



Camada de Enlace de Dados

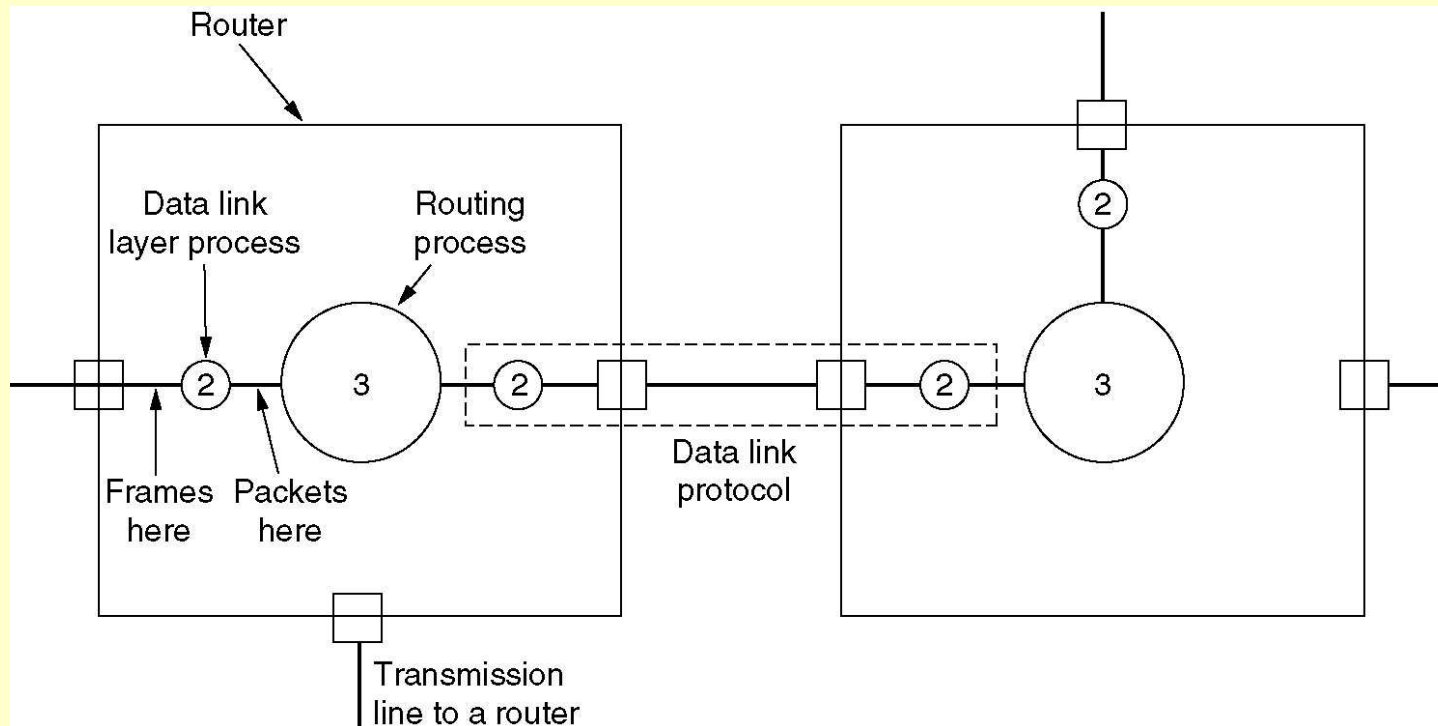


Funções da Camada de Enlace

- Fornecer serviços a Camada de Rede
- Delimitação de Quadros
- Controle de Erros
- Controle de Fluxo
- Variedade de Serviços
- Controle de Acesso
- Multiplexação



Serviços Fornecidos para a Camada de Rede



A localização da Camada de Enlace



Definição de Quadro

Quadro:

Header

User Data

Trailer

O quadro é composto de vários campos:

Header - Endereço de Destino, Endereço de Origem, Contagem

User Data - Parte útil do quadro, onde os dados são transportados

Trailer – FCS (Campo de Verificação de Erros)



Delimitação de Quadro

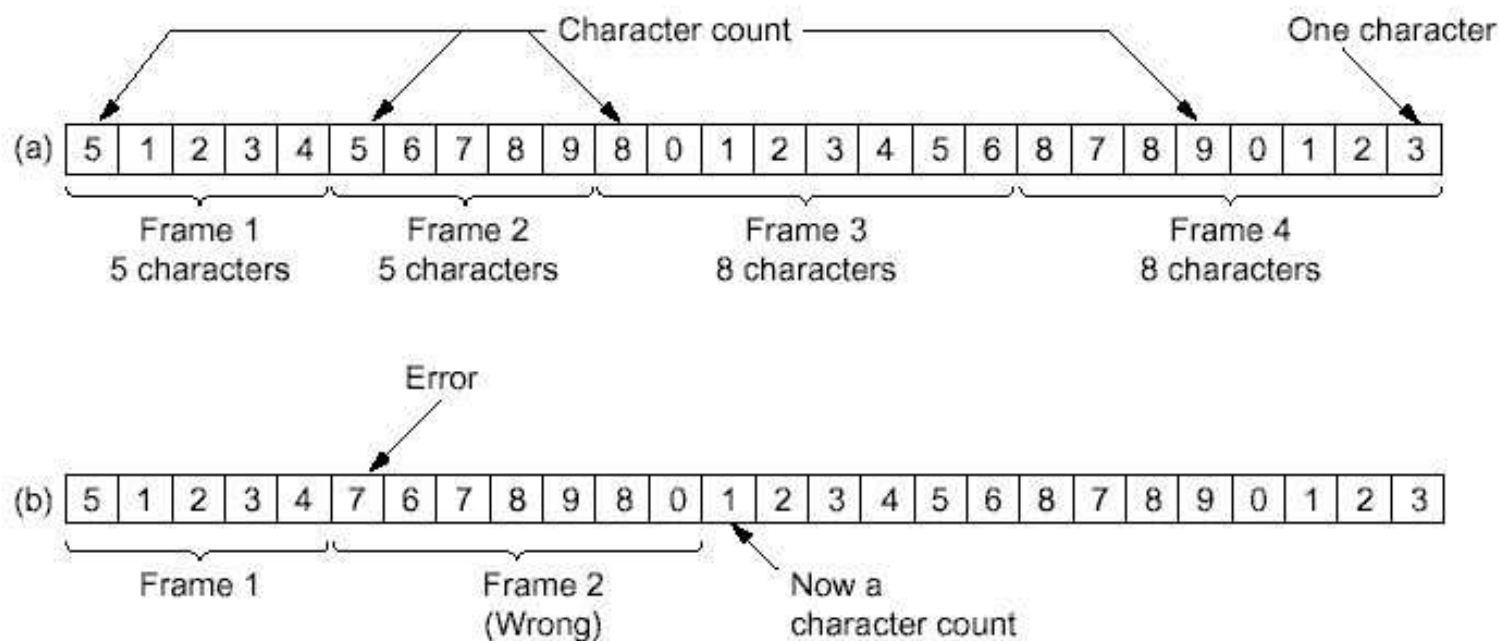
- A delimitação de quadros é necessária para a correta recepção dos quadros, entre seu início e fim, técnicas utilizadas:
 - Contagem de Bytes
 - Orientado a Bit
 - Orientado a Caractere
 - Violação de Código de Camada Física



Contagem de Bytes

- Um campo do cabeçalho é utilizado para definir o tamanho do quadro
- Sabendo a quantidade de caracteres (bytes) a serem transmitidos, a camada física sabe onde termina o quadro
- Esse campo normalmente é utilizado quando o campo de dados possui tamanho variável, possibilitando a delimitação de quadros

Contagem de Bytes



A character stream. (a) Without errors. (b) With one error.



Contagem de Bytes

- Desvantagem: o campo de contagem pode ser alterado. A camada de enlace na recepção não saberá onde começa/termina mesmo com campo FCS
- O método de contagem de bytes não deve ser usado sozinho



Orientado a Bit

- Dados transmitidos como uma sequência de bits
- Para que o receptor reconheça o início/fim de um quadro (*frame*) é utilizada uma sequência única de bits, chamada *flag* conforme figura a seguir





Orientado a Bit

- Problema: Sequência de bits igual ao *flag* precisar ser transmitida no meio do quadro
- Solução: usar bit de enchimento (*stuffing bit*)
- A cada 5 bits de dados sucessivos iguais a 1 usar o *stuffing bit* ($=0$)
- Exemplo de aplicação: protocolo *HDLC* (*Host Data Link Control*)



Orientado a Bit

EXEMPLO:

Flag = 01111110

(a) 011011111111110111110111111001

(b) 0110111110111101111100111110001

(c) 01111110 0110111110111101111100111110001
01111110

(d) 0110111110111101111100111110001

(e) 011011111111110111110111111001



Orientado a Bit

- (a) Mensagem a ser transmitida
- (b) Mensagem com bits 0 inseridos (sublinhados)
- (c) Mensagem pronta para ser transmitida no meio físico (com bits de enchimento e flags)
- (d) Mensagem recebida sem flags, bits de enchimento identificados
- (e) Mensagem recebida sem bits de enchimento



Orientado a Caractere

- Utiliza caracteres especiais para delimitar início e fim de quadro
- O problema é que uma sequência igual aos caracteres especiais pode aparecer no campo de dados
- A solução é utilizar um outro caractere especial (DLE) antes da sequência especial, se esta aparecer no campo de dados



Orientado a Caractere

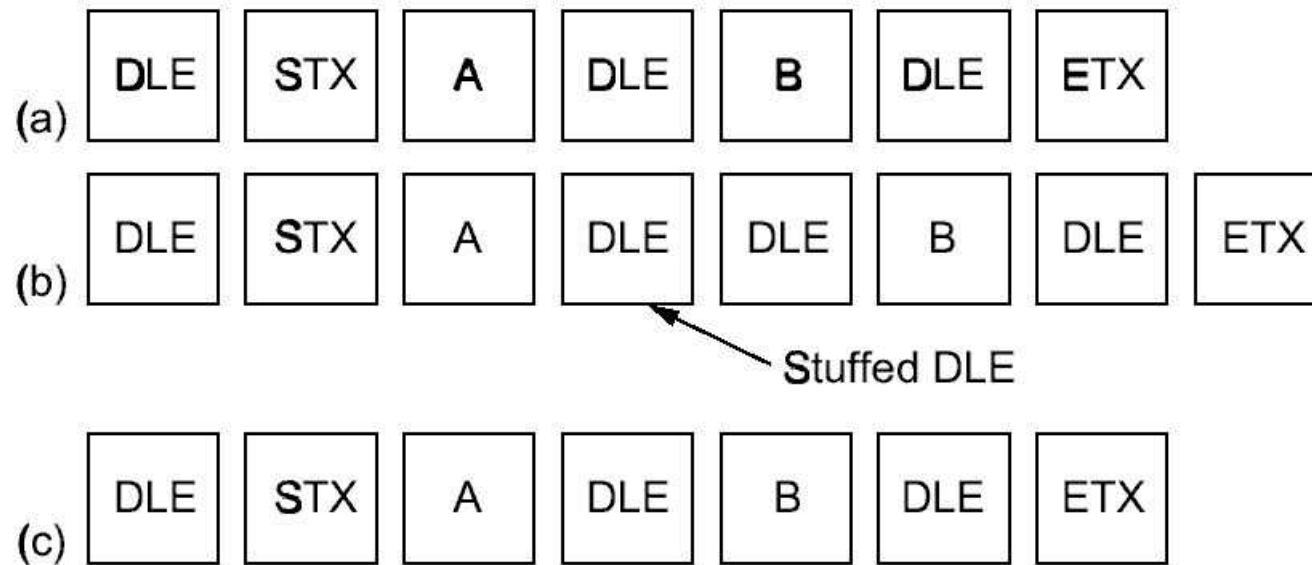


Fig. 3-4. (a) Data sent by the network layer. (b) Data after being character stuffed by the data link layer. (c) Data passed to the network layer on the receiving side.

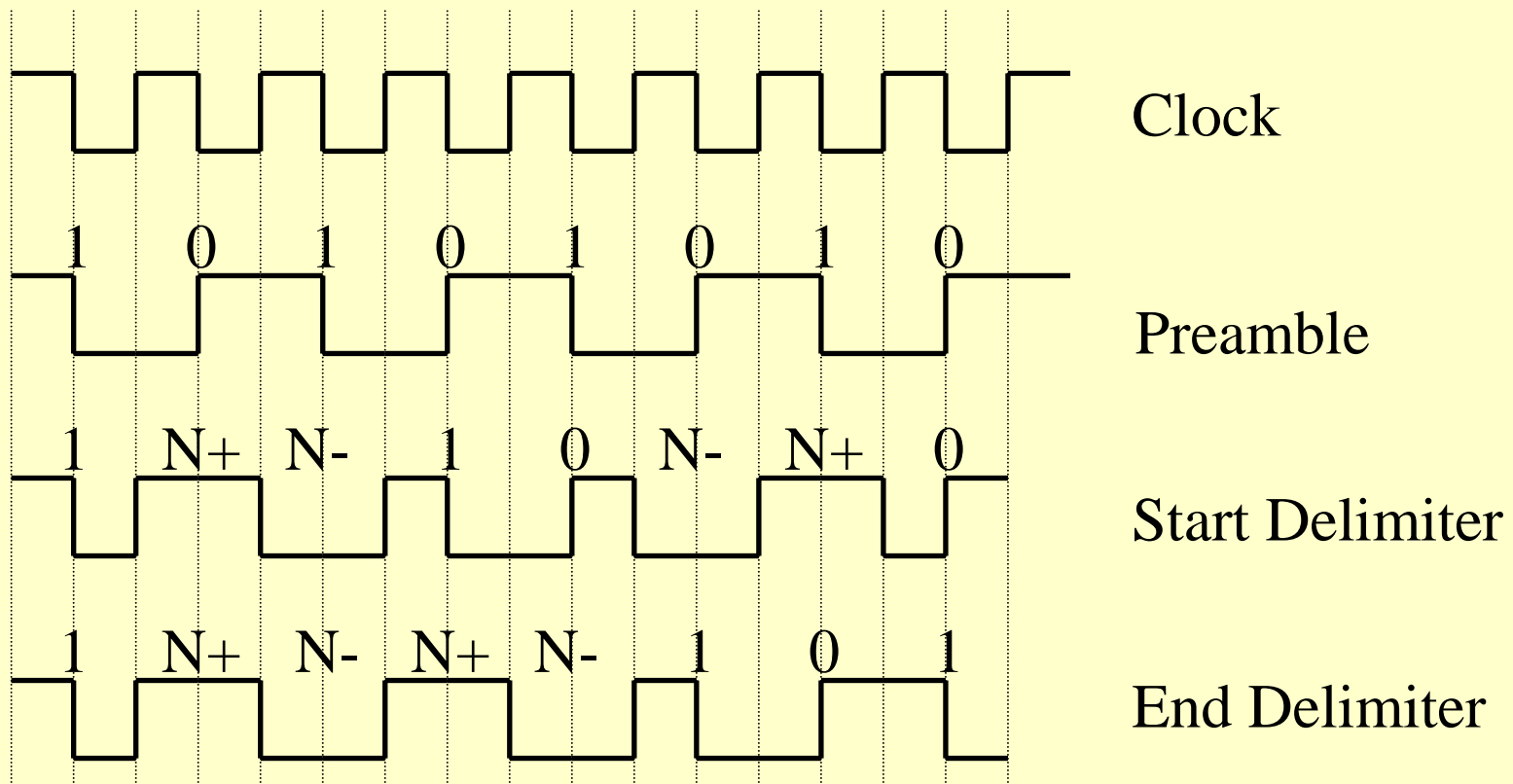


Violação de Código de Camada Física

- Cada nível na camada física é representado por uma sequência de bits
- Exemplo: Código Manchester (rede Ethernet)
 - Bit 1: sequência 01
 - Bit 0: sequência 10
 - Sequências 00 e 11 podem ser usadas na sincronização do quadro (início/fim de quadro)

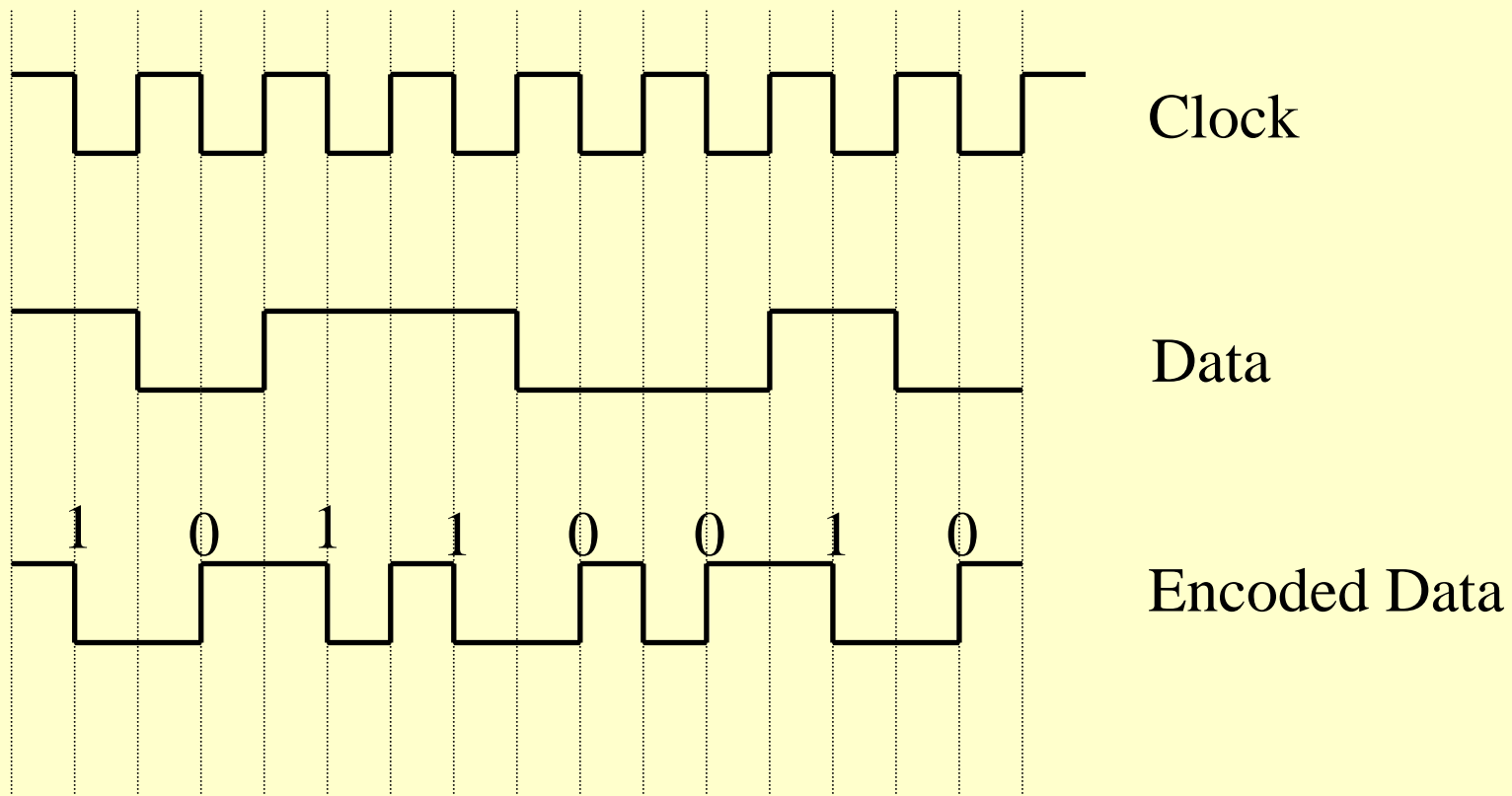


Violação de Código de Camada Física





Violação de Código de Camada Física





Controle de Erros

- Um pacote que chega com erro pode ser retransmitido ou ter seu erro corrigido (enlaces mais caros, satélite).
- Para garantir a comunicação confiável, a Camada de Enlace:
 - Verifica erro / solicita retransmissão
 - Corrige erro (enlaces mais caros) – exemplo: código de Hamming



Controle de Erros

- Toda mensagem recebida sem erros é sucedida por um pacote de confirmação de recebimento (ACK), enviada pelo receptor
- Para detectar erros que tornem a mensagem não inteligível é utilizado um temporizador para verificar se houve *timeout* (estouro de tempo = tempo suficiente para pacote e respectivo reconhecimento chegarem)
- Um reconhecimento negativo (NAK) acelera o processo de retransmissão



Controle de Fluxo

- Objetivo: transmissão ocorre apenas quando receptor estiver pronto. Se buffer de recepção estiver cheio, transmissor deve parar envio de quadros.
- Usa quadros especiais
 - *RNR = receive not ready*
 - *RR = receive ready*



Tipos de Serviço

- A camada de enlace se comunica em duas instâncias, com a camada de enlace do outro nó e com a camada acima (camada de rede)
- A camada de enlace fornece 2 tipos de serviço para a camada de rede:
 - Sem reconhecimento
 - Com reconhecimento



Sem Reconhecimento

- O transmissor envia pacote e não recebe pacotes de reconhecimento (ACK)
- Também conhecido com envio não confiável
- O controle de erros deverá ser feito pelas camadas superiores (camada de rede)
- Para redes onde taxa de erros é muito baixa ou onde a demora na espera pela confirmação é crítica
- A demora pode ser mais crítica que a perda de pequenas parcelas da mensagem (ex.: tráfego de voz)



Com Reconhecimento

- O transmissor envia quadro e aguarda quadro de reconhecimento (ACK) ou ocorrência de *timeout*
- Utilizado no envio de pequeno volume de dados de maneira confiável, quando não compensa esperar início e encerramento da conexão
- A perda de quadros de reconhecimento (ACK) pode implicar no recebimento de quadros repetidos. Camada de Enlace deve possuir um mecanismo para o descarte de cópias



Controle de Acesso

- Consiste num esquema definindo como e quando cada estação deverá acessar o meio físico
- É essencial quando o meio físico é compartilhado por várias estações (*Ethernet*)
- Existem várias técnicas que permite controlar o acesso ao meio (*Token ring*, *Token bus*, CSMA, *Polling*, ...)
- Algumas técnicas de Controle de Acesso são mais adequadas a uma topologia de rede específica (ex.: anel - *Token Ring*)



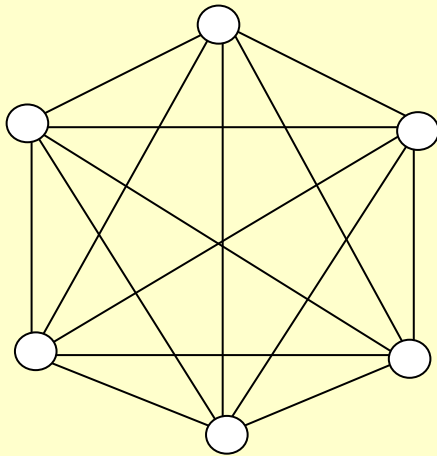
Multiplexação

- A partir do instante que uma estação pode fornecer um caminho confiável de envio livre de erros, esse caminho pode ser multiplexado (e compartilhado) por vários usuários nas camadas superiores
- Podem existir várias SAP (*Service Access Point*) da Camada de Enlace para a Camada de Rede
- Exemplo: programas de e-mail, *browser*, FTP, TELNET abertos simultaneamente com uma única conexão física de rede ou modem

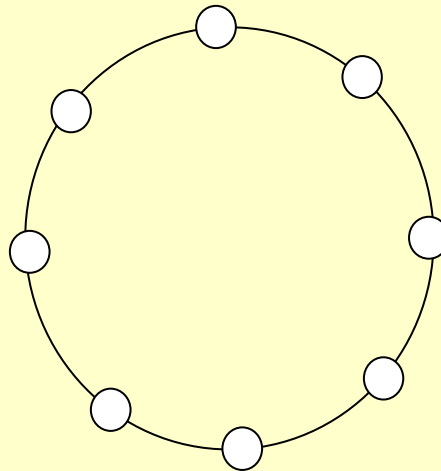


Redes Locais

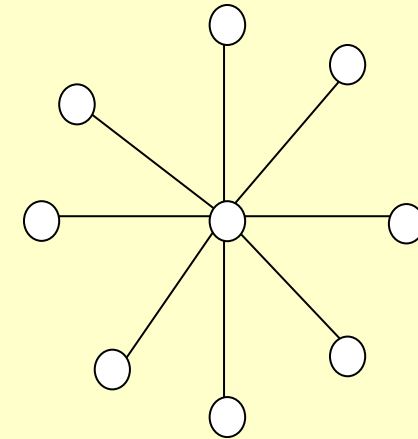
Topologias de Rede



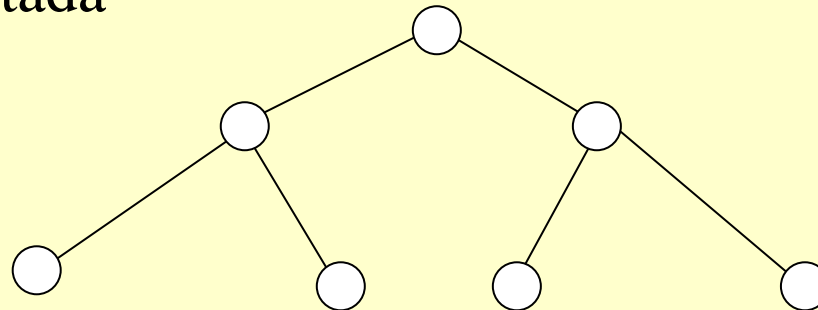
Rede Totalmente
Conectada



Rede em Anel

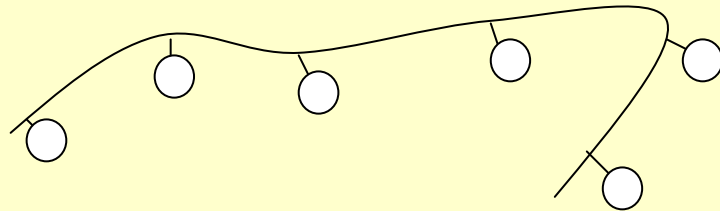


Rede em Estrela

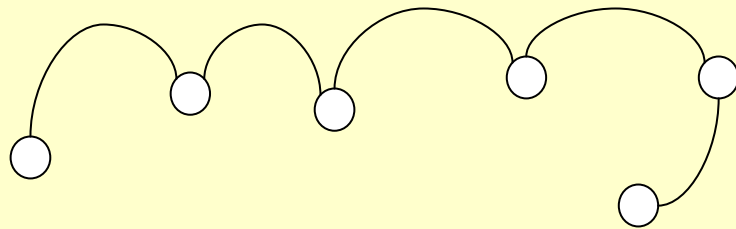


Rede em
Árvore

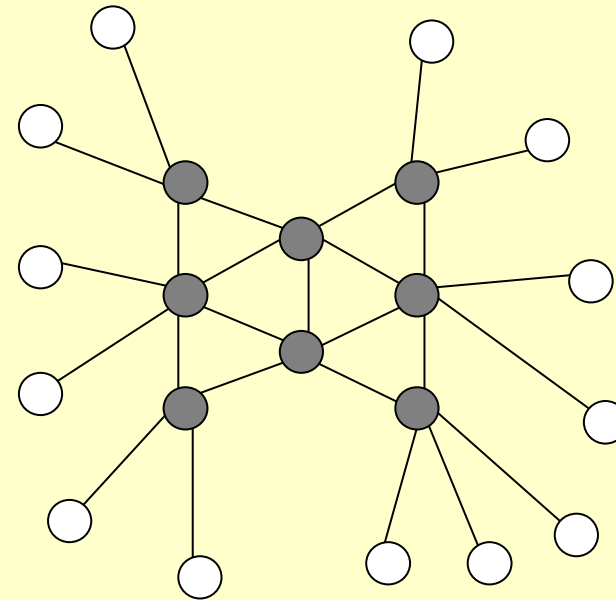
Topologias de Rede



Rede em Barra



Rede "Daisy Chain"



Rede em Grade



Topologias de Rede

- Rede Totalmente Conectada
 - Usada em supercomputadores (hipercubo)
 - Alto *throughput*
 - Redes de pequeno diâmetro (placa ou gabinete)
 - *Links* não precisam ser muito velozes



Topologias de Rede

- Rede em Estrela
 - não necessitam roteamento
 - nó central: *hub* ou *switch*
 - desempenho dependente do nó central
 - confiabilidade dependente do nó central
 - dificuldade de ampliação (depende do nó central)



Topologias de Rede

- Rede em Grade (ou hierárquica)
 - estrutura central formada por comutadores de rede
 - vários caminhos entre dois nós
 - suporta várias conexões simultâneas entre nós, limitada à quantidade e interligações entre os comutadores
 - boa confiabilidade
 - Ex: Sistema IA (Invensys)



Topologias de Rede

- Rede em Barra
 - rede do tipo difusão (mensagem difundida para todos os nós)
 - geralmente mais barata e mais simples
 - o cabo percorre todos os nós da rede, menor confiabilidade da rede
 - curto circuito num nó paralisa a rede
 - cada nó usa derivação no meio físico
 - Exemplos: EIA-485, Fieldbus Foundation



Topologias de Rede

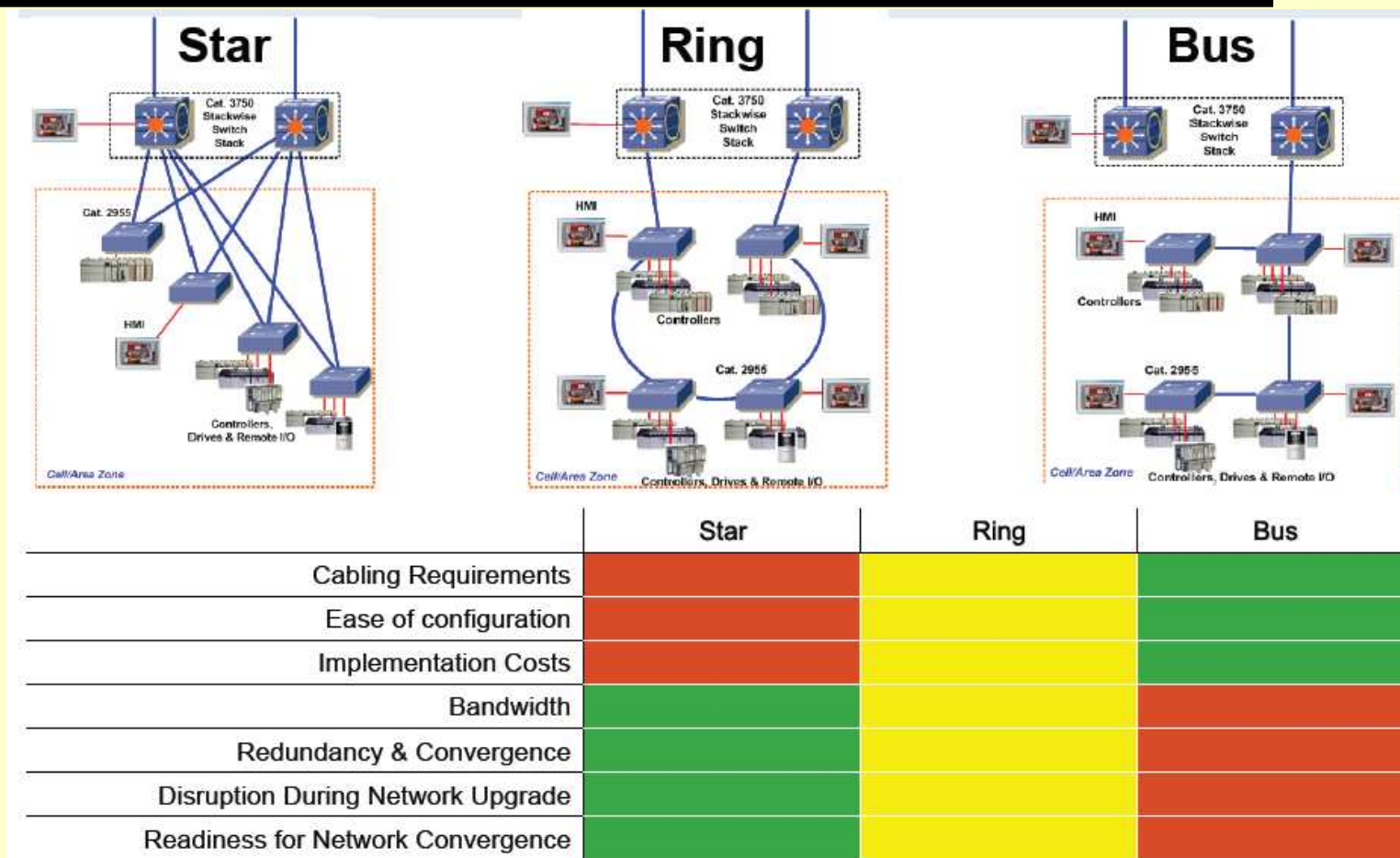
- Rede em Anel
 - mensagem entra no nó, circula na rede (anel) e é retirada quando retorna ao nó gerador ou pelo nó destinatário
 - cada nó é um repetidor de rede
 - ampliação requer interrupção da rede
 - Normalmente mais confiável que a rede em barra



Topologias de Rede

- Rede “Daisy Chain”
 - topologia similar a barra
 - cada trecho de rede é um enlace ponto a ponto
 - a desconexão de um nó interrompe a rede
 - redes de pequeno diâmetro (centenas de metros)

Comparação Estrela x Anel x Barra





Protocolos de Enlace

- No que se refere à camada de enlace de dados, o pacote repassado a ela pela camada de rede através da interface consiste em dados puros, em que cada bit deve ser entregue à camada de rede de destino.
- O fato de a camada de rede de destino poder interpretar parte do pacote como um cabeçalho não tem nenhum interesse para a camada de enlace de dados.
- Quando a camada de enlace de dados aceita um pacote, ela o encapsula em um quadro, acrescentando um cabeçalho e um final de enlace de dados
- Portanto, um quadro consiste em um pacote incorporado, em algumas informações de controle (no cabeçalho) e em um Campo de Verificação de Erro (FCS) no final
- Em seguida, o quadro é transmitido à camada de enlace de dados da outra máquina

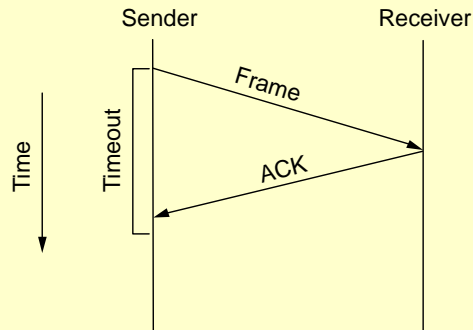


Reconhecimento (Ack) e Timeouts

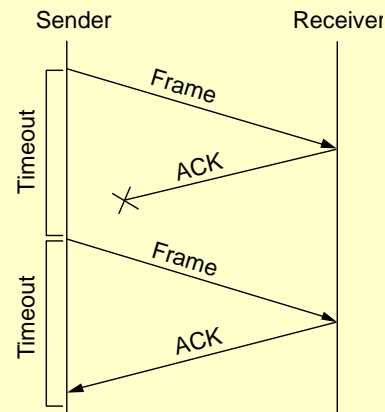
- O pacote de reconhecimento (*acknowledgement* = ACK) é enviado em resposta a um pacote recebido sem erros
 - Tornar um pacote ACK é basicamente obtido alterando-se um campo de um pacote
 - Os dados podem ser transportados em pacotes ACK (*piggyback*)
- Um *timeout* (estouro de tempo limite) é uma condição na qual um pacote ACK não foi recebido dentro de um limite de tempo esperado
 - Um timeout dispara a re-transmissão de um pacote pelo Host que está enviando dados
 - Como ajustar o timeout?



Reconhecimento (Ack) e Timeouts



(a)

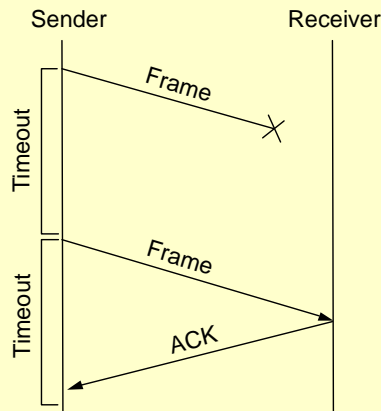


(c)

(a) Envio normal, com ACK antes do Timeout

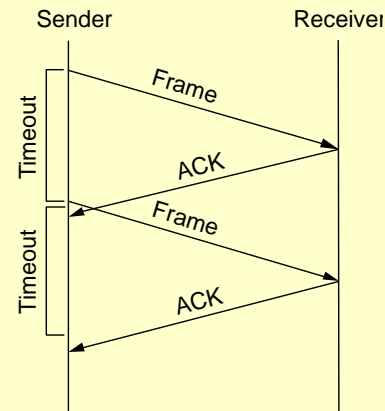
(b) Pacote perdido, ocorre Timeout e transmissor re-envia

(c) ACK perdido, ocorre Timeout e transmissor re-envia



(b)

GIL PINHEIRO



(d)

(c) ACK chega após Timeout e transmissor re-envia. Pacote é enviado duas vezes (pacote repetido)

Camada de Enlace de Dados



Protocolos de Janela Deslizante

- Devido ao baixo rendimento do canal na comunicação *Stop and Wait*, são necessários outros protocolos mais eficientes
- Protocolos utilizados no envio de quadros entre estações, de maneira bidirecional
 - Protocolo com um Bit
 - Protocolo Retorna a N
 - Protocolo de Repetição Seletiva

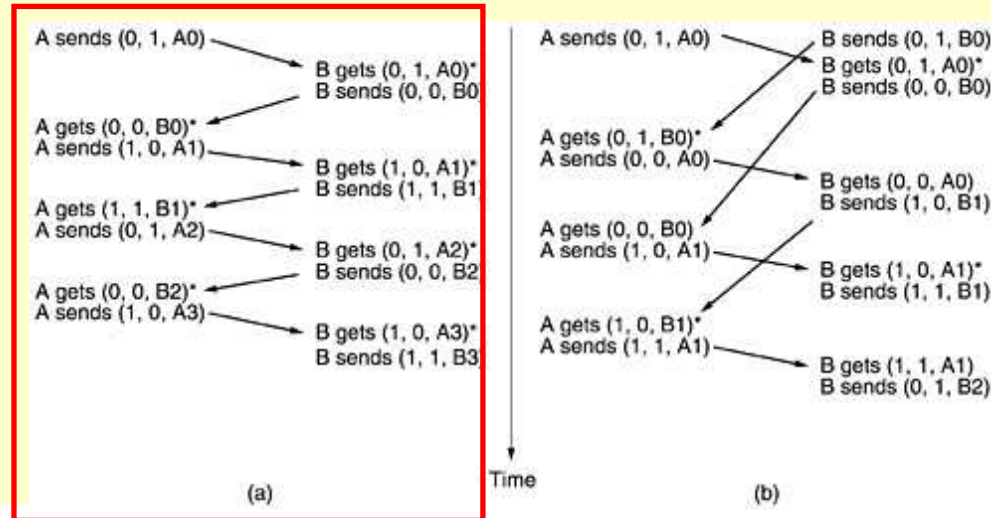


Protocolos de Janela

- O transmissor precisa armazenar os pacotes já transmitidos para o caso de pacotes perdidos tiverem de ser re-enviados
- O receptor precisa armazenar os pacotes para o caso de pacotes recebidos fora de ordem ou perdidos até que a sequência esteja correta
 - É necessário o Controle de Fluxo
- Como prevenir o estouro do buffer de recepção?
 - Receptor avisa ao transmissor o tamanho de seu buffer durante o estabelecimento da conexão
- Como assegurar a confiabilidade em transmissão *pipelined*?
 - Protocolo de Janela do tipo Retorna a N (*Go-Back-N*)
 - Enviar todos os N pacotes não reconhecidos quando uma perda for sinalizada
 - Ineficiente
 - Protocolo de Janela do Tipo Retransmissão Seletiva (*Selective repeat*)
 - Somente re-envia os pacotes não recebidos e não reconhecidos (*unACKed*)
 - Implementação mais complexa



Protocolo de Janela de 1-Bit

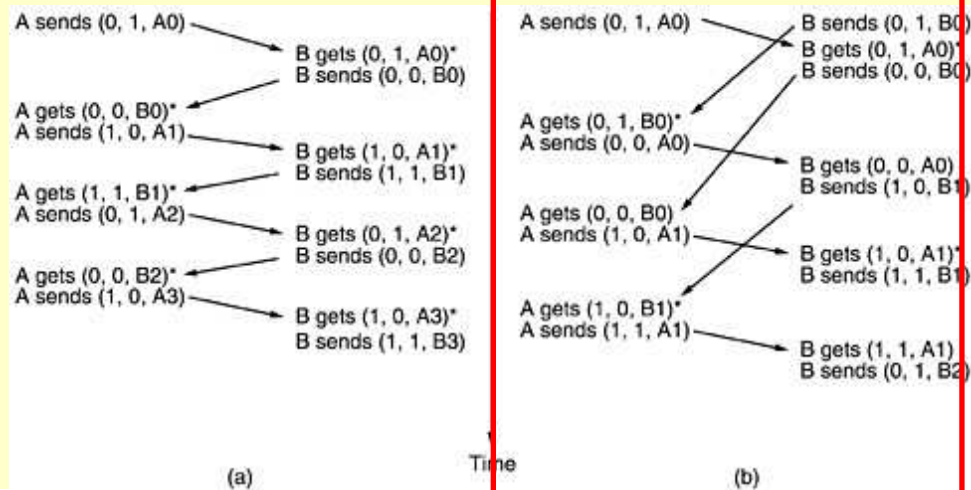


(a) Envio sem erros e sem repetição de quadros

- Os quadros são numerados com 1 bit (0 ou 1)
- Usa ACK também de apenas 1 bit (0 ou 1), referenciando o número do quadro recebido
- O Host A não envia um novo quadro até que saiba que foi recebido o quadro anterior em B
- Utiliza o esquema espera confirmação para enviar (*Stop and Wait*)



Protocolo de Janela de 1-Bit



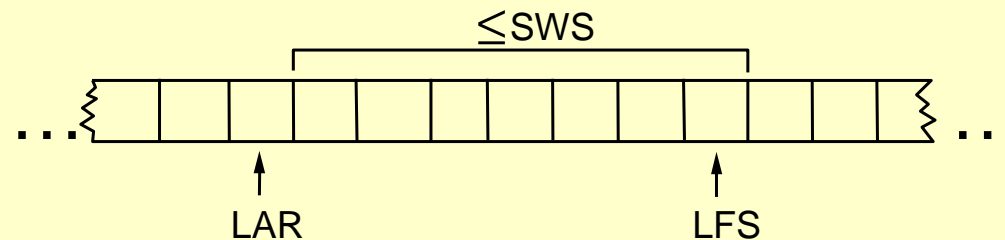
(b) Envio com quadros repetidos

- Se B esperar pelo primeiro quadro de A antes de enviar um de seus quadros, a sequência será a da parte (a), e todos os quadros serão aceitos. Porém, se A e B iniciarem a comunicação ao mesmo tempo, seus primeiros quadros se cruzarão e as camadas de enlace de dados recairão na situação (b). Em (a), cada quadro recebido traz um novo pacote para a camada de rede; não há cópias. Em (b), metade dos quadros contem cópias, embora não haja erros de transmissão.
- Situações similares podem ocorrer como resultado de timeouts prematuros, mesmo quando está claro que um lado começa primeiro. Na verdade, ao ocorrer vários timeouts prematuros, os quadros poderão ser enviados três vezes ou mais
- Logo, a janela de 1 bit não evita os quadros repetidos, mesmo sem erros de comunicação



Protocolos de Janela Deslizante: Transmissor

- Atribuir número de seqüência para cada quadro (**SeqNum**)
- Utiliza três variáveis de estado:
 - Tamanho da janela de envio (*send window size* = **SWS**)
 - Último quadro reconhecido (*last acknowledgment received* = **LAR**)
 - Último quadro enviado (*frame sent* = **LFS**)
- Manter constante: **$LFS - LAR \leq SWS$**

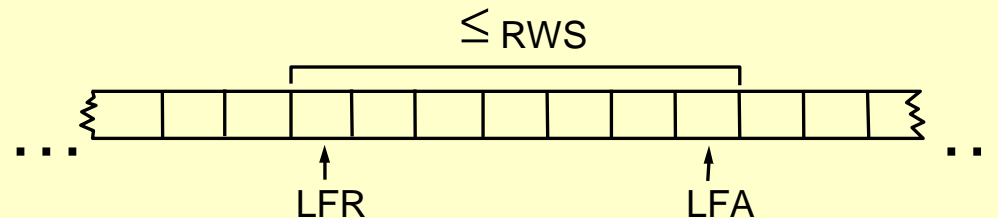


- Avançar **LAR** quando um **ACK** chegar
- Armazenar até **SWS** quadros



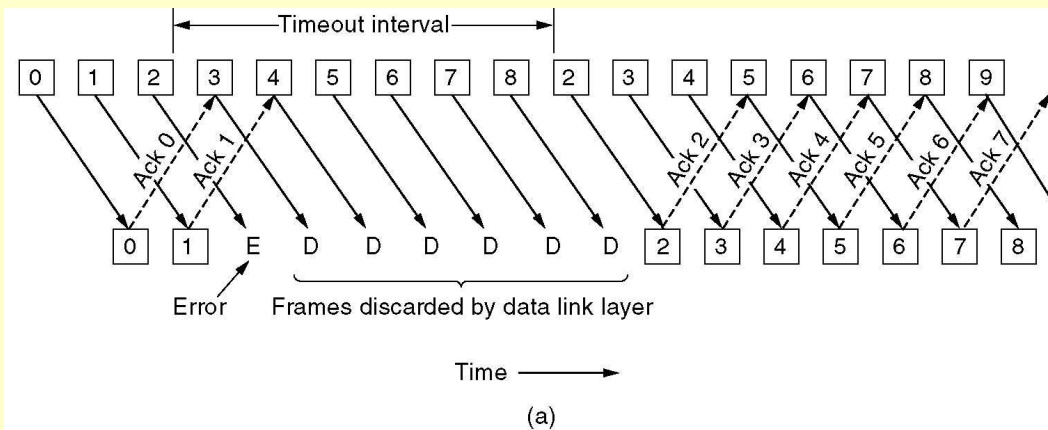
Protocolos de Janela Deslizante: Receptor

- Utiliza três variáveis de estado:
 - Tamanho da janela de recebimento (*receive window size* = **RWS**)
 - Último quadro recebido (*last frame received* = **LFR**)
 - Maior quadro aceitável (*largest frame acceptable* = **LFA**)
- Manter constante: **LFA - LFR \leq RWS**

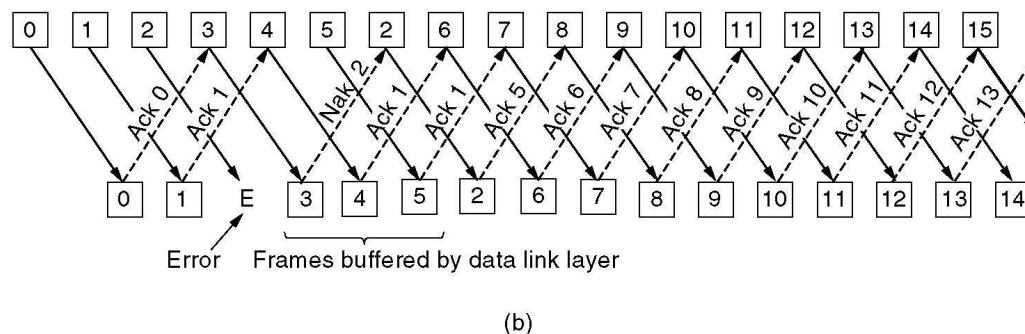


- Pacote com número **SeqNum** chega:
 - Se **LFR < SeqNum \leq LFA**, então: **Aceitar o pacote**
 - Se **SeqNum \leq LFR** ou **SeqNum > LFA**, então: **Descartar o pacote**
- Enviar ACKs cumulativos – enviar ACK para o quadro de maior número, de modo que os de número menor que este tenham sido recebidos

Protocolos de Janela Deslizante



(a) Retorna a N (*Go-Back-N*) - re-envia todos os quadros após o quadro com erro



(b) Retransmissão seletiva (*Selective repeat*) – re-envia apenas o quadro com erro



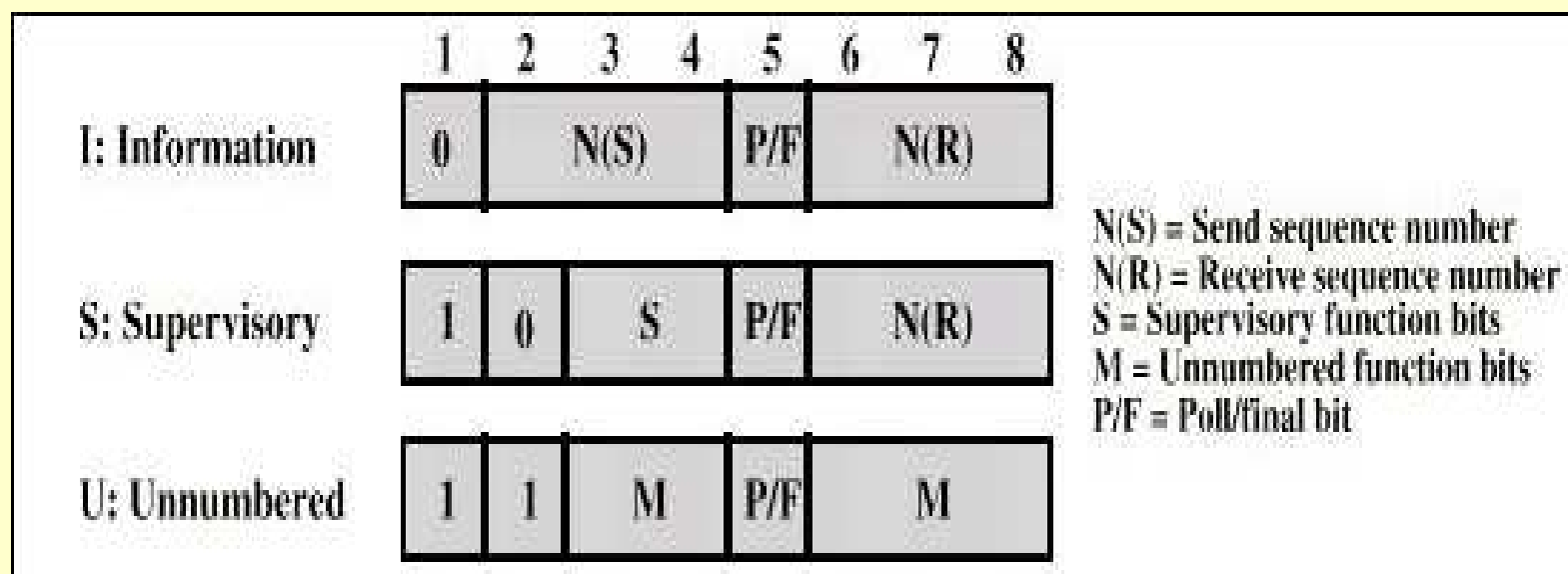
O Protocolo HDLC

- Protocolo de enlace padronizado pela ISO
- Existem três tipos de nós:
 - Primário: controlam o enlace e geram
 - Secundário: estão sob o controle do nó primário, respondem aos comandos solicitados. Quando existe mais do que um nó secundário, o nó primário mantém uma ligação lógica independente para cada um
 - Combinados: controla o link mas também responde a comandos
- Tipos de enlace lógico:
 - ABM: Modo Assíncrono Balanceado
 - NRM: Modo de Resposta Normal
 - ARM: Modo de Resposta Assíncrono



O Protocolo HDLC

Tipos de quadros



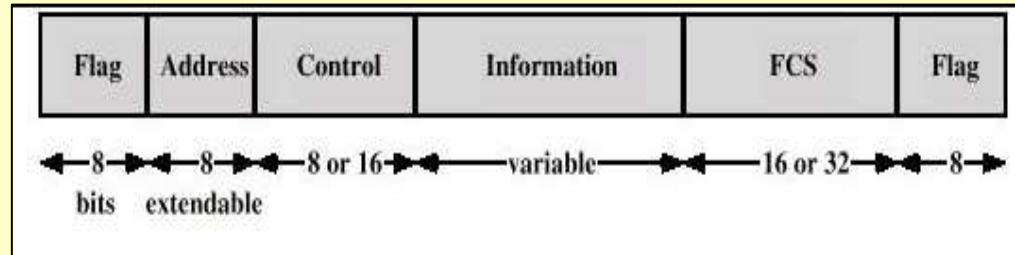


O Protocolo HDLC

- **P/F (*Normal Response Mode*)**
 - **Poll bit** para **comandos** (primário) e **Final bit** para **respostas** (P/F=1 indica que a estação Secundária i envia último quadro I)
- **P/F (*Asynchronous Balanced Mode*)**
 - A estação transmissora solicita um RR com P/F=1, quando não recebe resposta da estação receptora. Nesse caso, a receptora deve enviar uma resposta, com P/F=1 também evita a retransmissão de vários quadros, quando o ACK foi perdido



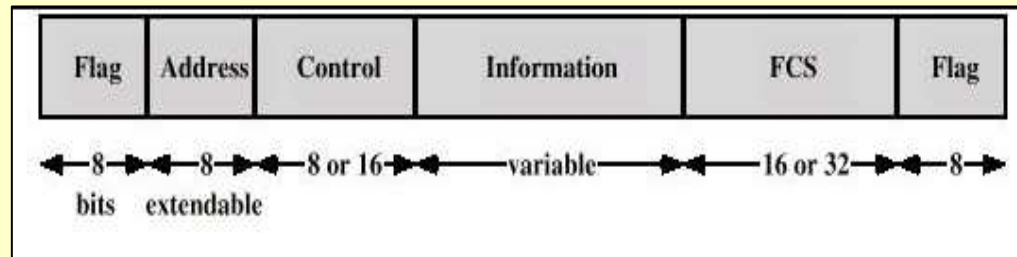
O Quadro HDLC



- **FLAG = 01111110**
 - Define início e fim de quadro
 - O mesmo FLAG pode ser usado para fim de um quadro e início de outro
 - Utiliza o mecanismo de **BIT-STUFFING**
- **CAMPO DE ENDEREÇO**
 - Indica a estação secundária que transmitiu ou a que deve receber o quadro
 - Este campo é desnecessário para ligações ponto-a-ponto;
 - O endereço **11111111** indica **broadcast**
- **CAMPO DE CONTROLE** : Define três tipos de quadro:
 - **Informação**: contém o dado a ser transmitido e o controle de fluxo e de erro pode ser efetuado pelo mecanismo de carona (*piggybacking*)
 - **Supervisão**: fornece o mecanismo adequado de feedback quando não é utilizado o mecanismo de carona
 - **Não-Numerado**: usado para gerenciar a conexão. Não possui número de seqüência (funções de controle suplementares)



O Quadro HDLC



- **CAMPO DE INFORMAÇÃO**
 - Contém os dados (informação) a serem transmitidos
 - Somente presente nos quadros do tipo Informação (I)
 - O tamanho do campo não é definido no padrão mas é múltiplo de 8 bits.
- **CAMPO FCS - Frame Check Sequence**
 - Contém o código de detecção de erro (CRC) calculado para todo frame, exceto FLAG
 - Geralmente utiliza CRC de 16 bits, podendo, opcionalmente, utilizar CRC de 32 bits



O Protocolo HDLC

– *Unnumbered Frame (U)*

Comando(C) Resposta (R)	Descrição
SNRM (C)	“Set Normal Response Mode”, é enviado pelo nó primário ao secundário indicando que deve entrar em modo de conexão normal, com campo de controle de 8 bits.
SARM (C)	“Set Asynchronous Response Mode”, idêntico ao SNRM, para o modo assíncrono.
SABM (C)	“Set Asynchronous Balanced Mode”, idêntico ao SNRM, para o modo assíncrono balanceado.
DISC (C)	“Disconnect”, informa o destinatário que a estação emissora deixará de estar ativa.
UA (R)	“Unnumbered Acknowledgment”, indica a aceitação do comando enviado.
DM (R)	“Disconnected Mode”, resposta enviada sempre que outro nó envia um comando para indicar que o nó não está conectado. Esta resposta é igualmente usada para indicar a não aceitação de um comando.
RD (R)	“Request Disconnect”, esta resposta é usada para pedir o fim de uma conexão estabelecida.
RSET (C)	“Reset”, este comando é usado para a recuperação de erros graves (FRMR), indica que o nó reinicializou o seu número de seqüência de envio, o destinatário deve também reinicializar o seu número de seqüência de resposta.
FRMR (R)	“Frame Reject”, indica que o quadro recebido tem um erro que não pode ser recuperado por ARQ



O Protocolo HDLC

– *Supervisory Frame (S)*

Código no Campo de Controle (Supervisory control bits)	Comando-Função	Descrição
00	RR	“Receiver Ready”, é um ACK que indica que o receptor se encontra pronto para receber mais quadros (“stop & wait”).
10	RNR	“Receiver Not Ready”, é um ACK que indica que o receptor não se encontra pronto para receber mais quadros (“stop & wait”).
01	REJ	“Reject”, é um NAK do modo “go-back-n”
11	SREJ	“Selective Reject”, é um NAK do modo “selective-repeat”

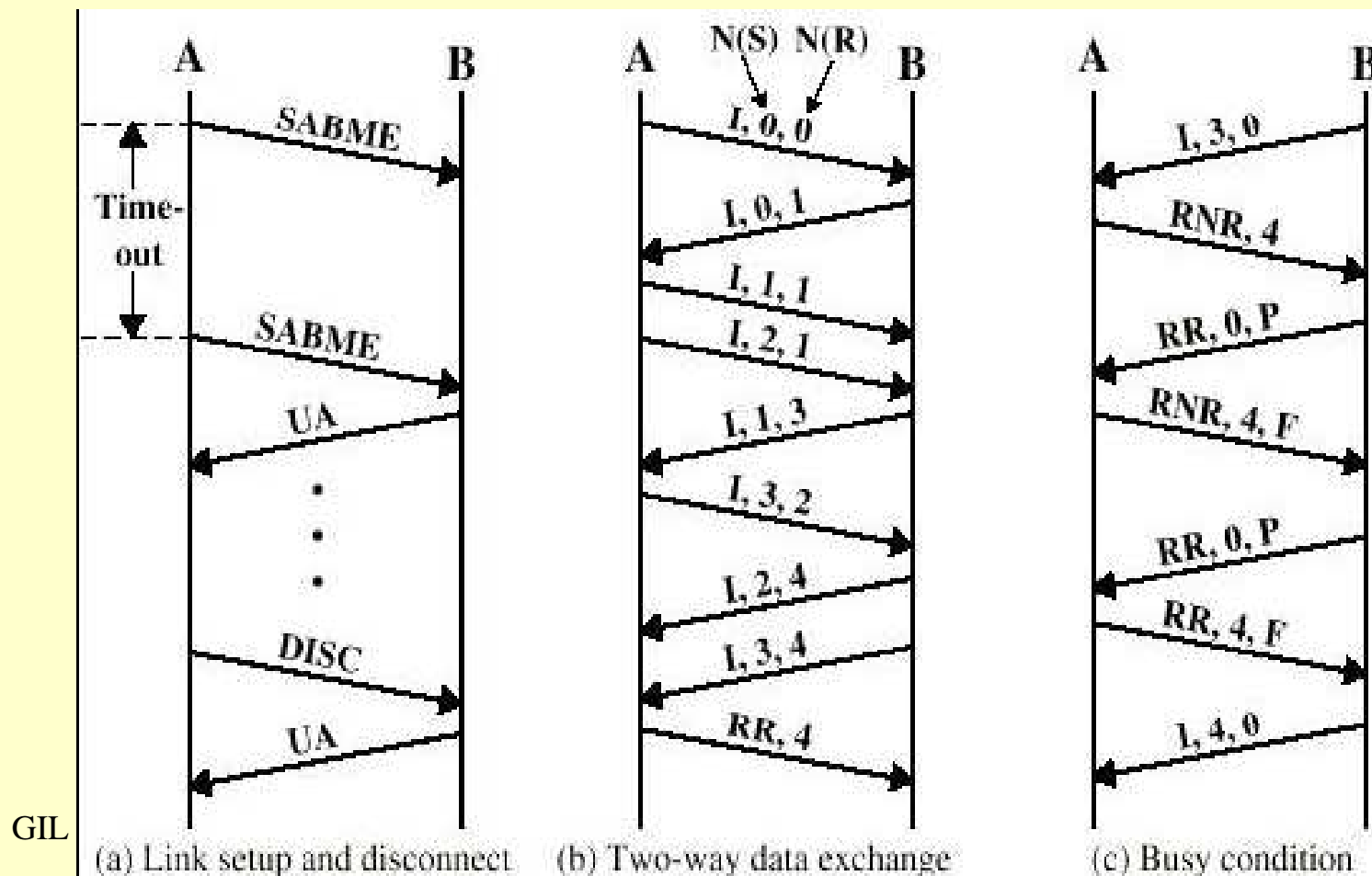


O Protocolo HDLC

- **Envolve 3 fases:**
 - ***Inicialização***
 - Sinaliza ao outro lado que se deseja iniciar uma comunicação
 - Especifica qual dos modos irá operar (NRM, ABM, ARM);
 - Especifica se será usado janela de 7 ou de 127 quadros
 - ***Transmissão de Dados***
 - Após a inicialização ter sido solicitada e aceita, passa-se à fase de transferência de dados em quadros de informação, usando N(S) e N(R);
 - Pode-se utilizar também quadros de supervisão, caso não tenha quadro I para enviar de carona (*piggybacking*)
 - ***Desconexão***
 - Qualquer das estações pode solicitar a desconexão

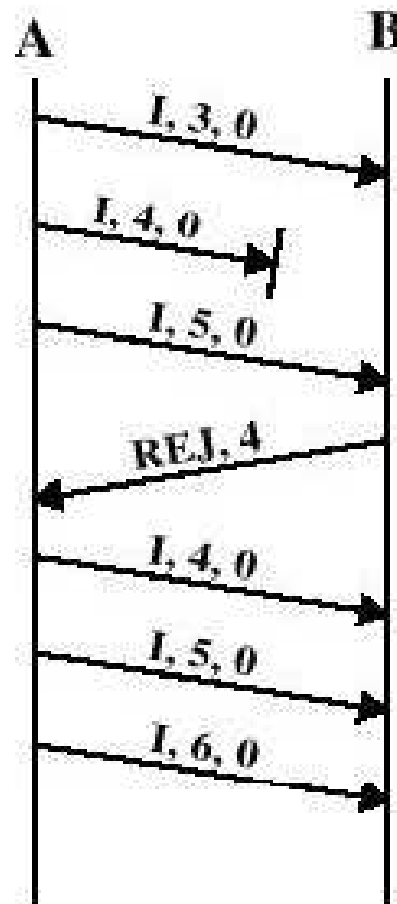
O Protocolo HDLC

- Seqüências de conexão e desconexão

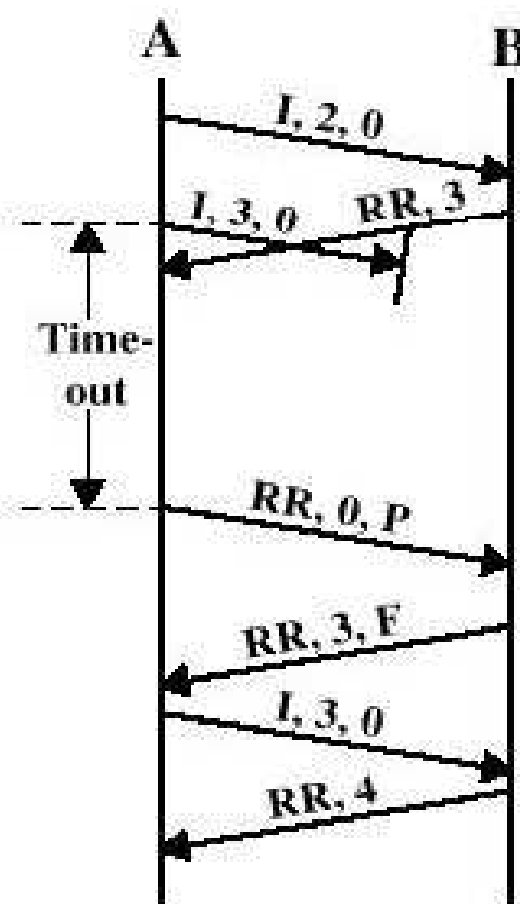


O Protocolo HDLC

- Seqüências de conexão e desconexão



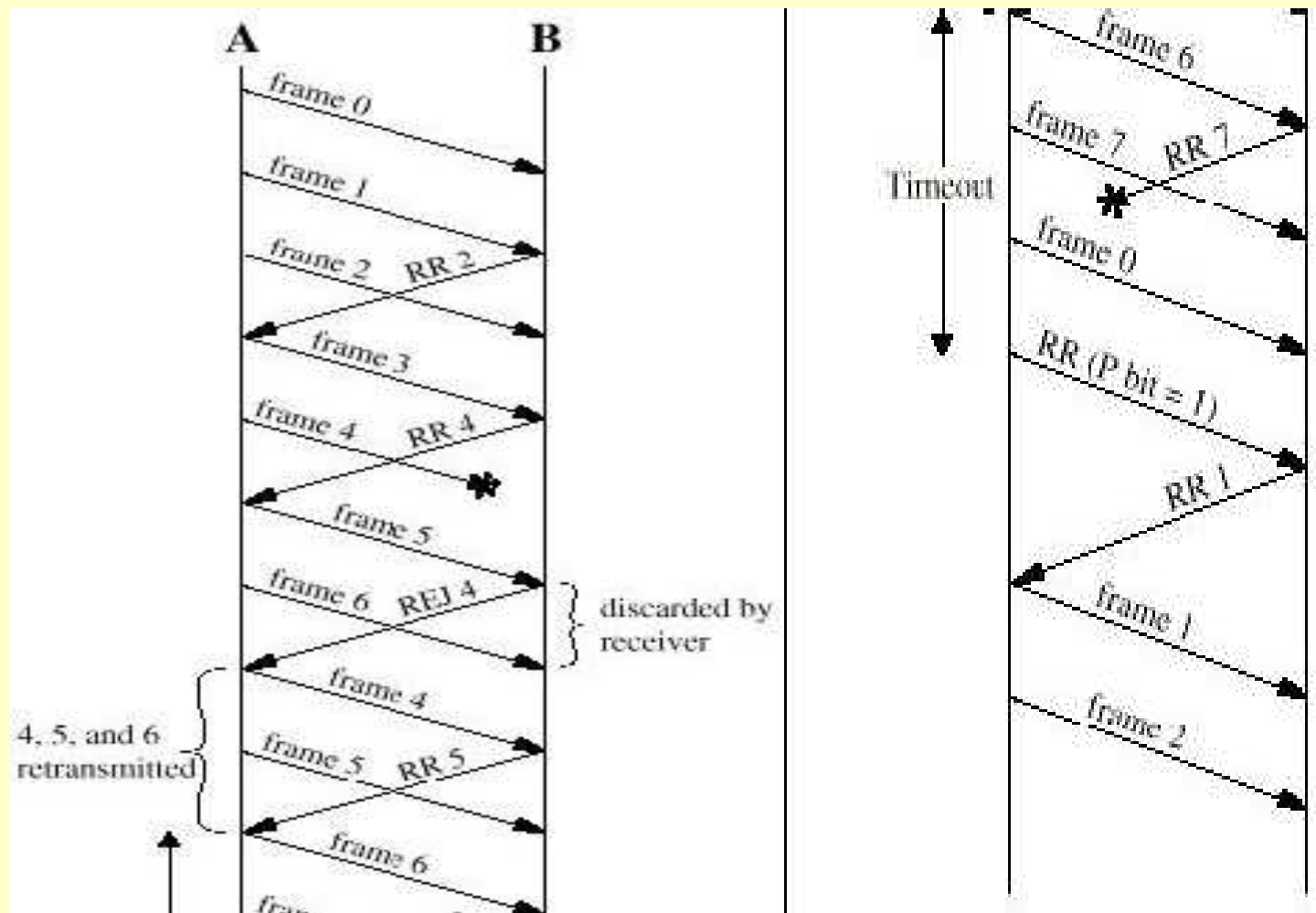
(d) Reject recovery



(e) Timeout recovery

O Protocolo HDLC

- Seqüências de conexão e desconexão

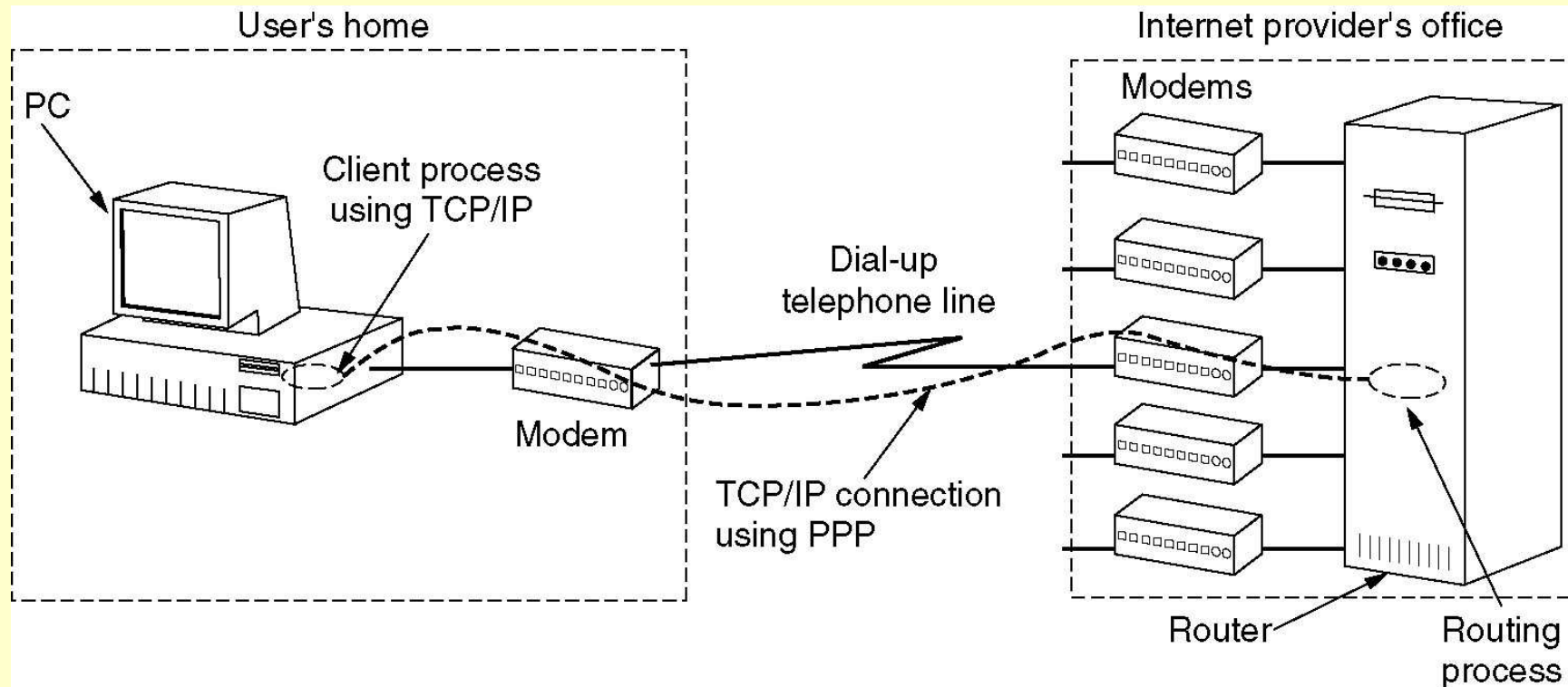




O Protocolo PPP

- A Internet requer um protocolo ponto a ponto para diversos fins, inclusive cuidar do tráfego de roteador para roteador e de usuário doméstico para ISP (provedor de serviços da Internet)
- Para isso, foi criado o protocolo PPP (Point-to-Point Protocol — protocolo ponto a ponto)
- Definido pelas RFC 1661, 1662 e 1663
- O PPP possui detecção de erros, suporta vários protocolos, permitindo que endereços IP sejam negociados
- Em tempo de conexão, permite a autenticação e muitas outras características

O Protocolo PPP



O estabelecimento de uma conexão à Internet através de um ISP



O Protocolo PPP

- O PPP dispõe de três recursos
 1. Um método de enquadramento que delimita, de forma não ambígua, o fim de um quadro e o início do quadro seguinte. O quadro também lida com a detecção de erros
 2. Um protocolo de controle de enlace usado para ativar linhas, testá-las, negociar opções e desativá-las novamente quando não forem mais necessárias. Esse protocolo é denominado LCP (*Link Control Protocol* — protocolo de controle de enlace). O LCP admite circuitos síncronos e assíncronos, e também codificações orientadas a bytes e a bits
 3. Uma maneira de negociar as opções da camada de rede de modo independente do protocolo da camada de rede a ser utilizado. O método escolhido deve ter um NCP (*Network Control Protocol* — protocolo de controle de rede) diferente para cada camada de rede suportada



O Protocolo PPP

- Considere a situação típica em que um usuário doméstico se conecta a um provedor de serviços da Internet, para transformar um PC doméstico em um host temporário da Internet.
 - Primeiro, o PC chama o roteador do provedor por meio de um modem. Depois que o modem do roteador atende ao telefone e estabelece uma conexão física, o PC envia ao roteador uma série de pacotes LCP no campo de carga útil de um ou mais quadros PPP.
 - Esses pacotes e suas respostas selecionam os parâmetros PPP a serem utilizados



O Protocolo PPP

- Quando todos esses parâmetros estão corretamente definidos de comum acordo, uma série de pacotes NCP é enviada para configurar a camada de rede
- Geralmente, o PC precisa executar uma pilha de protocolos TCP/IP, e assim, necessita primeiramente de um endereço IP
- Como não há endereços IP suficientes, normalmente, cada provedor da Internet possui um conjunto de endereços IP. Na conexão, atribui dinamicamente um endereço a cada PC recém-conectado durante sua sessão de login, através do protocolo DHCP
- Se tiver n endereços IP, um provedor poderia ter até n máquinas conectadas simultaneamente, mas sua base total de clientes poderá estar muito acima desse número. Para resolver o problema, o provedor normalmente atribui um endereço IP virtual ao Host
- Nesse momento, o PC passa a ser um host da Internet e pode enviar e receber pacotes IP, da mesma forma que os hosts fisicamente conectados. Quando o usuário termina, o NCP é utilizado para desativar a conexão da camada de rede e liberar o endereço IP.
- Em seguida, o LCP encerra a conexão da camada de enlace de dados.
- Finalmente, o computador solicita que o modem desligue o telefone, liberando a conexão da camada física.



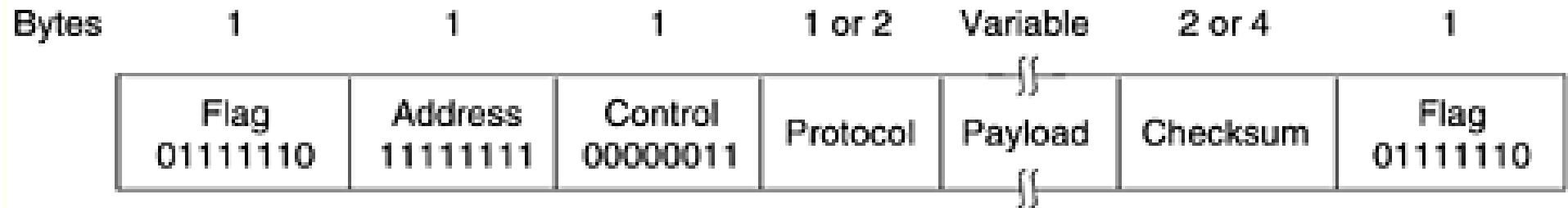
O Protocolo PPP

- O formato de quadro PPP foi definido de modo a ter uma aparência semelhante ao formato de quadro HDLC
- A principal diferença entre o PPP e o HDLC é que o primeiro é orientado a caracteres, e não a bits. Especificamente, o PPP utiliza a técnica de inserção de bytes, portanto, todos os quadros representam um número inteiro de bytes. Não é possível enviar um quadro formado por 30,25 bytes, como ocorre com o HDLC.
- Os quadros PPP não só podem ser enviados por linhas telefônicas de acesso por discagem, mas também podem ser enviados por linhas SONET ou por verdadeiras linhas HDLC orientadas a bits (por exemplo, para as conexões entre roteadores)



O Protocolo PPP

O quadro PPP no modo não numerado



- Flag - Delimitador do quadro
- Endereço - Não tem função, é de uso futuro
- Controle - Não tem função; no futuro é possível ter múltiplos campos de controle
- Protocolo - Indica o protocolo da camada superior ao qual o conteúdo do quadro deve ser entregue (ex. PPP-LCP, IP, IPCP, etc.)
- Info - Dados da camada superior, sendo transportados
- Checksum - Algoritmo de detecção de erros



O Protocolo PPP

- Todos os quadros PPP começam e terminam pelo byte de *flag* padrão do HDLC (01111110)
 - Que é complementado por inserção de bytes se este padrão ocorrer dentro do campo de carga útil. Utiliza o *byte stuffing*, usando o caractere Escape (01111101) antes do *flag* (ver *Byte Stuffing*)
- O campo Endereço sempre é definido com o valor binário 11111111
 - Indicando que todas as estações devem aceitar o quadro. A utilização desse valor evita o problema da necessidade de atribuição de endereços de enlace de dados.



O Protocolo PPP

- O campo Controle possui valor padrão 00000011
 - Esse valor indica um quadro **não numerado**
 - Em outras palavras, o PPP não oferece uma transmissão confiável, com o uso de números de sequência e confirmações como o padrão.
 - Em ambientes ruidosos, como em redes sem fio, pode ser utilizada a transmissão confiável, que emprega o **modo numerado**
 - Os detalhes exatos são definidos na RFC 1663, mas, na prática, raramente ele é utilizado
- Como os campos Endereço e Controle são sempre constantes na configuração padrão, o LCP fornece o mecanismo necessário para que as duas partes negociem uma opção que os omita totalmente e que economize 2 bytes por quadro

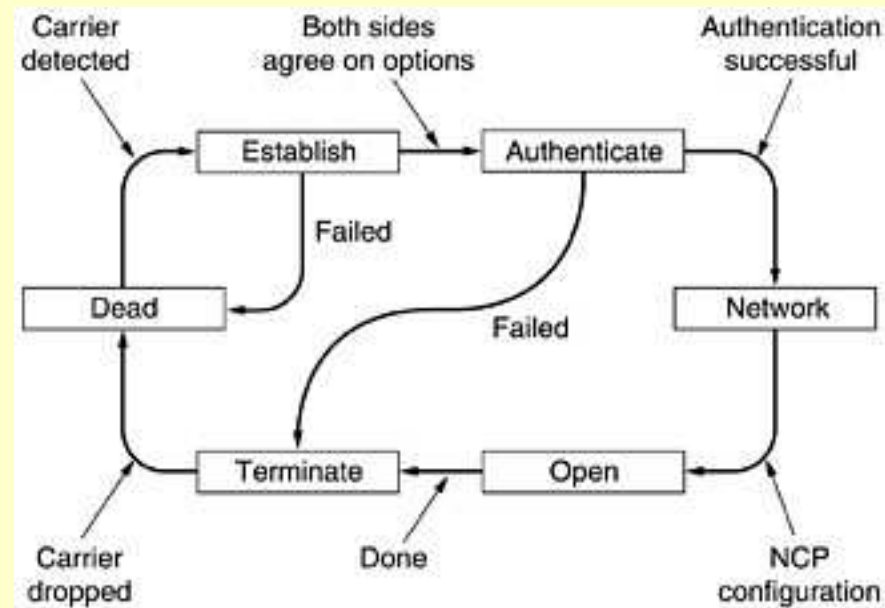


O Protocolo PPP

- O campo Protocolo
 - Informa o tipo de pacote que se encontra no campo Carga útil. Os códigos são definidos para representar os protocolos LCP, NCP, IP, IPX, AppleTalk e outros.
 - Os protocolos que começam por um bit 0 são os protocolos da camada de rede, como o IP, o IPX, o OSI, o CLNP, o XNS.
 - Aqueles que começam por um bit 1 são utilizados na negociação de outros protocolos. Entre eles, estão incluídos o LCP e um NCP diferente para cada protocolo da camada de rede admitido.
 - O tamanho padrão do campo Protocolo é de 2 bytes, mas é possível negociar uma redução para 1 byte, utilizando-se o LCP
- O campo Carga Útil (*Payload*)
 - Possui comprimento variável, podendo se estender até o tamanho máximo negociado. Se o comprimento não for negociado com o uso do LCP durante a configuração da linha, será empregado um comprimento padrão de 1.500 bytes. Poderá haver um preenchimento logo após a carga útil, caso seja necessário
- O Campo Checksum
 - Serve para detecção de erros, normalmente tem 2 bytes, embora seja possível negociar um checksum de 4 bytes



O Protocolo PPP



Funcionamento do Protocolo PPP (estados do protocolo)



O Protocolo PPP

- Funcionamento do PPP
 - O protocolo começa com a linha no estado *DEAD*, o que significa que não há nenhuma portadora da camada física presente e não existe qualquer conexão da camada física.
 - Depois de estabelecida a conexão física, a linha passa para a fase *ESTABLISH*. Nesse ponto, começa a negociação de opções do LCP que, se for bem-sucedida, leva à fase *AUTHENTICATE*
 - Agora, as duas partes poderão verificar suas identidades mutuamente, se desejarem
 - Quando a fase *NETWORK* é alcançada, o protocolo NCP apropriado é invocado para configurar a camada de rede. Se a configuração for bem sucedida, a fase *OPEN* é alcançada e o transporte de dados pode ser feito
 - Quando o transporte de dados é concluído, a linha entra na fase *TERMINATE* e, de lá, volta a *DEAD* quando a portadora é desativada



O Protocolo PPP

- Funcionamento do PPP (Continuação)
 - O LCP é utilizado para negociar as opções de protocolo de enlace de dados durante a fase *ESTABLISH*
 - Na verdade, ele não está preocupado com as opções propriamente ditas, mas com o mecanismo de negociação.
 - O protocolo LCP proporciona um meio para que o processo inicial faça uma proposta que será aceita ou rejeitada, total ou parcialmente, pelo processo de resposta.
 - Ele também permite que os dois processos testem a qualidade da linha, verificando se ela é boa o suficiente para estabelecer uma conexão.
 - Por fim, o protocolo LCP também permite que as linhas sejam desativadas quando não forem mais necessárias.



Referências

- [1] A S. Tanenbaum, **Redes de Computadores** - 3ª Edição Americana, Campus, Rio de Janeiro, 1994.
- [2] L.F. Soares, G. Lemos, e S. Colcher, **Redes de Computadores - Das LANs, MANs e WANs as Redes ATM**, Campus, Rio de Janeiro, 3ª Edição, 1995.
- [3] Pinheiro, Gil R. V., **Comunicação de Dados e Redes (para Automação Industrial)** - Petrobrás, junho/1997
- [4] Pinheiro, Gil R. V., **Técnicas de Detecção e Correção de Erros**, UERJ-FEN-DETEL
- [5] Kurose, James F.; Ross, Keith W. **Redes de Computadores e a Internet**, Editora Pearson