.

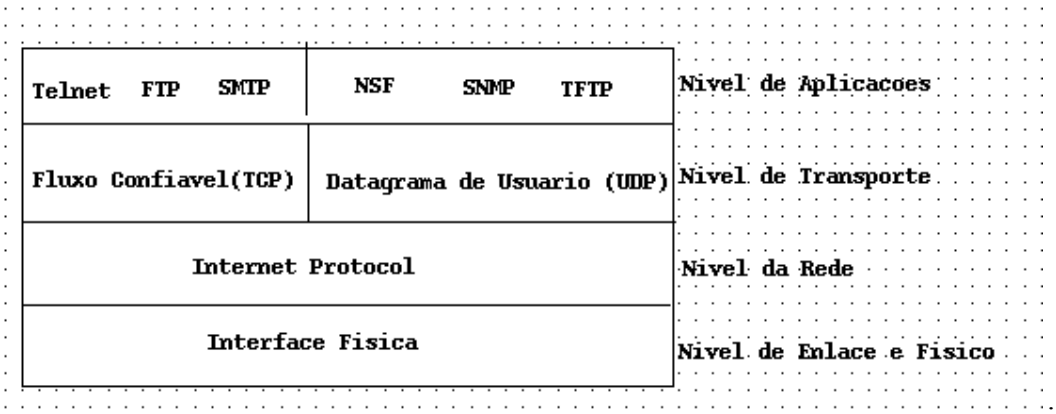

Figura 26: Camadas do modelo TCP/IP.
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

O TCP permite que múltiplos programas de aplicação numa determinada máquina se comuniquem concorrentemente e neste caso o TCP se encarrega de desmultiplexar o tráfego TCP entrante entre os programas de aplicação. Isso é possível pois o TCP usa número de portas para identificar o último destino numa máquina. A cada porta é associado um número inteiro pequeno para identificá-lo.

O TCP foi construído sobre a abstração de CONEXÃO, na qual os objetos a serem identificados são conexões de circuitos virtuais e não portas individuais. As conexões são identificadas por um par de *endpoints*. Uma conexão consiste de um circuito virtual entre dois programas de aplicações, então pode-se assumir que existe um programa de aplicação como a conexão entre os *endpoints*, mas isto não é certo, TCP define um *endpoint* como um par de inteiros (host, port), onde host é o endereço IP para um computador e Port é uma porta TCP nesse computador.

Já que o TCP identifica uma conexão por um par de endpoints, um número de porta pode ser compartilhado por múltiplas conexões na mesma máquina. O TCP vê o fluxo de dados como uma sequência de bytes, que ele divide em segmentos para a transmissão. Usualmente cada segmento viaja através da Internet com um único datagrama IP.

A unidade de transferência entre o *software* TCP de duas máquinas é chamada de segmento. Os segmentos são trocados para estabelecer conexões, transferir dados, enviar aceitações e fechar conexões. Uma aceitação viajando de uma máquina A a B pode ir no mesmo segmento de dados que estão sendo enviados de A a B, embora o reconhecimento refere-se a dados enviados da máquina B a A.

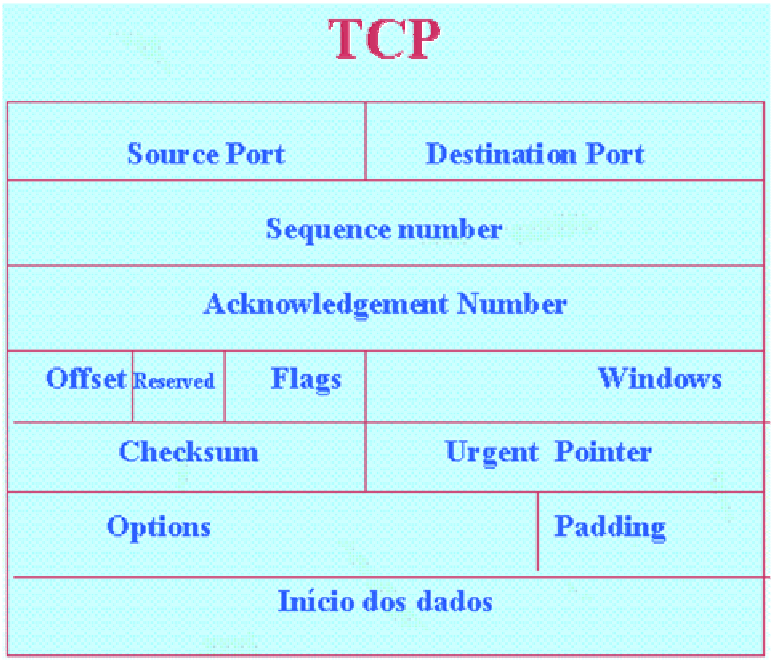

Figura 27: Estrutura do PDU transmitido pelo TCP.
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

Segue as definições dos campos do segmento TCP, conforme ilustrado na figura 27:

- Porta Fonte e Destino: estes campos no cabeçalho TCP contêm os números de portas TCP que identificam os programas de aplicação dos extremos de uma conexão;
- Número de seqüência (32 bits): identifica a posição no fluxo de bytes do segmento enviado pelo transmissor. O número de seqüência refere-se ao fluxo de dados que vai à mesma direção do segmento;
- Número de Reconhecimento (32 bits): este campo identifica a posição do byte mais alto (ou último byte) que a fonte recebeu. O número de reconhecimento refere-se ao fluxo de dados na direção contrária ao segmento. Os

reconhecimentos sempre especificam o número do próximo byte que o receptor espera receber;

- Offset: contém um inteiro que especifica o início da porção de dados do segmento. Este campo é necessário já que o campo Options varia em comprimento dependendo de quais opções tenham sido incluídas. De modo que o tamanho do cabeçalho TCP varia dependendo das opções selecionadas;
- RES: reservado para uso futuro;
- CODE(6 bits): determina o propósito e conteúdo do segmento;
- WINDOW: através deste campo o *software* TCP indica quantos dados ele tem capacidade de receber em seu buffer;
- URGENT POINTER: TCP através deste campo permite que o transmissor especifique que alguns dados são urgentes, isto significa que os dados serão expedidos tão rápido quanto seja possível;
- OPTIONS: o *software* TCP usa este campo para se comunicar com o *software* do outro extremo da conexão;
- CHECKSUM: é usado para verificar a integridade tanto do cabeçalho como dos dados do segmento TCP.

3.4.2 – UDP – *User Datagram Protocol*

O *User Datagram Protocol* (UDP) provê um serviço sem conexão não confiável, usando IP para transportar mensagens entre duas máquinas. Este protocolo, igualmente o TCP, provê um mecanismo que o transmissor usa para distinguir entre múltiplos receptores numa mesma máquina, a diferença entre o TCP e o UDP é que no TCP é estabelecido um canal de comunicação entre as máquinas para iniciar a transmissão enquanto que no UDP as mensagens são simplesmente enviadas e aguarda-se a confirmação do recebimento. Caso o emissor não receba a resposta do receptor o pacote é re-enviado.

Cada datagrama UDP é formado por um cabeçalho UDP e uma área de dados. O formato do cabeçalho UDP está dividido em quatro campos de 16 bits.

Definições dos campos:

- *Source and Destination Ports*: estes campos contêm os números de portas fonte e destino do protocolo UDP. A porta fonte é opcional, quando é usada ela

especifica a porta a qual uma resposta poderia ser enviada, se não é usada contém zeros.

- *Length*: contém um contador de bytes no datagrama UDP. O valor mínimo é oito, sendo este só o comprimento do cabeçalho.
- *Checksum*: Este campo é opcional. Um valor de zero indica que o *checksum* não é computado.

3.5 – FF-HSE – Camada de Aplicação: FDA

A rede HSE suporta todas as funcionalidades das camadas de enlace de dados da especificação H1. Isto teve que ser feito para possibilitar o sincronismo de uma ligação em cascata entre segmentos H1 independentes. Os instrumentos de campo também podem *bypassar* o protocolo H1 e transmitir usando o protocolo HSE diretamente.

Os Agentes HSE de administração de sistema e rede, blocos de função e agentes de acesso a dispositivos de campo *Field Device access Agent* (FDA) residem na camada de aplicação, sobre as camadas de rede e transporte da arquitetura TCP/IP. Vale registrar que a utilização do protocolo Ethernet na especificação do FF HSE tem o objetivo de estender a aplicabilidade do FF até a interligação de qualquer dispositivo que utilize TCP/IP com protocolo Ethernet. Possibilitando a interligação do FF com qualquer outra rede, industrial ou comercial que venha a se consolidar no mercado.

O FDA – *Field Device Access Agent* que tem os seguintes objetivos:

a) Carregar serviços de sistemas de gerenciamento SM – *System Management* sobre UDP e serviços FMS sobre UDP/TCP. Isto permite que dispositivos HSE e H1, dispositivos I/O e dispositivos I/O não FF sejam conectados através de um *linking device* ou *gateway*.
b) Republicar dados H1 de *linking devices* que não suportem *bridge*. Isto permite que *linking devices* sejam construídos com redes H1 independentes.
c) Enviar e receber mensagens de redundância de LAN para permitir a redundância de interfaces em dispositivos HSE. O Agente FDA permite que o sistema de controle funcione em cima do HSE através de *Linking Devices* e habilita o acesso remoto de aplicações aos dispositivos de campo utilizando UDP/TCP.

3.6 – FF-HSE – Aplicações do Usuário

O *User Application* (API) existente no FF-HSE, assim como o API do FF-H1 é especificada pela norma IEC61804. O API do FF-HSE utiliza e suporta os mesmos blocos e protocolos do FF-H1.

3.7 – Endereçamento de Dispositivos FF-HSE

Além do protocolo SNMP, os dispositivos HSE utilizam os protocolos DHCP *Dynamic Host Configuration Protocol*, UDP *User Datagram Protocol*, IP *Internet Protocol* e a funcionalidade de gerenciamento de sistema para o endereçamento dos dispositivos.

Inicialmente o dispositivo FF solicita um endereço IP para o Servidor DHCP. O servidor disponibiliza um endereço IP válido. O dispositivo então envia uma mensagem para que o sistema de gerenciamento consiga identificar qual elemento recebeu o endereço lógico e designa um endereço físico para o dispositivo. Assim que os endereços lógico e físico estão definidos o dispositivo pode começar a operar coletando e fornecendo informações.

Neste capítulo foi exposto que nos dispositivos FF-HSE foi implementada a arquitetura TCP/IP utilizando o protocolo Ethernet como especificação das camadas Física e de Enlace. Na camada de aplicação dos dispositivos foi implementado o protocolo FDA que é responsável por encapsular os dados das aplicações de usuário e transmiti-los através da rede IP. O FDA também é o responsável pelo acesso aos dispositivos FF-H1 e realiza a conversão das informações FMS para o formato IP.

No capítulo a seguir será exposta as principais características e o princípio de funcionamento do protocolo SNMP da Arquitetura TCP/IP e que serve para o gerenciamento de dispositivos.

4 – PROTOCOLO SNMP

O *Simple Network Management Protocol* (SNMP) [STALLINGS, 1999] é um protocolo de gerência definido no nível de aplicação na pilha de protocolos TCP/IP [EVANS & WASHBURN, 1996], vide figura 28. O gerenciamento da rede através do SNMP permite o acompanhamento simples e fácil os estado da rede, em tempo real, podendo ser utilizado para o gerenciamento de diversos tipos de sistemas.

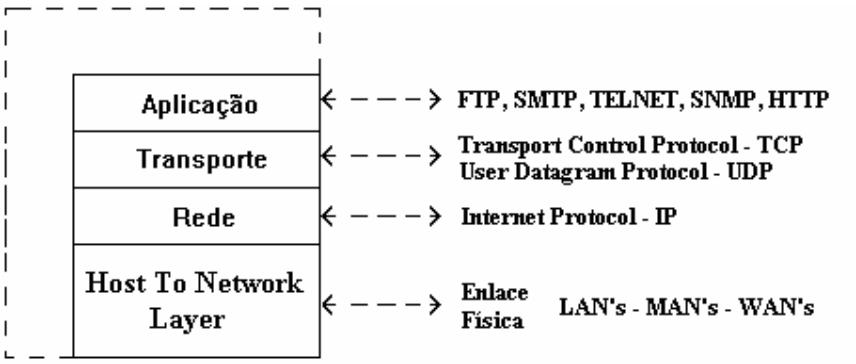

Figura 287 – O SNMP é um protocolo da camada de aplicação da arquitetura TCP/IP.
Fonte: Apontamentos de aula do Professor Msc Sérgio F. Brito

Os dados são obtidos através de requisições de um gerente a um ou mais agentes utilizando os serviços do protocolo de transporte *User Datagram Protocol* (UDP) [TANENBAUM, 2007] para enviar e receber mensagens através da rede. Os comandos do SNMP são limitados e baseados no mecanismo busca/alteração onde estão disponíveis as operações: de alteração de um valor de um objeto; de obtenção dos valores de um objeto e suas variações (Figura 29).

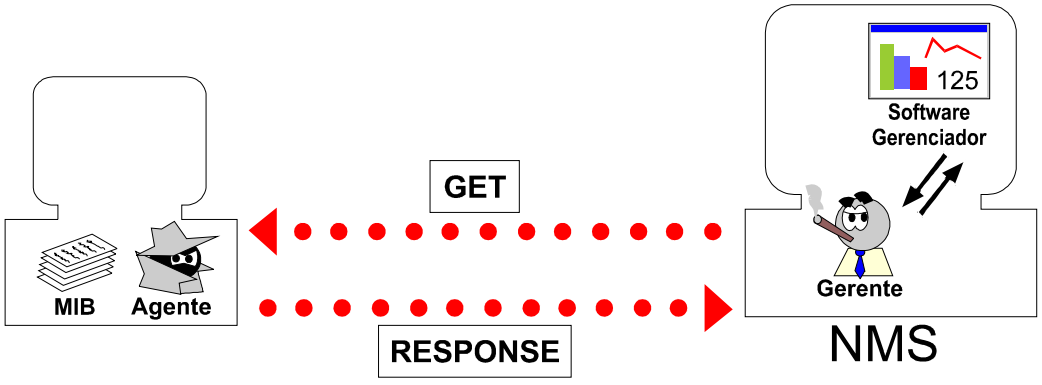

Figura 29 – Todas as operações do SNMP são solicitadas/requisitadas pelo Gerente SNMP.
Fonte: Monografia – Usando SNMP para Gerência de Rede, Araújo&Pinheiro&Pondé, 1995.

Observe que um Gerente SNMP tem a capacidade de monitorar diversos dispositivos que contém agentes SNMP. Basicamente o Gerente SNMP realiza a operação de *polling* onde em intervalos de tempo regular o Gerente questiona ao Agente SNMP se houve alguma alteração no status do dispositivo ou se foi gerado algum alarme desde a última comunicação (Figura 30).

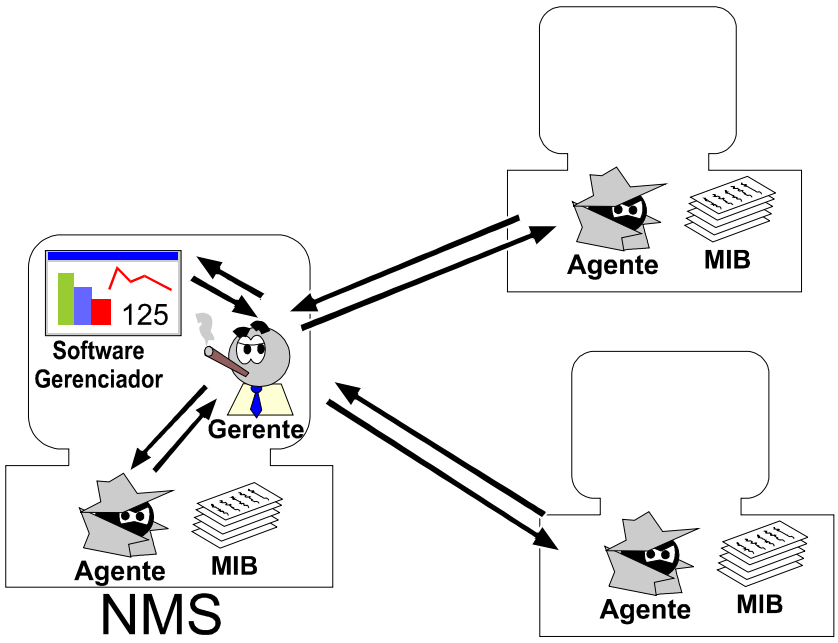

Figura 30 – Todas as operações do SNMP são solicitadas/requisitadas pelo Gerente SNMP.
Fonte: Monografia – Usando SNMP para Gerência de Rede, Araújo&Pinheiro&Pondé, 1995.

O SNMP pode ser utilizado para a supervisão de qualquer dispositivo da Rede TCP/IP que possua este protocolo instalado. Dependendo do objetivo do sistema de gerenciamento é possível que um Gerente SNMP questione o status de um equipamento que está localizado em uma outra rede, desde que as duas redes possuam interligação.

Uma característica interessante do SNMP é que o Agente SNMP pode responder as requisições de diversos Gerentes SNMP. Como exemplo, a figura 31 ilustra uma rede corporativa que possui três sistemas de gerenciamento de dispositivos independentes.

Na situação proposta na figura 31 existem os sistemas de gerenciamento de dispositivos locais e remotos. Neste exemplo cada sistema de gerenciamento é executado em servidores independentes e desta forma possui três gerentes SNMP funcionando ao mesmo tempo. Como todas as mensagens SNMP possuem no cabeçalho os endereços do Gerente

solicitante e do Agente solicitado as mensagens que não pertencem a um Gerente SNMP são descartadas.

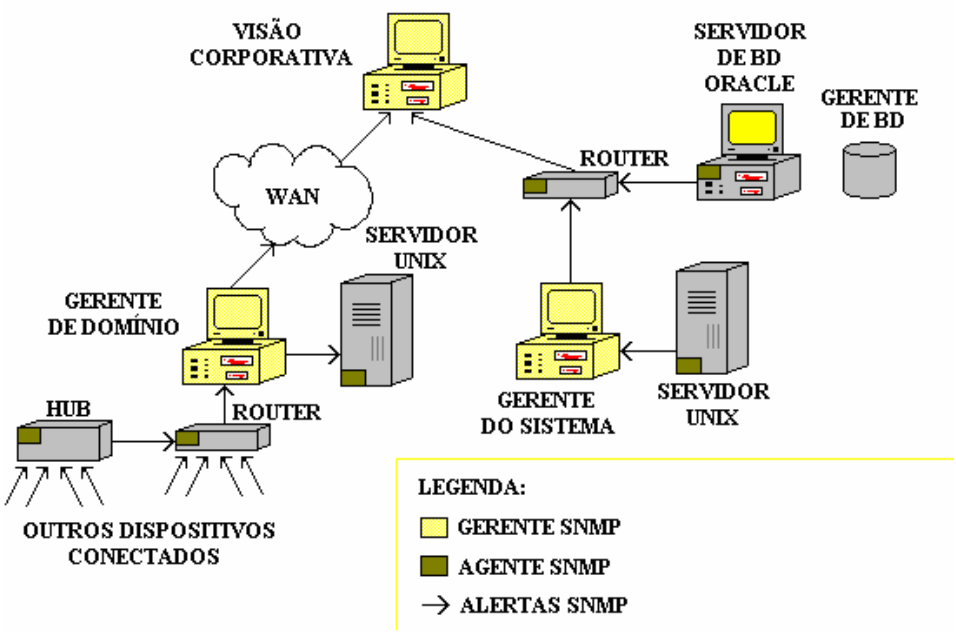

Figura 31 – Rede corporativa com diversos Gerentes SNMP e agentes SNMP.
Fonte: Apontamentos de aula do Professor Msc. Sérgio F. Brito

A utilização de um número limitado de operações, baseadas em busca/alteração, torna o protocolo de fácil implementação, estável, simples e flexível. O que reduz o tráfego de mensagens de gerenciamento através de rede e permite a introdução de novas características.

A simplicidade do SNMP permite que as implementações dos agentes sejam compactas e eficientes, de forma que os dispositivos utilizam o maior parte de sua memória e recursos no processamento de suas funções primárias e não para processar as solicitações dos administradores de rede. Independente de sua simplicidade, o SNMP é um protocolo robusto que se desenvolveu com muito êxito apesar das condições adversas da rede sendo adotado como padrão para redes TCP/IP desde 1989. [CISCO SYSTEMS INC, 2003].

Como as redes, na sua maioria, têm seus dispositivos de diferentes fabricantes é necessário que a natureza das informações mantidas por todos os dispositivos deve ser rigidamente especificada. Por isso o SNMP descreve com riqueza de detalhes as informações exatas que cada agente deve manter e o formato a ser aplicado a essas informações. A maior parte do modelo SNMP se refere à definição de que agente deverá acompanhar qual informação e o modo como essa informação será comunicada.

Existem três versões de protocolo SNMP: SNMP versão 1 (SNMPv1), SNMP versão 2 (SNMPv2) e SNMP versão 3 (SNMPv3). Todas as versões têm várias características em comum, mas SNMPv2 oferece melhoramentos em relação ao SNMPv1 e o SNMPv3 possuem melhoramentos em relação ao SNMPv2.

O protocolo SNMPv2 acrescenta alguns comandos em relação ao protocolo SNMPv1, enquanto que o SNMPv3 além dos comandos adicionais provê acesso seguro aos dispositivos através da combinação dos recursos de e autenticação com e encriptografia dos pacotes que trafegam na rede.

Um detalhe importante é que os agentes e os sistemas de gerenciamento necessitam utilizar a mesma versão de SNMP. Por esta razão este trabalho abordará apenas a versão 1 do SNMP pois necessariamente é aceita por todos os dispositivos que suportam o protocolo SNMP.

4.1 – Componentes Básicos do SNMP

Uma rede gerenciada através de SNMP consiste de três componentes fundamentais: dispositivos gerenciados, agentes e sistemas de gerenciamento de rede, os *Network Management Systems* (NMSs) ou gerentes de rede (Figura 32).

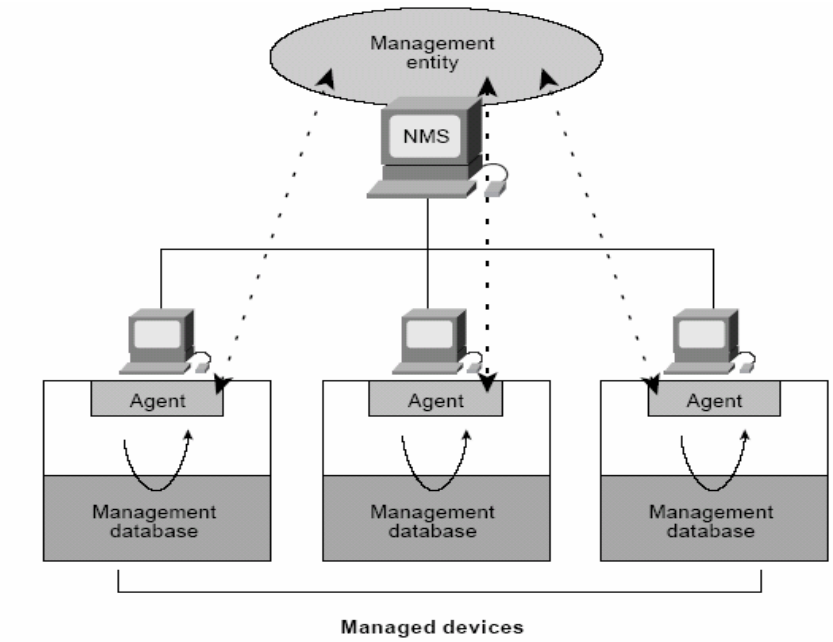

Figura 32 – Relacionamento entre o Sistema de Gerenciamento e os dispositivos
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

Cada máquina gerenciada é vista como um conjunto de variáveis que representam informações referentes ao seu estado. Estas informações são disponibilizadas para os NMSs podendo ser consultadas e modificadas por eles. Cada dispositivo gerenciado por SNMP deve possuir um agente e uma base de informações de gerenciamento.

Um agente é um módulo de *software* de administração de rede que reside no dispositivo administrado. Um agente tem conhecimento local de informações de gerenciamento e traduz esta informação para um formato compatível com o SNMP.

Um NMS executa aplicações que monitoram e controlam os dispositivos gerenciados. Os NMSs provêem a empilhamento dos recursos de processamento e memória requeridos para a administração da rede. Podem existir mais de uma NMS em uma rede gerenciada.

As informações de gerenciamento, denominadas objetos, presentes em cada elemento gerenciado (agente), descrevem a configuração, o estado, as estatísticas e controlam as ações do dispositivo gerenciado. O conjunto destes dados é definido através de especificações denominadas MIB *Management Information Base*. As MIBs são descritas utilizando a notação ASN.1 *Abstract Syntax Notation One*, que especifica como as informações serão codificadas. [STALLINGS, 1999].

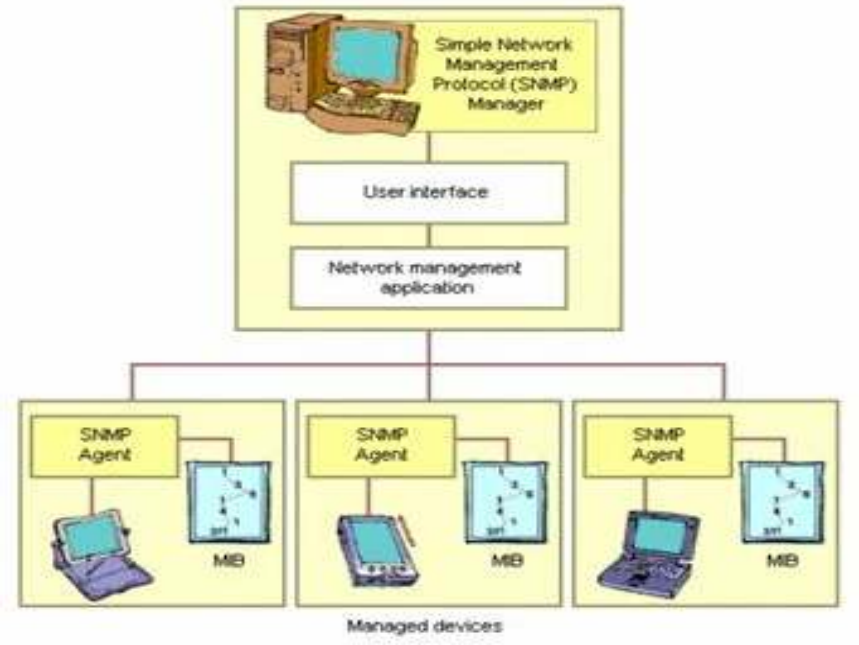

Figura 33: Principais elementos da gerência SNMP
Fonte: Introduccion SNMP, Vincenzo Medillo, 2003.

A figura 33 ilustra o relacionamento entre os dispositivos gerenciados e o sistema de gerenciamento. Destacando a existência da MIB e dos Agentes no dispositivo e a existência do NMS ou gerente na estação de gerenciamento [MEDILLO, 2003].

4.1.1 – Agente

Os agentes são *softwares* residentes no equipamento a ser gerenciado que efetuam duas ações básicas: inspeção e modificação de variáveis na MIB. A inspeção consiste em examinar os valores dos: contadores, apontadores e outros parâmetros, à medida que modificar consiste em alterar os valores das variáveis que constam na MIB.

Um agente pode suportar um ou mais módulos MIB, incluindo as MIBs padrões, especificadas através de RFCs, e as MIBs proprietárias, definidas pelos fabricantes dos equipamentos para seus produtos específicos.

De uma forma geral, o agente responde as solicitações do gerente, exceto na ocorrência de eventos pré-determinados em que o agente envia uma mensagem chamada TRAP para o gerente.

4.1.2 – Gerente

O sistema de gerenciamento, chamado de Gerente ou NMS, é um *software* residente na estação de gerenciamento que enviam comandos aos agentes para realizarem consultas e alterações de variáveis na MIB e interpretar as mensagens TRAPs recebidas dos agentes. Usualmente apresentam uma interface gráfica ao usuário, apresentando um mapa da rede.

Quando uma solicitação chega ao gerente SNMP, ele transforma esta solicitação em um comando SNMP, que através dos trâmites para transferência de mensagens atingirá um agente que por sua vez pode ou não enviar uma resposta ao gerente a depender do comando recebido.

Observe que inicialmente, quando é iniciado o *software* de gerenciamento, não se conhece nada a respeito da rede, mas aos poucos as informações vão chegando ao gerente através de TRAPs, que a partir daí realiza *polling* para manter a comunicação com os agentes,

possibilitando ao *software* de gerenciamento mapear, monitorar e controlar a rede. [STALLINGS, 2001].

Vale salientar que o fato de uma máquina possuir o gerente instalado não a exclui de possuir o agente, pois as máquinas que auxiliam o gerenciamento também devem ser monitoradas de forma a garantir o bom funcionamento da rede.

4.2 – MIB Management Information Base

Os recursos reais do sistema que necessitem ser monitorados pela rede são modelados em estruturas de dados chamadas de objetos gerenciados, que podem ser apenas de leitura ou de leitura e escrita, sendo que cada leitura reflete o estado real do recurso e cada alteração modifica o próprio recurso.

Podemos descrever a MIB como um conjunto de objetos gerenciados que busca abranger todas as informações necessárias para o gerenciamento do recurso.

O *Request For Comment* (RFC) 1066 apresentou a primeira versão da MIB que explicou e definiu a base de informação necessária para monitorar e controlar redes baseadas no pilha de protocolos TCP/IP. Posteriormente o RFC 1213 propôs três versões de MIBs: MIB II, MIB Experimental, MIB Privada.

A MIB II é considerada a evolução da MIB fornecendo informações gerais de gerenciamento sobre um determinado equipamento, tais como número de pacotes transmitidos, estado da interface e outros.

A MIB Experimental é aquela em que seus componentes (objetos) estão em fase desenvolvimento e teste, em geral, eles fornecem características mais especificas sobre a tecnologia dos meios de transmissão e equipamentos empregados.

A MIB Privada é aquela em que seus componentes fornecem informações especificas dos equipamentos gerenciados, como configuração, taxa de colisões, através da qual é possível reinicializar, habilitar ou desabilitar portas e recursos dos dispositivos.

As regras de construção da MIB são descritas através de *Structure of Management Information* (SMI). A estrutura de informações de gerencia SMI é um conjunto de documentos que definem: forma de identificação e agrupamento das informações; sintaxes e tipos de dados permitidos.

Os objetos de uma MIB são especificadas de acordo com o *Abstract Syntax Notation One* (ASN.1). A notação sintática abstrata é uma forma de descrição de dados com o objetivo de compatibilizar a troca de informações entre sistemas diferentes independente do equipamento em que está sendo implementado.

Para cada objeto são definidos: nome; identificador; sintaxe; definição e acesso.

- O nome do objeto é um texto com poucos caracteres;
- O identificador do objeto é formado por números separados por pontos;
- A sintaxe do objeto descreve o formato, ou o valor, da informação podendo ser: número inteiro, endereço de rede, contador, medida, intervalo de tempo e outros;
- A definição é uma descrição textual do objeto;
- O acesso define o tipo de controle que se pode ter sobre o objeto, podendo ser somente leitura, leitura e escrita ou não acessível.

A MIB de qualquer equipamento está dividida em duas partes. A primeira delas se chama MIB padrão ou SMI. A segunda chama-se MIB Estendida e contém informações específicas a respeito do equipamento em questão.

A SMI é comum a qualquer equipamento gerenciável e obedece a um padrão definido em RFCs da *Internet Activities Board* (IAB) e da IETF *Internet Engineering Task Force* e utilizados pela *International Standards Organization* (ISO) para definição de especificações técnicas garantindo que independentemente do tipo, marca e modelo do equipamento ele possa ser monitorado e gerenciado, mesmo que sem muitos recursos, pelo SNMP.

A estrutura que provê suporte a definição de uma MIB está dividida nos seguintes quatro grupos principais de objetos:

- *directory*: é um grupo reservado para uso com um futuro memorando que discutirá como o diretório OSI pode ser usado na Internet.
- *mgmt*: usada para objetos definidos em documentos aprovados pelo IAB. Esta sub-árvore contém, por exemplo, a MIB-II, que faz parte das MIBs que podem ser consultadas no equipamento;
- *experimental*: utilizada para identificar objetos usados em experimentos da Internet;
- *private*: utilizada para identificar objetos privados registrados pelo IAB. A sub-árvore private possui apenas uma sub-árvore, denominada *enterprises*, identificada pelo valor (1). Nesta sub-árvore, cada empresa (fabricante) pode ter um número atribuído para ela, que é seu enterprise *number*. Este número designa a raiz da sub-árvore específica para a empresa, onde são inseridas as MIBs proprietárias desta empresa.

Estes objetos obedecem a uma estrutura de árvore onde para se ter acesso a um determinado ramo desta árvore, deve-se nomeá-lo desde o nome da raiz, passando por todos os ramos subseqüentes, e separando-os por um ponto, até se chegar ao ramo desejado.

A figura 34 ilustra a arvore hierárquica definida pela ISO para representar a estrutura hierárquica da MIB, ilustrando identificador e nome de cada objeto.

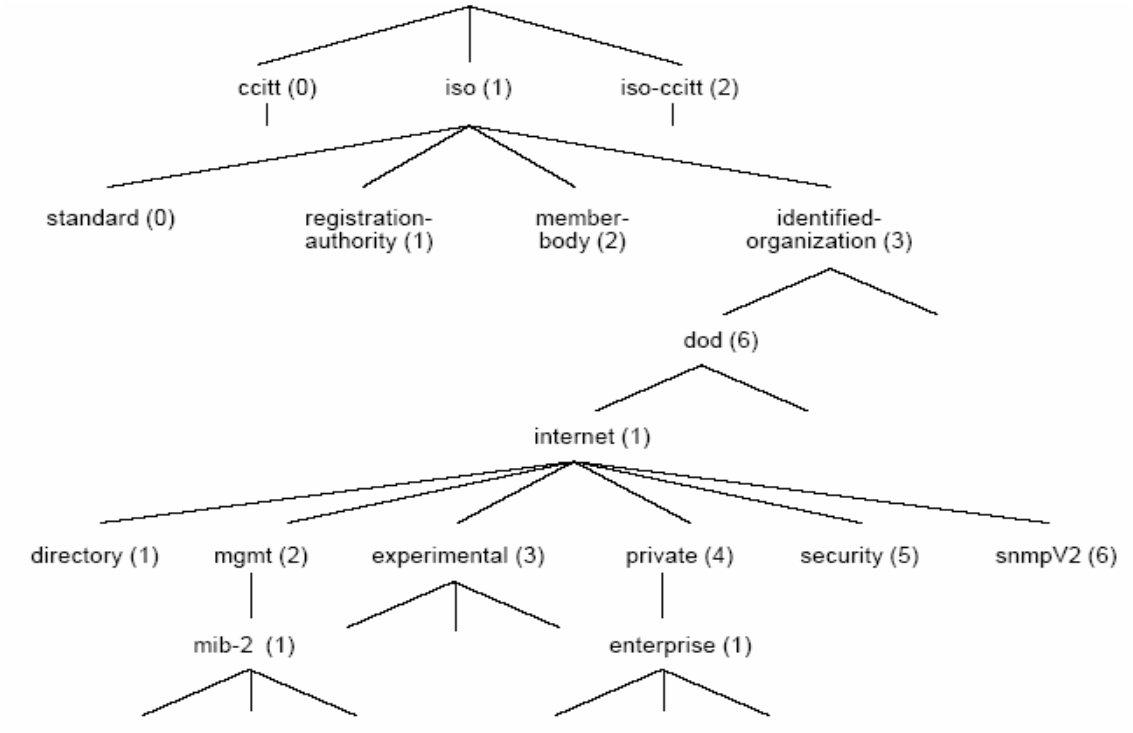

Figura 34 – Arvore hierárquica definida pela ISO para representar a estrutura da MIB
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

O nó raiz da arvore não possui rótulo e possui três subníveis: o nó 0 que é administrado pela *Consultative Commite for International Telegraph and Telephone* (CCIT); o nó 1 é administrado pela ISO; o nó 2 é administrado conjuntamente pela ISO e pela CCIT. Sob o nó ISO fica o nó que pode ser utilizado por outras instituições. Assim teríamos o nó org[3], abaixo deste o nó do Departamento de Defesa do EUA (*Departament Of Defense* – DoD [6]). Abaixo do nó dod[6] existe o nó internet que é administrado pela IAB. Abaixo do nó da IAB temos diversos nós, onde se destaca os nós: *management* [2], *experimental* [3] e *private*[4].

Dentro do grupo *mgmt* (*management*) está localizado os sub-ramos MIB e MIB-2 que por sua vez são considerados os mais importantes, pois dentro deles existe toda uma outra ramificação de informações que será vista a seguir.

As MIBs proprietárias de uma empresa são publicadas e distribuídas por ela, que é responsável pelo seu conteúdo. A estrutura de cada sub-árvore de uma empresa é organizada por ela segundo as suas necessidades. O IANA *Internet Assigned Number Authority* é o encarregado de fornecer o enterprise number de uma empresa

4.3 – Comandos do SNMP

No SNMPv1, os dispositivos gerenciados são controlados utilizando duas operações básicas (SET e GET) e suas derivações (GET-NEXT e TRAP). O funcionamento do SNMP através da utilização dos comandos SET e GET são ilustrados na figura 35.

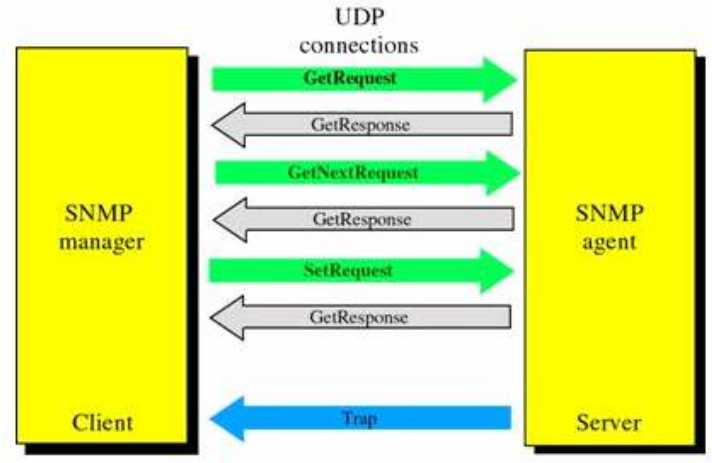

Figura 35 – Funcionamento dos comandos SET e GET no SNMP.
Fonte: Introduccion SNMP, Vincenzo Medillo, 2003.

A seguir uma descrição simplificada dos comandos utilizados no SNMP:

- A operação SET é utilizada pelo para alterar o valor da variável do dispositivo. O gerente solicita que o agente faça a alteração do valor de uma variável determinada;
- A operação GET é utilizada pelo gerente para ler o valor da variável. O gerente solicita que o agente informe o valor de determinada variável;
- A operação GET-NEXT é utilizada pelo gerente para ler o valor da próxima variável. O gerente fornece o nome de uma variável e o agente informa o nome e o valor da próxima variável. Também é utilizado para obter nomes e valores de variáveis de uma tabela de tamanho desconhecido;
- A operação TRAP é utilizada pelo agente para comunicar ao gerente a ocorrência de algum evento previamente determinado. Existem sete tipos básicos de TRAP:
    1. *ColdStart* – a entidade que a envia foi reinicializada e a configuração do agente ou a implementação pode ter sido alterada;

2. *WarmStart* – a entidade que a envia foi reinicializada, mas a configuração do agente e a implementação não foram alteradas;
3. *LinkDown* – o enlace de comunicação foi interrompido;
4. *LinkUp* – o enlace de comunicação foi estabelecido;
5. *AuthenticatinFailure*- o agente SNMP recebeu uma mensagem SNMP do gerente que não foi autenticada;
6. *EgpNeighborLoss* – Um par EGS parou;
7. *EnterpriseSpecific* – indica a ocorrência de uma operação TRAP não básica.

4.4 – ASN.1 Abstract Syntax Notation 1

Para tornar a comunicação entre os equipamentos de diferentes fornecedores possível, é necessário que os objetos sejam definidos de uma forma padronizada e neutra. Além disso, uma forma padronizada é necessária para que os objetos sejam codificados para a transferência na rede. Por essa razão é utilizada uma linguagem de definição de objetos padronizada, juntamente com as regras de codificação.

A linguagem utilizada pelo SNMP foi tirada da OSI, a ASN.1, uma linguagem que comporta padrões programados para comunicação de dados, pois além de ser padronizada, determina uma codificação de bits na conexão física de forma que uma estação de gerenciamento de complemento 2 com 32 bits possa trocar informações sem qualquer ambigüidade com um agente contido em uma CPU de complemento dois com 16 bits.

O *Abstract Syntax Notation 1* (ASN.1) é uma notação formal utilizada para descrever os dados transmitidos por protocolos de telecomunicações, independentemente da linguagem em que foi implementada os protocolos ou da representação física destes dados. O ASN.1 foi proposto como padrão internacional em 1984, sendo descrito pelos padrões ISO 8824 e ITU-TX 680 a 683.

A notação oferece um certo número de tipos básicos pré-definidos, como inteiros, cadeias de caracteres, valores lógicos, etc. e possibilita definir tipos compostos como vetores, listas, etc.

Uma das principais razões do êxito do ASN.1 é que sua notação está associada como vários formatos de codificação padronizados. Estes formatos descrevem como os valores definidos em ASN.1 devem ser codificados para transmissão, independentemente da máquina ou linguagem de programação que está sendo utilizada. Conhecendo os vários formatos do ASN.1 sempre é possível encontrar um que se adapte a cada aplicação o que se reflete em uma comunicação mais rápida o que consome menos recursos da rede.

O SMI, do SNMPv1, especifica que todos os objetos gerenciados tenham três tipos de dados ASN.1 *name*, *syntax* e *encoding*. O *name* é utilizado como um identificador do objeto (Object ID), a syntax define o tipo de dados e o *encoding* descreve como a informação associado aos objetos gerenciados serão formatada para ser transmitida através da rede.

4.5 – Mensagens do SNMP

Uma mensagem SNMP deve definir o servidor do qual vai se obter ou alterar os atributos de um objeto, e que será o responsável pela conversão das operações requisitadas sobre a MIB. Após a analise da mensagem o servidor deve enviar a resposta da operação ao cliente que a solicitou [STALLINGS, 1999].

As mensagens SNMP não possuem campos fixos e são construídas de trás para frente possuindo três partes principais: *version*, *community* e SNMP PDU, ver figura 31.

- *Version* – contem a versão do SNMP. Tanto o gerente quanto o agente devem utilizar a mesma versão, pois mensagens com versões diferentes são descartadas sem tratamento;
- *Community* – identifica a comunidade e permitem o acesso dos gerentes as MIBs;
- SNMP PDU – parte de dados, constituída ou por um pedido ou por uma resposta a um pedido.

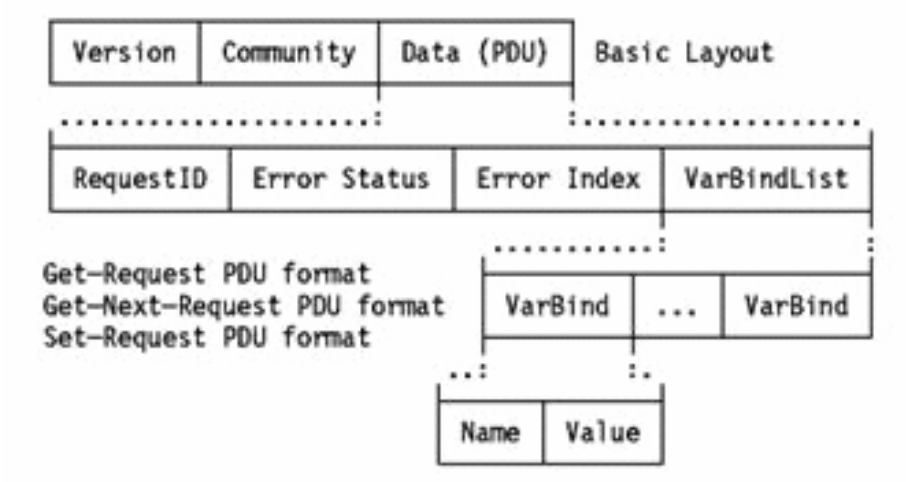

Figura 36: Formato da Mensagem SNMP *Version* + *Community* + Data (PDU)
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

A figura 36 ilustra a formação do SNMP PDU para os comandos *GetRequest*, *GetResponse* e *SetRequest* com os seguintes campos:

- PDU *type* – Especifica o tipo de PDU transmitido;
- Request ID – Associa as solicitações SNMP com as respostas às mesmas;
- *Error Status* – indica um número que sintetiza erro ocorrido. Apenas em operações de resposta este campo possui um valor válido, nas operações de solicitação este campo possui o valor zero;
- *Error index* – Associa o erro a instancia de um objeto particular. Apenas em operações de resposta este campo possui um valor válido, nas operações de solicitação este campo possui o valor zero;
- *Variable bindings* - é o campo de dados do PDU SNMPv1 onde cada variável associada a um objeto e a um valor é armazenada para ser transmitido.

A composição da mensagem TRAP possui um PDU especifico, conforme ilustrado na figura 37.

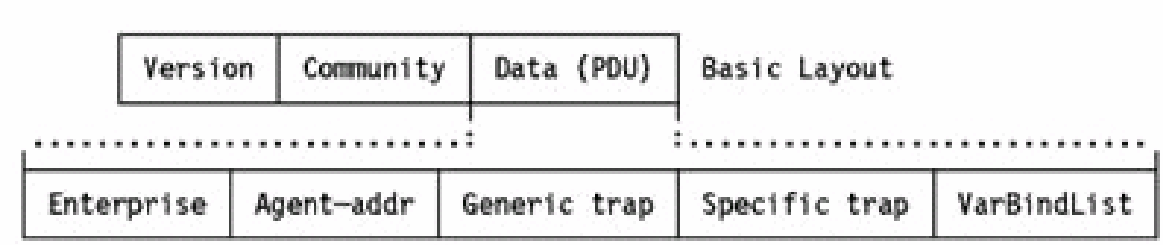

Figura 37 – Formação da PDU de uma mensagem TRAP.
Fonte: Internetworking Technologies Handbook, Cisco Systems, 2003.

A figura 37 ilustra a formação da PDU de uma mensagem TRAP, possuindo os seguintes campos:

- *Enterprise* – Identifica o tipo de objeto gerenciado que gerou o TRAP;
- *Agent address* – Identifica o endereço do objeto gerenciado que gerou o trap;
- *Generic trap type* – Indica um de vários tipos de TRAPs genéricos;
- *Specific trap code* – Indica um de um dos códigos de TRAPs específicos;
- *Time stamp* – Indica o tempo decorrido entre a última reinicialização e a geração do TRAP;
- *Variable bindings* – Indica o valor atual de cada variável que está *linkada* com um objeto particular.

5 – GERENCIAMENTO DE DISPOSITIVOS FF

O SNMP é um protocolo, da arquitetura TCP/IP, bastante simples e eficaz para o gerenciamento de dispositivos. O Funcionamento do SNMP é baseado na troca de informações existentes em uma estrutura fixa denominada de MIB. Nas MIBs existem os objetos que são as representações lógicas das características do dispositivo.

Os dispositivos FF H1 possuem MIBs para gerenciamento de Rede e para gerenciamento do dispositivo. Os dados presentes nestas MIBs são acessíveis através dos VCRs sendo que a comunicação cliente-servidor é realizada através do formato FMS. Assim como o SNMP, as mensagens FMS são transmitidas utilizando o ASN.1.

O comportamento da rede é gerenciado através dos objetos da *Network Management Information Base* (NMIB) e o comportamento do sistema é gerenciado através dos objetos da *System Management Information Base* (SMIB) (Figura 38).

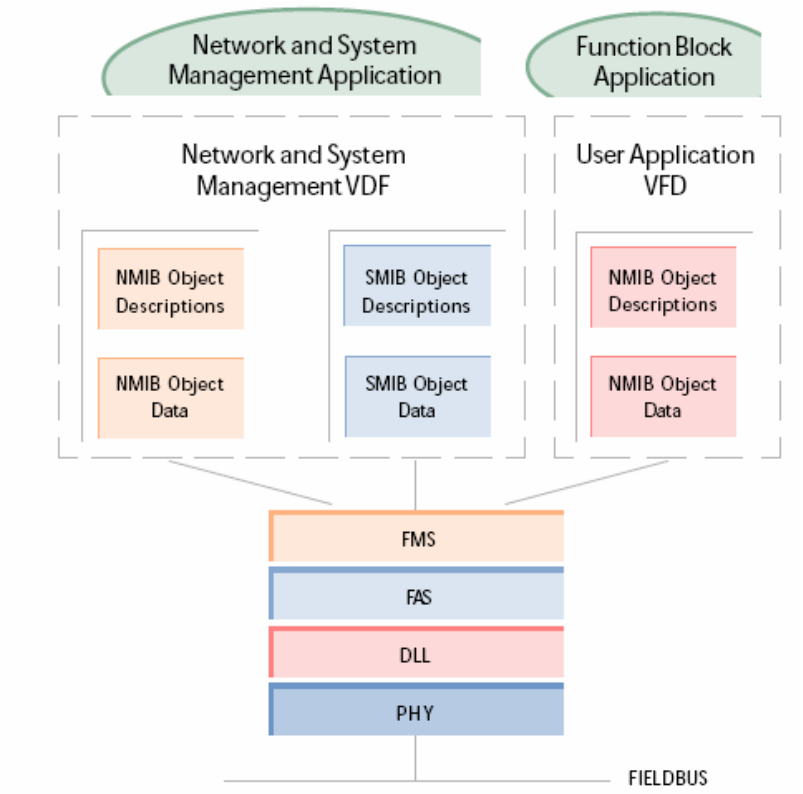

Figura 38 – Todos os dispositivos FF possuem SMIB e NMIB que definem o seu comportamento.
Fonte: Fieldbus Tutorial , SMAR, 2001.

O mesmo VFD usado para administração de rede também é utilizado para administração do sistema. Estes VFDs provêem acesso para as informações da NMIB e da SMIB. Os dados da NMIB incluem VCRs, variáveis dinâmicas, estatísticas, e agendamento do LAS (caso se trate de dispositivo mestre). Os dados da SMIB incluem informações de endereço e identificação do dispositivo e agendamento para execução de blocos de funções.

Os dispositivos FF-HSE possuem muitos protocolos da arquitetura TCP/IP, incluindo o SNMP. Estes dispositivos possibilitam o acesso direto as suas MIBs através do modelo cliente-servidor do SNMP. Os dispositivos HSE com a funcionalidade de *Gateway* conseguem acessar os VCRs de gerenciamento dos dispositivos FF-H1 e converter suas informações do formato FMS para o formato SNMP, possibilitando que equipamentos FF-H1 possam ser monitorados/gerenciados através do SNMP.

O gerenciamento de rede HSE permite que *hosts* HSE gerenciem os seus dispositivos associados utilizando a interface HSE. As seguintes capacidades são providas pelo gerenciamento de rede HSE:

a) Configuração de *Bridges* H1 para encaminhar e republicar dados entre interfaces H1;
b) Carregamento da *HSE Session List* ou entradas únicas da lista. Um *HSE Session Endpoint* representa um canal de comunicação lógica entre dois ou mais dispositivos HSE;
c) Carregamento da lista HSE VCR ou dos valores únicos desta lista. Um HSE VCR é uma relação de comunicação usado para acessar VFDs através do HSE;
d) Monitoramento do desempenho através da estatística de *Session Endpoints*, HSE VCRs e Bridges H1;
e) Monitoração de detecção de falhas.

Os dispositivos FF podem ser identificados através de três identificadores:

- *Device Identifier* (ID): um texto curto especificado pelo fabricante que nunca muda;
- *Physical Device* (PD) *Tag*: Um nome individual definido pelo usuário e que não pode ser utilizado em outros segmentos. Usualmente é usado para identificar um dispositivo de acordo com as aplicações específicas da planta;

- (*Physical*) *Node Address*: Um número de 8 bits, sem igual em um segmento de FF nomeado pelo usuário por configuração de rede.

Device ID é um valor único atribuído pelo fabricante a cada dispositivo de forma a não haver outro dispositivo no mundo com o mesmo *Device ID*. Este valor é gravado pelos fabricantes e não podem ser modificados pelos usuários, ou seja, nunca muda.

PD Tag é nomeada pelo usuário para identificar o dispositivo usado na planta. É um campo de texto com 32 caracteres utilizado para identificar o dispositivo. É muito comum, quando ocorre a substituição de um dispositivo obsoleto ou quebrado, utilizar o mesmo *PD Tag* no novo dispositivo.

As identificações *Device ID* e *PD Tag* são muito grandes (32 bytes) e por isso são inadequadas para a comunicação na rede de instrumentação H1, que possui taxa de transmissão limitada a 31,25kbps. Assim existe a possibilidade de utilizar o endereço físico (*Node Address*) do dispositivo com apenas 8 bits para a comunicação com o dispositivo.

São providos serviços para correlacionar estes três identificadores. Por exemplo, em um transmissor de pressão está gravado o *Device ID* 59454300031999DEC22001102344, configurado o *PD Tag* FI1001 e o endereço físico (*Node Address*) 0xF5.

O Agente do Sistema de gerenciamento do dispositivo de campo responde as solicitações do *System Management Kernel Protocol* (SMKP) acerca da configuração do dispositivo. Suas funcionalidades são:

- Saber informações sobre o dispositivo de um endereço específico, incluindo *Device ID*, fabricante, nome e tipo do dispositivo;
- Definir ou apagar o endereço de um determinado dispositivo de acordo com o *Device ID*
- Definir ou apagar o *PD Tag* do dispositivo, e
- Localizar o dispositivo através do *PD Tag*.

Até mesmo quando o endereço físico de um dispositivo é limpo, o mesmo deve conseguir comunicar-se. Para este propósito, uma gama de endereço especial (0xF8 a 0xFB) está preparada para que um dispositivo sem endereço possa utilizar um endereço desta faixa para comunicar-se com outros dispositivos.

O agente FDA permite que os sistemas de controle operem sobre HSE e através de *Linking Devices* permite que aplicações remotas acessem qualquer tipo através de TCP/UDP utilizando uma única interface.

O Gerenciamento do sistema HSE é a atividade que integra dispositivos de uma rede HSE dentro de um sistema de comunicação que suporta as seguintes funções:

a) Cada dispositivo tem uma identidade única permanente e um nome específico configurado no sistema;
b) Os dispositivos mantém informações de controle de versão;
c) Os dispositivos respondem as requisições de localização de objetos, incluindo as do próprio dispositivo;
d) Tempo é distribuído para todos os dispositivos da rede;
e) Blocos de funções agendados são usados para iniciar outros blocos de funções;
f) Os dispositivos são adicionados e removidos da rede sem afetar os outros dispositivos.

Embora os dispositivos FF-H1 não possuam o protocolo SNMP, os mesmos tem todas as suas funcionalidades monitoradas e controladas por MIBs. Estas MIBs podem ser acessadas por FDAs através das VCRs correspondente e consequentemente as informações das MIBs dos dispositivos FF-H1 podem ser acessadas remotamente através do SNMP utilizando os dispositivos FF-HSE como *gateways*.

6 – CONSIDERAÇÕES FINAIS

Os Sistemas de Integração da Manufatura são realidade nas indústrias modernas. A interligação de forma rápida, confiável e robusta é imprescindível para muitos processos com alto nível de critícidade. O padrão FF foi criado com o intuito de definir especificações e normas diversas de forma a unificar os diversos padrões proprietários, facilitando, dessa maneira, a busca das melhores soluções de conexões de chão de fábrica.

O gerenciamento de redes é uma tarefa complexa, envolvendo a configuração, monitoração e controle dos mais variados componentes de *hardware* e *software*. Suas principais funções envolvem a configuração e monitoração do desempenho dos equipamentos, o controle de acesso aos recursos da rede, a contabilização dos recursos disponíveis e custos envolvidos na sua utilização e a localização e correção dos problemas (falhas) ocorridos nas redes.

Para estas atividades, a habilidade de adquirir informações sobre os equipamentos envolvidos e as mudanças ocorridas nestes é um fator fundamental. Assim, para manusear a grande quantidade de dados provenientes da ampla gama de tipos de equipamentos existentes nas redes, o uso de protocolos de gerenciamento padronizados específicos para o gerenciamento de redes se torna necessário.

Uma das razões que alavancaram a criação do padrão FF foi justamente a dificuldade em realizar a compatibilização de dispositivos de diferentes fabricantes. Assim o novo padrão a ser criado deveria ser aberto de forma que dispositivos de fabricantes variados pudessem interoperar sem perda de funcionalidades.

Uma observação importante é que a especificação do modelo FF foi desenvolvida pelos grandes fornecedores, membros da *Fieldbus Foundation*, sem a participação direta da comunidade acadêmica. Talvez por isto, até hoje, é extremamente difícil obter algumas informações técnicas acerca do modelo FF que não estejam descritos nos *Technical Overview* distribuído pelos fornecedores. De uma forma simplificada, a *Fieldbus Foundation* realizou a divulgação de um *Technical Overview* para os fornecedores dos dispositivos FF formataram e

republicaram este manual com o objetivo de convencer os novos clientes potenciais a aderirem ao Sistema FF.

Alguns materiais citaram que as MIBs dos dispositivos FF possuiriam informações de número de *reset*, tempo entre falhas, perdas de pacotes, perdas de comunicação com o LAS e identificação do dispositivo. Contudo em nenhum dos materiais pesquisados foi informado com exatidão os dados contidos nas MIBs e se os mesmos são acessíveis utilizando SNMP V1.

Hoje existem empresas especializadas em vender os projetos de produtos FF, mas estas empresas vendem o projeto dos dispositivos e a implementação dos protocolos e não as especificações. Esta modalidade de produção de produtos FF funciona como uma franquia *Fast-Food* onde o investidor implanta o modelo da matriz, incluindo instalações físicas, maquinários e treinamento de pessoal.

O estudo realizado neste trabalho demonstra que é possível a utilização do protocolo SNMP para gerenciar, monitorar e adquirir informações dos diversos elementos que compõem o modelo FF. Estas informações podem ser utilizadas para os mais diversos fins, variando desde sistema de monitoração de falhas até controle de produção integrado à sistemas de ERP (Figura 39).

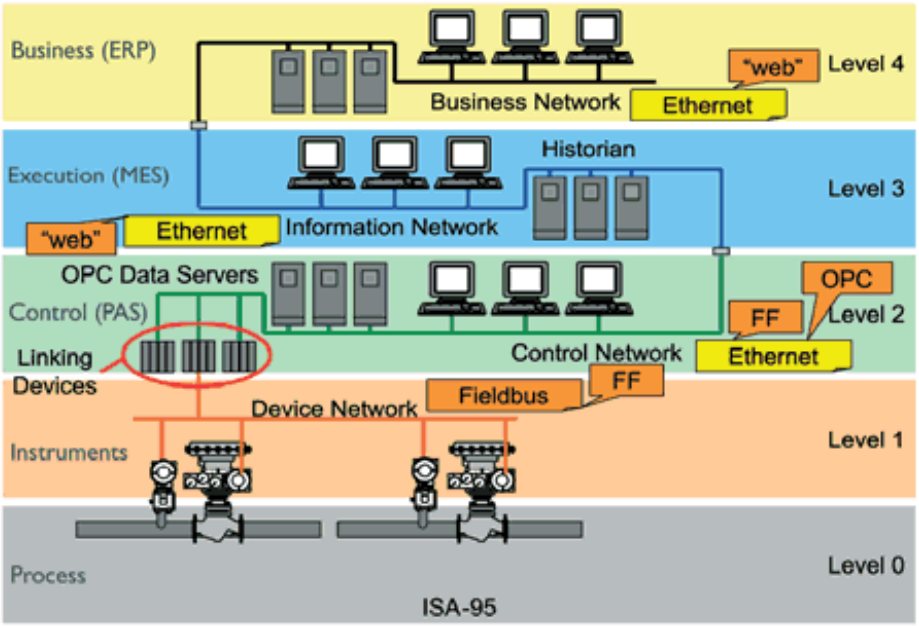

Figura 39 – Sistema integrado de comunicação Fieldbus Foundation
Fonte: Artigo Fieldbus, Ethernet and Reality Convergence Jonas Berge - SMAR.

O modelo de rede de automação proposto pela *Fieldbus Foundation* busca a integração das informações e por isto apresenta forte tendência a conquistar maior participação no mercado. Contudo a adoção em massa deste modelo envolve a quebra do paradigma de utilização de controle centralizado pelos engenheiros da área de automação.

Assim o maior desafio a ser enfrentado pelo FF é a mudança da filosofia de controle centralizado para um sistema de controle descentralizado. Neste aspecto estudos de tempo de resposta do FF pode ajudar na aceitação do mesmo. Outro ponto do FF, como sistema de controle "aberto", que se faz necessário é a identificação de quais variáveis de controle podem ser acessadas através de SNMP e quais os tipos de restrições existem ao acesso e modificação da mesma e até mesmo quais as restrições que deveriam existir.

7 – REFERÊNCIAS BIBLIOGRÁFICAS

Allen-Bradley. DeviceNet-Technical Overview - Ray Romito/ Allen Bradley, 1996.

CISCO SYSTEMS INC. Cisco Systems Inc. *Internetworking Technologies Handboo*k. 4ª Ed., 2003, Capítulo 56.

EMERSON. *Fieldbus Foundation Comunications*, Emerson, Proccess Management, 2002.

EVANS&WASHBURN. *TCP/IP Running a Successful Network*, Evans, J.T. & Washburn K., 2ª Ed. Harlow: Addison-Wesley, 1996

FIELDBUS FOUNDATION. *Technical Overview - Fieldbus Foundation*, Fieldbus Foundation, 2003.

FISHER-ROSEMOUNT. *Understanding Foundation Fieldbus Technology*, Fisher-Rosemount,1998.

GOMES. Integração Industrial, A terceira Revolução, Bruno Souza Gomes, 2005.

MEDILLO. *Introduccion SNMP*, Vincenzo Medillo , 2003.

MONTEZ. Redes de Comunicação Para Automação Industrial, Carlos Montez, 2005.

OGATA. Engenharia de Controle Moderno, Katsuhiko Ogata, 4ª Ed, 2003.

OLIVEIRA. Redes para Automação Industrial, L. Oliveira, 2005.

PERLMAN. *Interconnections: bridges, routers, switches, and internetworking protocols* . 2nd. ed. Massachusetts Harlow, Addison-Wesley, Radia Perlman, 2000. 537p

RODRIGUES. Apostila Informática Industrial, Pedro Ivo Rodrigues, 2007.

SEIXAS FILHO, UFMG. Apostila Foundation Fieldbus, Constantino Seixas Filho, UFMG.

SITE http://www.fieldbus.org, Acesso em 2007.

SMAR FF. *Fieldbus Tutorial, A Foundation Fieldbus Technology Overview*, SMAR, 2001.

SMAR PROFIBUS. SMAR – PROFIBUS-DP/PA, César Cassiolato, 2003.

STALLINGS. SNMP: SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, (3rd Edition), William Stallings, 1999.

TANENBAUM. *Network Computers*, Andrew S. Tanenbaum, 4ª Ed, 2007.

VICENTE. *Foundation Fieldbus High Speed Ethernet Control System*, Sean J. Vicente, 2001.

YOKOGAWA. *Fieldbus Book - A Tutorial*, YOKOGAWA Electric Corporation, 2001.