

# **Arquitetura de Redes de Computadores**

**Dr. Edson Moreira Silva Neto**

**Professor/Redes de Computadores**

**Camada de Enlace**

# Agenda

- 1) Introdução
- 2) Quadros
- 3) Enquadramento
- 4) Endereçamento
- 5) Detecção de Erro
- 6) Correção de Erro
- 7) Protocolo ARQ
- 8) Controle de Fluxo

# Introdução



# Introdução

- ◆ A principal função da **Camada de Enlace** é:
  - ◆ **GARANTIR** a comunicação entre dispositivos adjacentes.
- ◆ Enquanto a C-1 trabalha com bits, a C-2 trabalha com blocos de bits (quadros/frames).

# Introdução

- ◆ É função da C-2 (enlace):
  - ◆ Criar e interpretar corretamente os quadros;
  - ◆ Detectar possíveis erros e, quando necessário, corrigi-los.
  - ◆ Deve também controlar o fluxo de quadros que chegam ao destino, de forma a não sobrecarregar com um volume excessivo de dados.

# Quadros



# Estrutura de um quadro

Camada  
de rede

PDU de rede

Camada  
de enlace

Cabeçalho

Dados

CDE

# Quadros

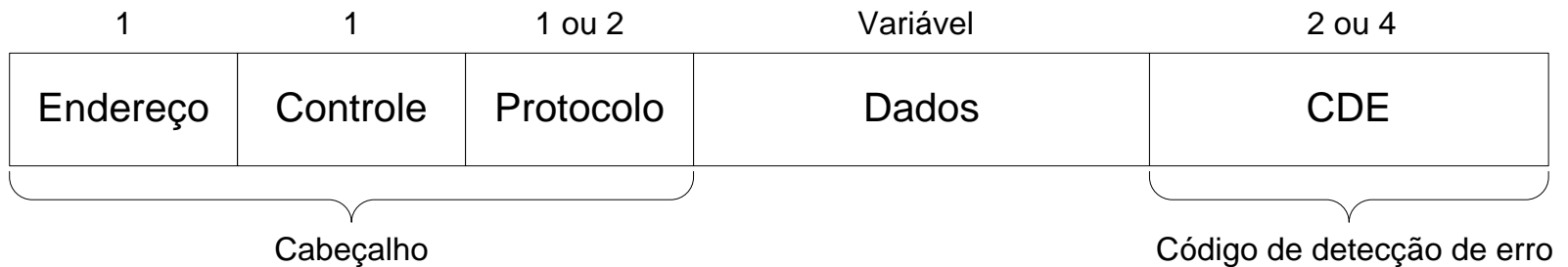
- ◆ O **cabeçalho** possui informações de controle para que haja a comunicação horizontal entre as camadas de enlace da origem e do destino.
  - ◆ É formado por diversos campos – cada um com uma função específica.
- ◆ O campo de **dados** encapsula a PDU de redes.
- ◆ O **código de detecção de erro** (CDE) tem a função de controlar erros na C-2.



# Quadro PPP

## *Point-to-Point Protocol*

💧 Consultar RFC-1661 e RFC-1662



# Quadros

- ◆ Pode ser formado por uma sequência de caracteres ou uma sequência de bits.
- ◆ Protocolos orientados a caractere
  - ◆ Ex: **BSC** (*Binary Synchronous Control*)
- ◆ Protocolos orientados a bit
  - ◆ Ex: **HDLC** (*High-Level Data Link Protocol*)

# Enquadramento



# Problema de enquadramento

- ◆ O receptor deve ser capaz de identificar o início e o final de cada bloco transmitido.
  - ◆ Enquadramento ou *framing*.

0 0 1 1 0 0 0 0 1 1 1 1 0 0 0 1 0 0 1 1 1 1 (a)

0 0 1 1 0 0 0 0	1 1 1 1 0	0 0 1 0 0 1 1 1 1
-----------------	-----------	-------------------

(b)

# Uso de delimitadores

Flag 01111110	Quadro	Flag 01111110	(a)
------------------	--------	------------------	-----

Flag 01111110	Endereço	Controle	Protocolo	Dados	CDE	Flag 01111110	(b)
------------------	----------	----------	-----------	-------	-----	------------------	-----

# Byte stuffing

Quadro  
original

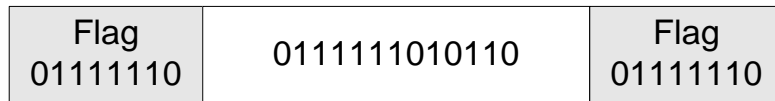
FI	FF ... FI ... FF ... CECE	FF
----	---------------------------	----

Quadro  
transmitido

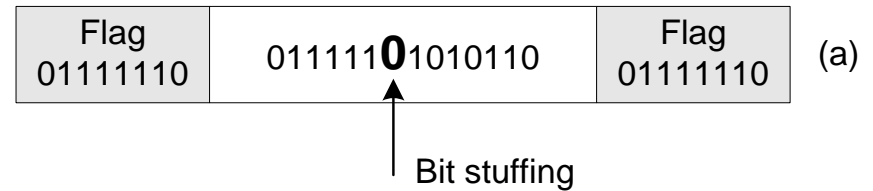
FI	<b>CEFF ... CEFI ... CEFF ... CECECECE</b>	FF
----	--	----

# Bit stuffing

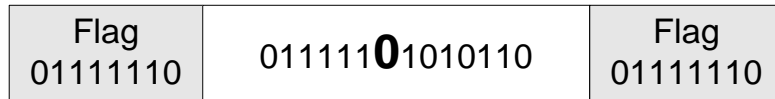
Quadro original



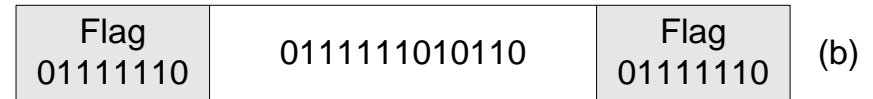
Quadro transmitido



Quadro recebido



Quadro original



# Enquadramento

- ◆ Em protocolos orientados a caractere é possível implementar um esquema alternativo de enquadramento, que leva em consideração o tamanho do quadro.
  - ◆ O Cabeçalho teria um campo que indica o número de bytes que compõe o restante do quadro.
    - ◆ Não há necessidade de delimitador de término, apenas início.
    - ◆ Não necessita implementar o STUFFING.
- ◆ Outra técnica: usando a sinalização.
  - ◆ Ex: Ethernet de 10Mbps. O término do quadro é identificado pela ausência de sinal no meio.



# Endereçamento



# Endereçamento

- ◆ Está associado à identificação da interface de comunicação, que conecta o dispositivo à rede.
- ◆ Cada interface possui um endereço único que permite identificá-la.
- ◆ O formato do endereço é definido pelo protocolo de enlace e deve ser seguido pelos fabricantes de interface de rede.

# Quadro Ethernet

💧 Versão simplificada do quadro Ethernet.

6	6	2	0-1500	4
End. destino	End. origem	Tam	Dados	CDE

# Exemplo de endereçamento

```
c:\>ipconfig/all
```

```
Adaptador Ethernet Conexão local:
```

```
Descrição . . . . . : Realtek RTL8169/8110 Family Gigabit Ethernet NIC  
Endereço físico . . . . . : 00-08-54-46-4B-FA
```

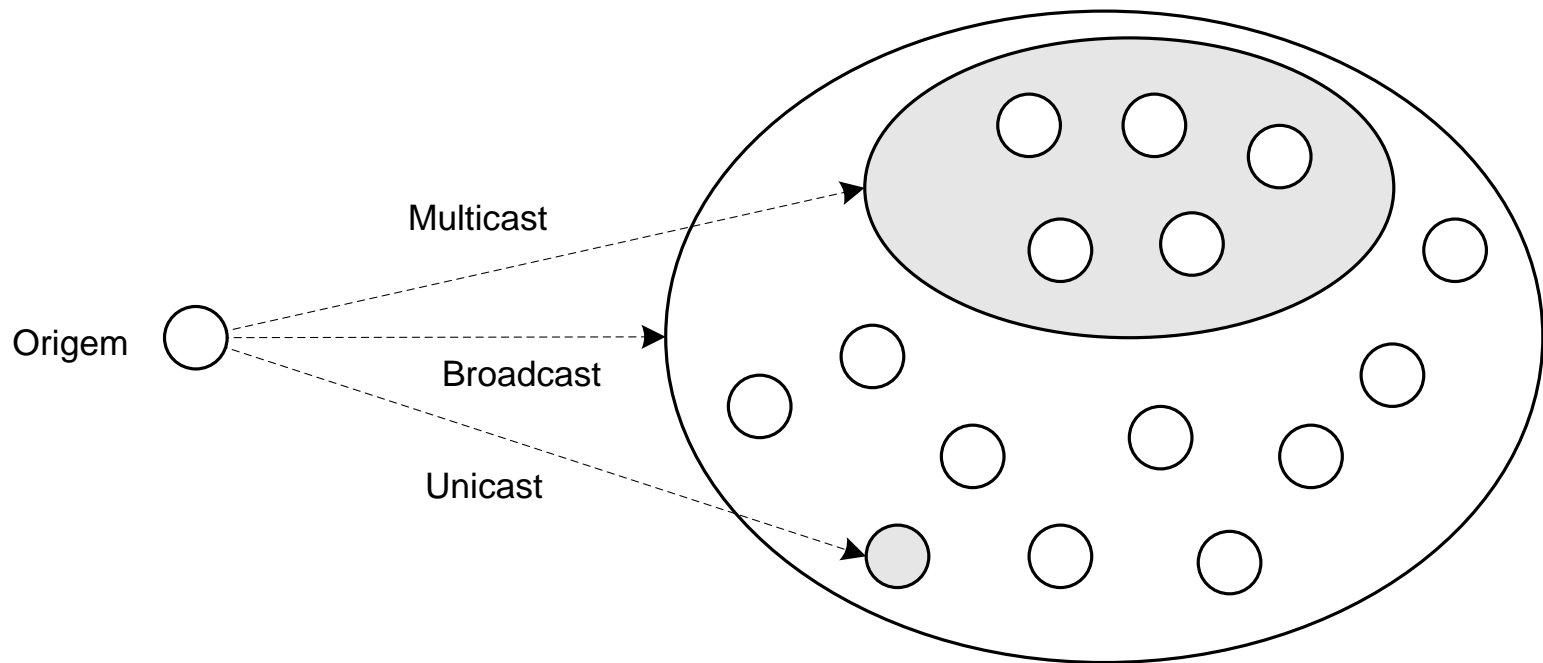
```
# ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 00:02:B3:B1:96:09  
          inet  addr:10.10.0.50  Bcast:10.10.255.255  Mask:255.255.0.0
```

# Endereçamento

- Existem basicamente 3 formas de endereçamento implementadas por uma rede:
  - Unicast
  - Broadcast (ou difusão)
  - Multicast (multidifusão)

# Endereçamentos unicast, multicast e broadcast



# Detecção de Erro

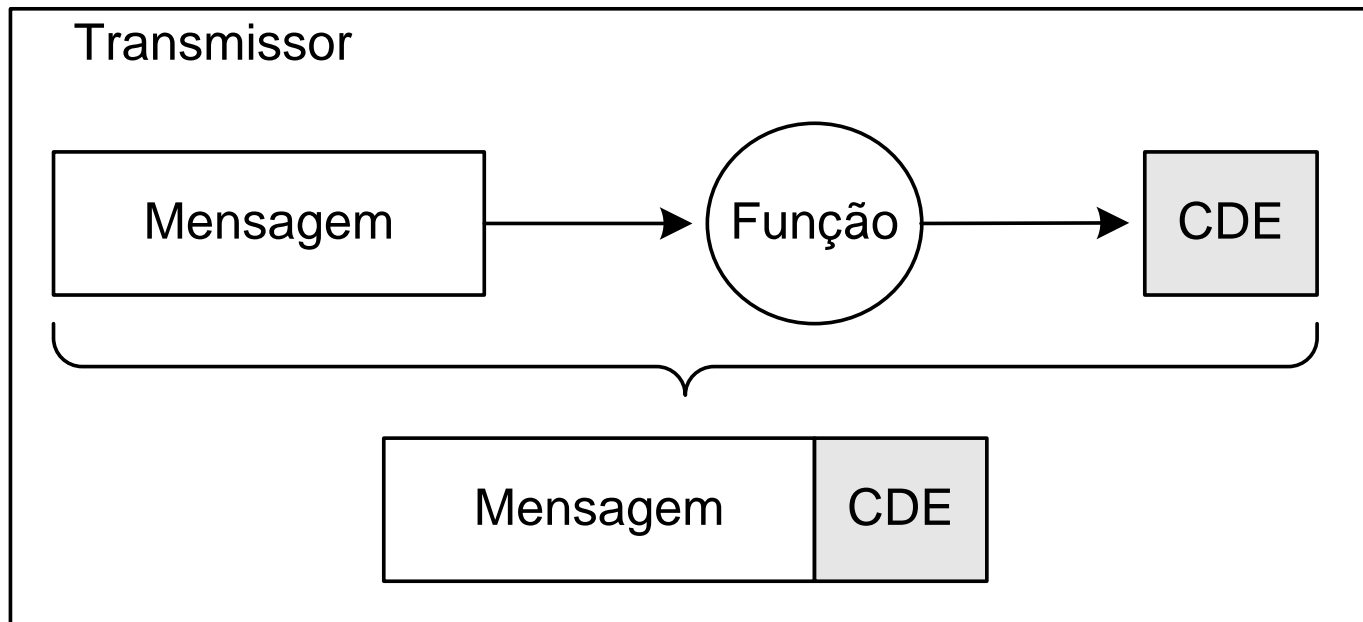


# Detecção de Erro

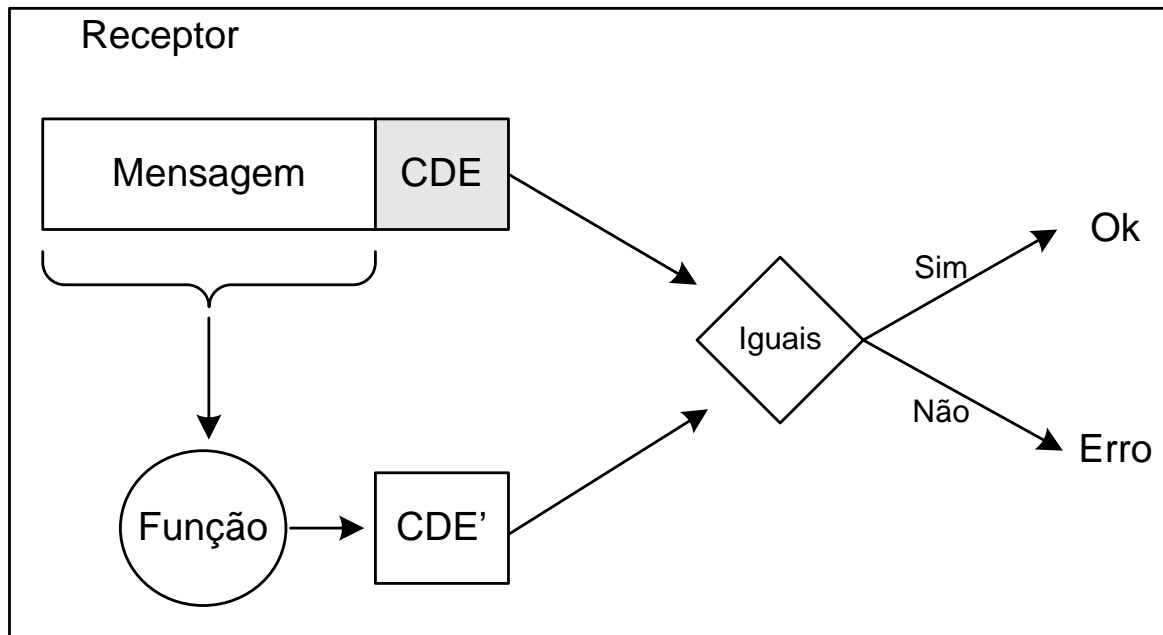
- Qualquer transmissão está sujeita a problemas, como ruídos e atenuação, e a camada de enlace tem a função de realizar o tratamento dos possíveis erros.
- Controle de Erros – 2 Etapas distintas:
  - Detecção dos possíveis erros nos dados transmitidos
  - Correção dos erros encontrados



# Geração do código de detecção de erro



# Verificação do código de detecção de erro



# Detecção de Erro

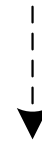
- ◆ Duas técnicas amplamente utilizadas:
  - ◆ Bit de Paridade
    - ◆ Paridade Simples (PAR ou ÍMPAR)
    - ◆ Paridade Múltipla
  - ◆ Verificação de Redundância Cíclica (VRC)
    - ◆ Ou **CRC** (*Cyclic Redundancy Check*)

# Exemplos de bit de paridade

<b>Caractere</b>	<b>Paridade par</b>	<b>Paridade ímpar</b>
1011010	10110100	10110101
0000001	00000011	00000010

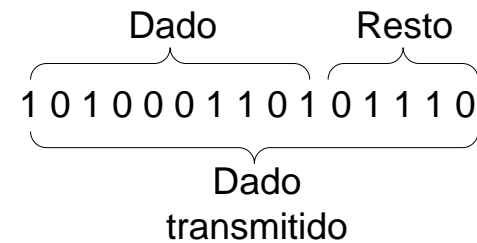
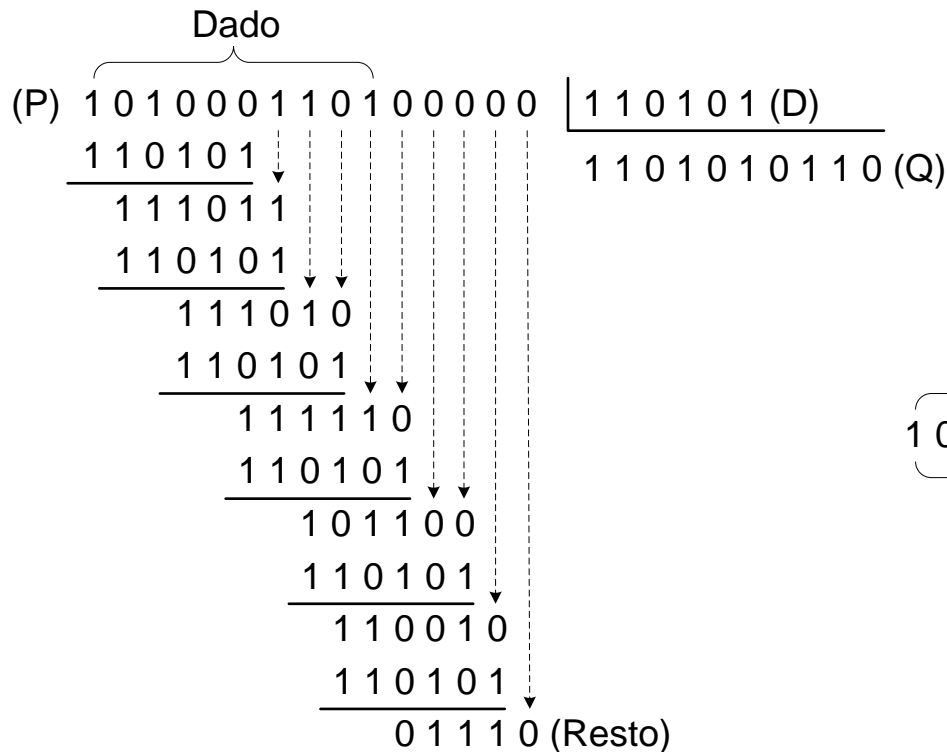
# Paridade múltipla

PS



C1	---->	0110010	0
C2	---->	1000100	1
C3	---->	0011011	1
PM	---->	0010010	1

# Cálculo do CRC



# Exemplos de polinômios geradores

Nome	Polinômio
CRC-12	$x^{12}+x^{11}+x^3+x^2+1$
CRC-CCITT	$x^{16}+x^{12}+x^5+1$
CRC-16	$x^{16}+x^{15}+x^2+1$
CRC-32	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

# Correção de Erro

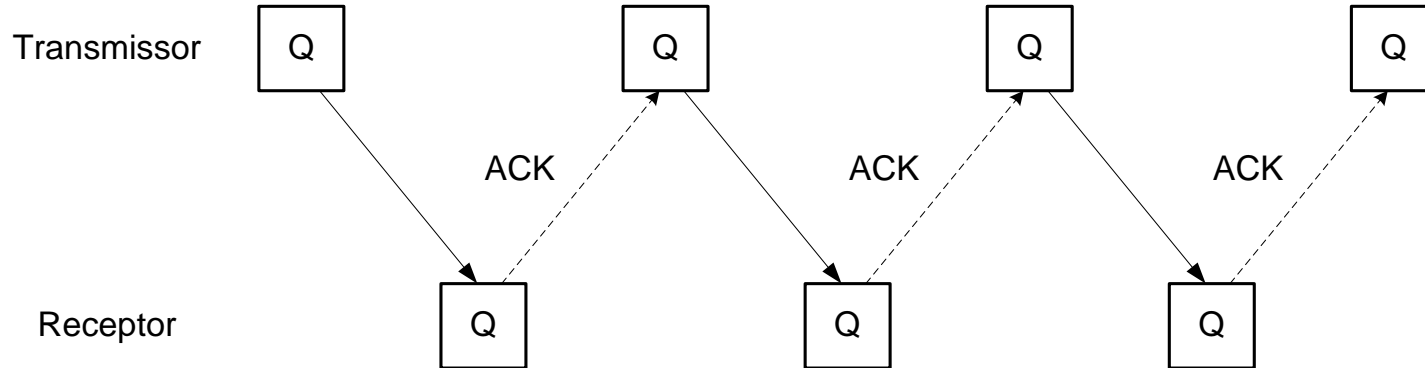




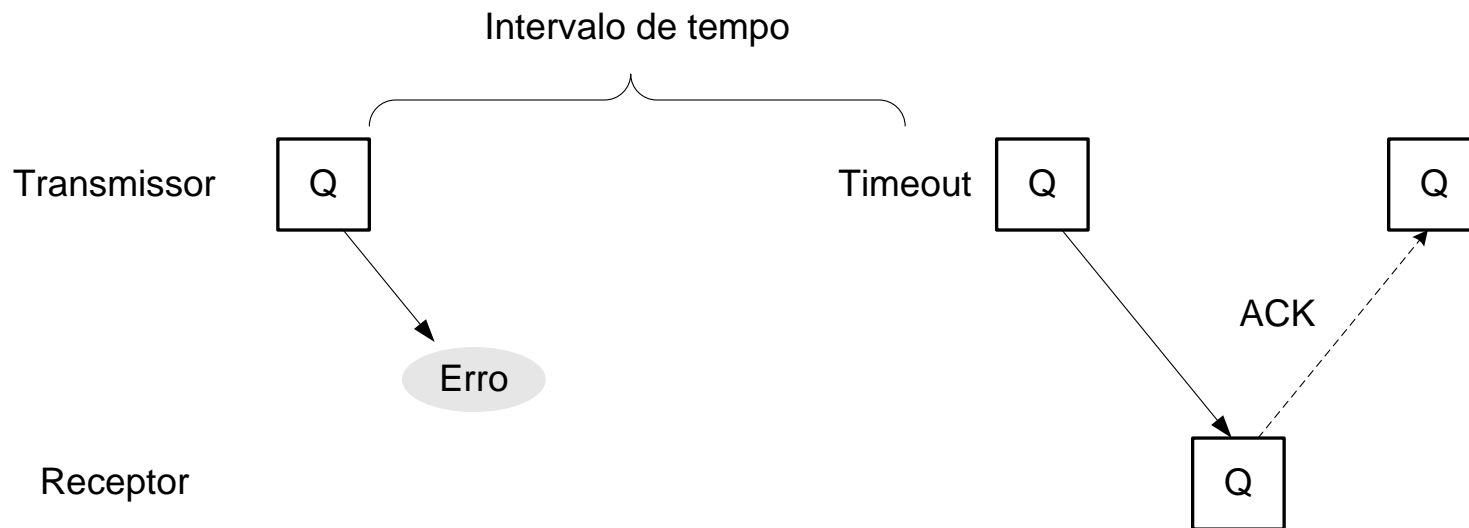
# Correção de Erro

- ◆ Nem sempre a C-2 implementa algum mecanismo de correção de erro
  - ◆ Neste caso, ela ignora o erro
  - ◆ A correção fica para as camadas superiores
- ◆ A correção de erro na C-2 está muito relacionada ao tipo do meio de transmissão utilizado e à sua taxa de erro
  - ◆ FO *versus* Redes Wireless

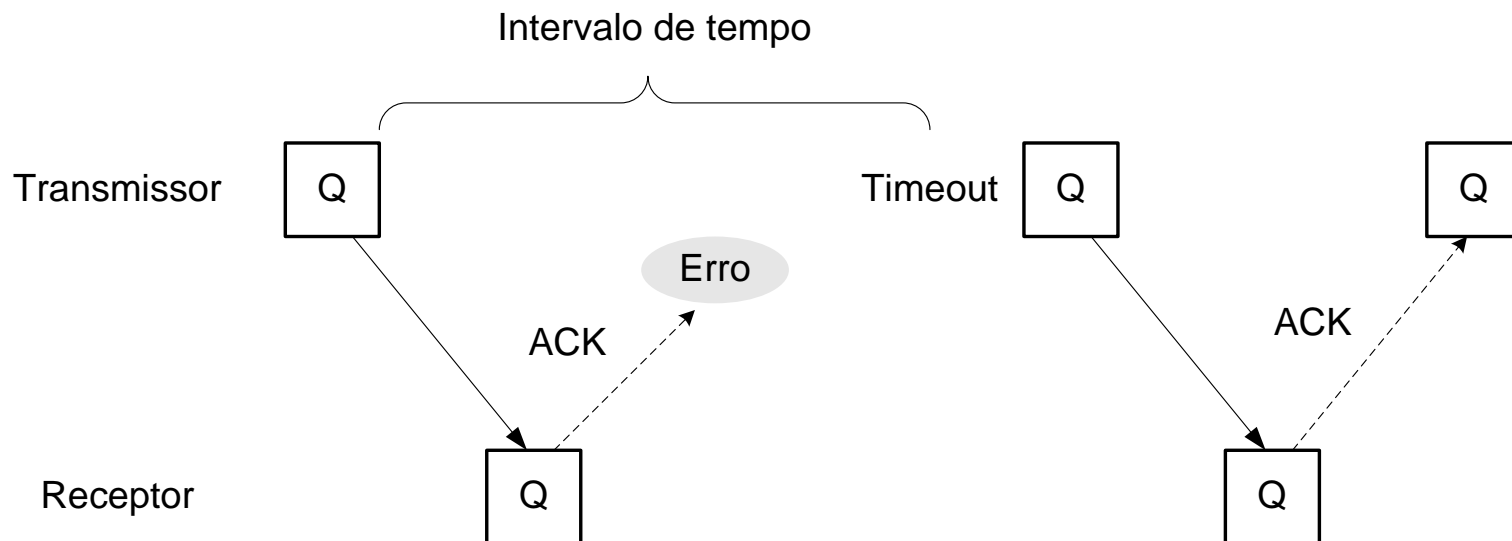
# Reconhecimento positivo



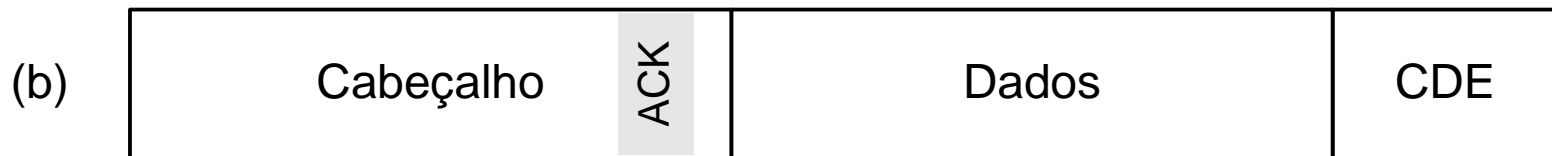
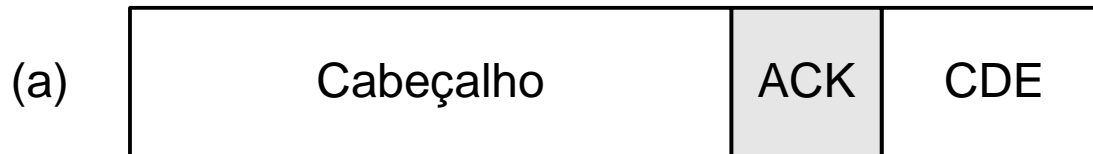
# Retransmissão por timeout



# Retransmissão por timeout



# Implementação do ACK



# Quadro chega ao destino com erro

- 2 estratégias:

- Descartar o quadro e aguardar que ocorra um *timeout* no transmissor e ele reenvie o quadro
  - Uma variante seria enviar um NAK (reconhecimento negativo com retransmissão)
  - Se o temporizador expirar antes do recebimento do NAK, esta técnica é inócua
- FEC (*Forwarding Error Correction*)
  - Os bits do quadro são corrigidos no próprio destino

# Protocolos ARQ



# Protocolos ARQ

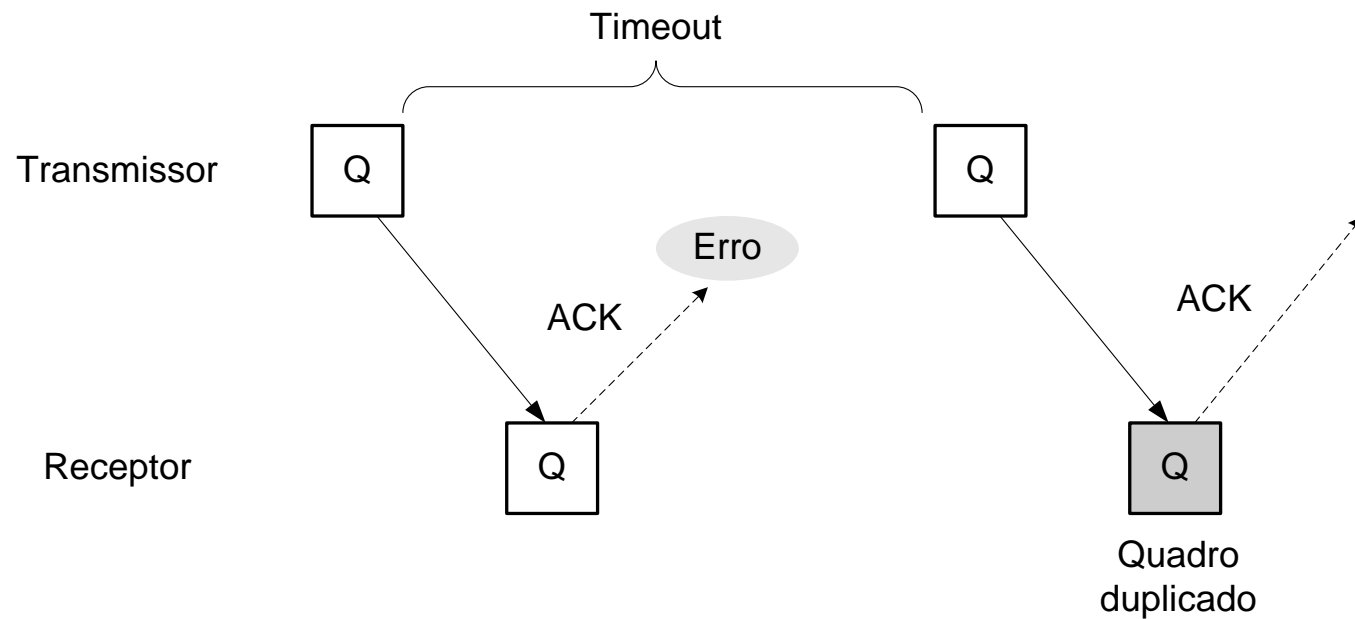
- Os mecanismos que utilizam o reconhecimento e a retransmissão de quadros como mecanismos para a correção de erro são chamados genericamente de **Protocolos ARQ** (*Automatic Repeat Request*).
- 3 Implementações:
  - Bit alternado
  - Retransmissão Integral
  - Retransmissão Seletiva



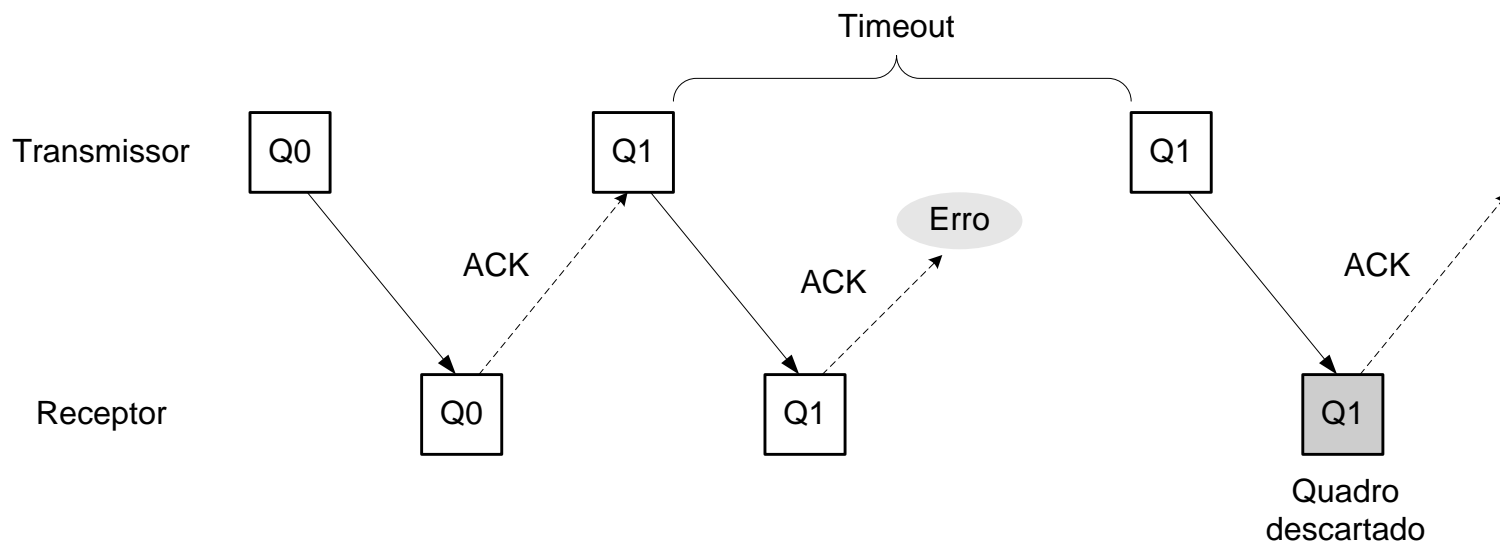
# Bit Alternado

- ◆ Para cada quadro enviado o transmissor deve aguardar o seu reconhecimento antes de enviar o próximo quadro.
- ◆ Um protocolo simples
- ◆ 2 problemas:
  - ◆ Um relacionado à duplicação de quadros de dados
  - ◆ Outro relacionado à subutilização do canal de comunicação

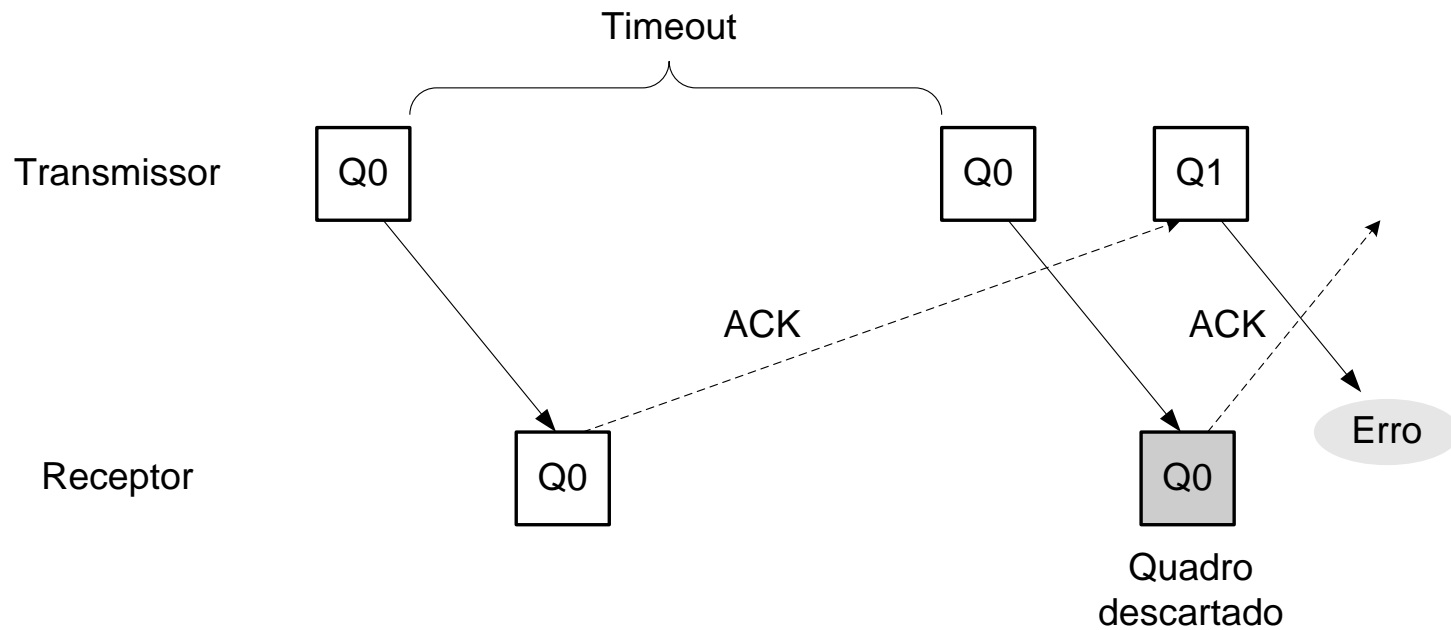
# Problema com a duplicação de quadros



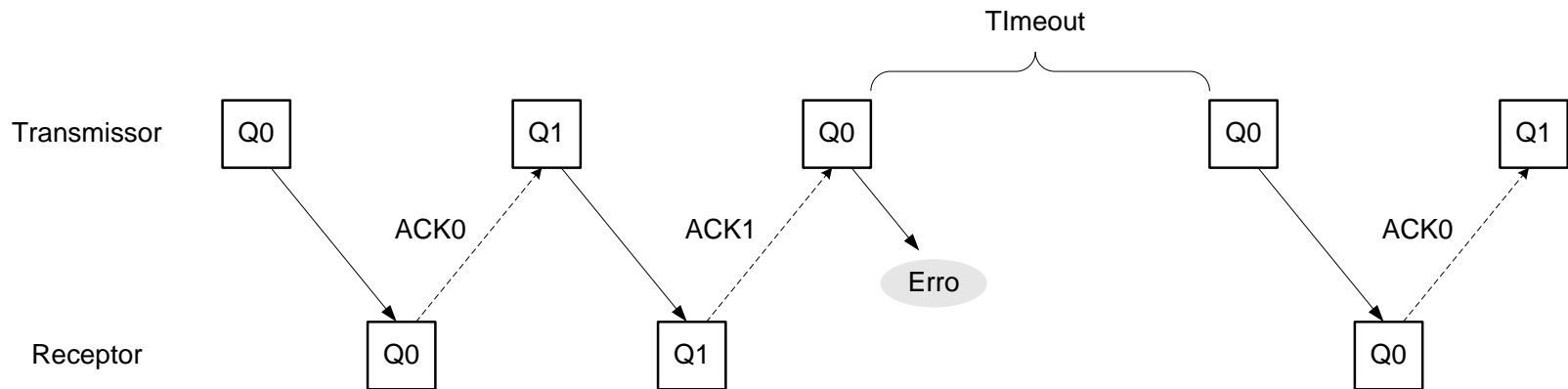
# Numeração dos quadros



# Problema com ACK duplicado



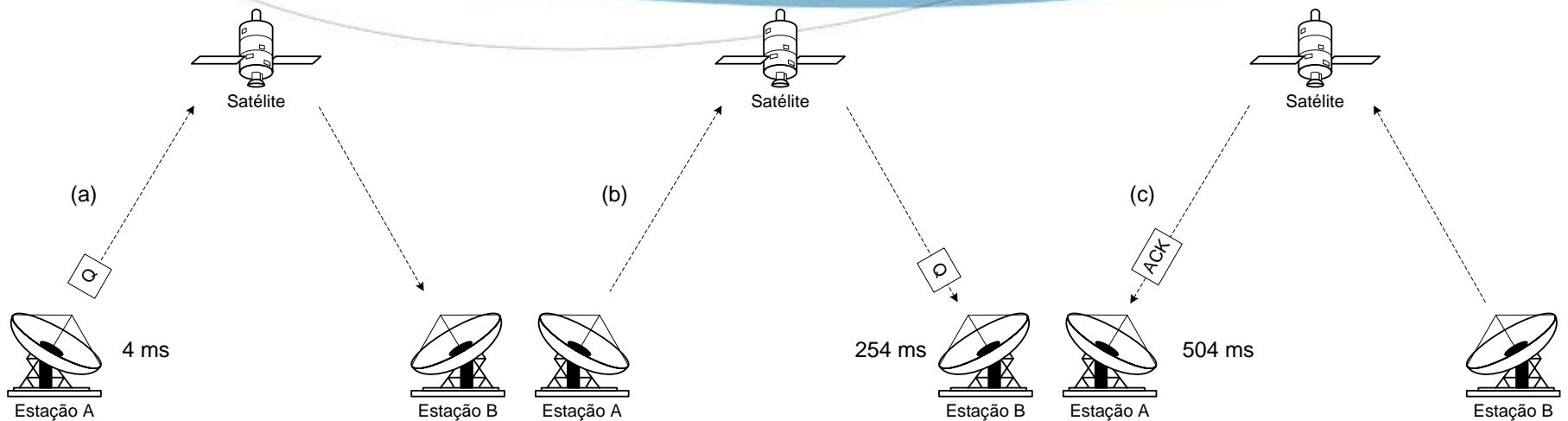
# Protocolo de bit alternado



# Protocolo Bit Alternado

- Apesar da simplicidade, é bastante ineficiente. Dependendo da distância entre os dispositivos, da taxa de transmissão e do tamanho do quadro, a transmissão de um único frame por vez pode gerar uma enorme **subutilização do canal de comunicação**.

# Transmissão de um quadro via satélite



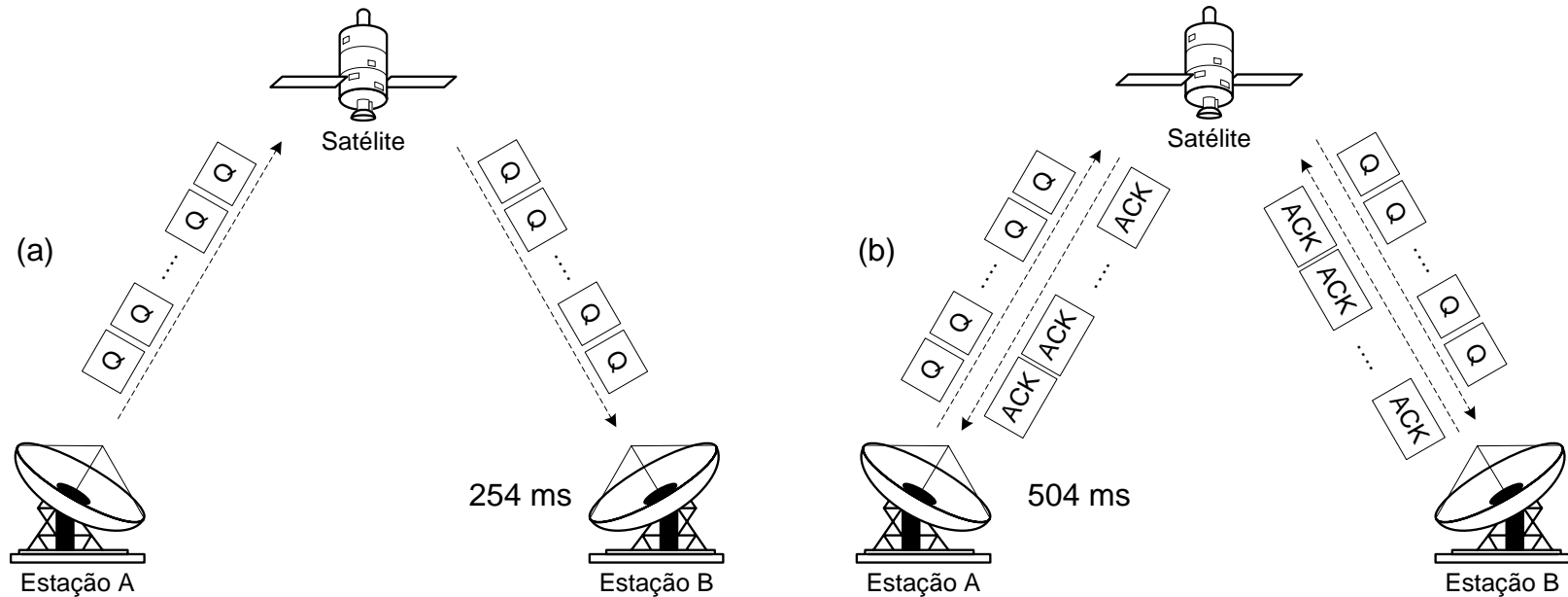
$$\begin{aligned} \text{TUM} &= 4 \text{ ms} / 504 \text{ ms} \\ \text{TUM} &\approx 0,8\% \end{aligned}$$

# Retransmissão Integral

- ◆ Para melhorar a utilização do canal de comunicação, vários quadros podem ser transmitidos sem a necessidade de aguardar o reconhecimento de cada frame individualmente.
- ◆ No exemplo da transmissão via satélite poderíamos transmitir até 125 quadros em sequência até a chegada do primeiro ACK.



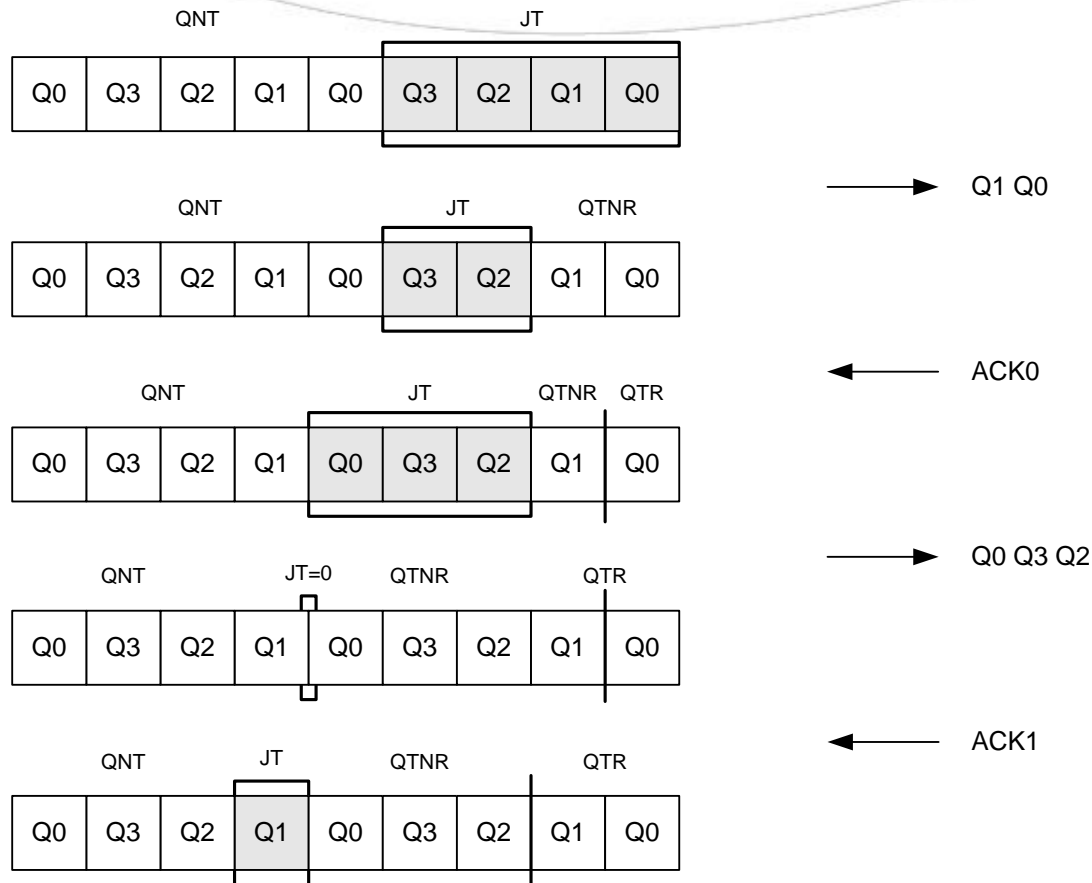
# Transmissão de uma sequência de quadros



# Janela Deslizante

- ◆ Ou *sliding window*
  - ◆ Permite que as camadas de enlace da origem e destino tenham mais flexibilidade para enviar e receber quadros.
- ◆ Na origem é utilizada uma JANELA DE TRANSMISSÃO que permite controlar os quadros que podem ser transmitidos de forma a maximizar o uso do canal de comunicação.
- ◆ O tamanho da janela é uma potência de 2 e os quadros são numerados de **zero** a  $2^n - 1$ ;  $n$  = tamanho máximo da janela.

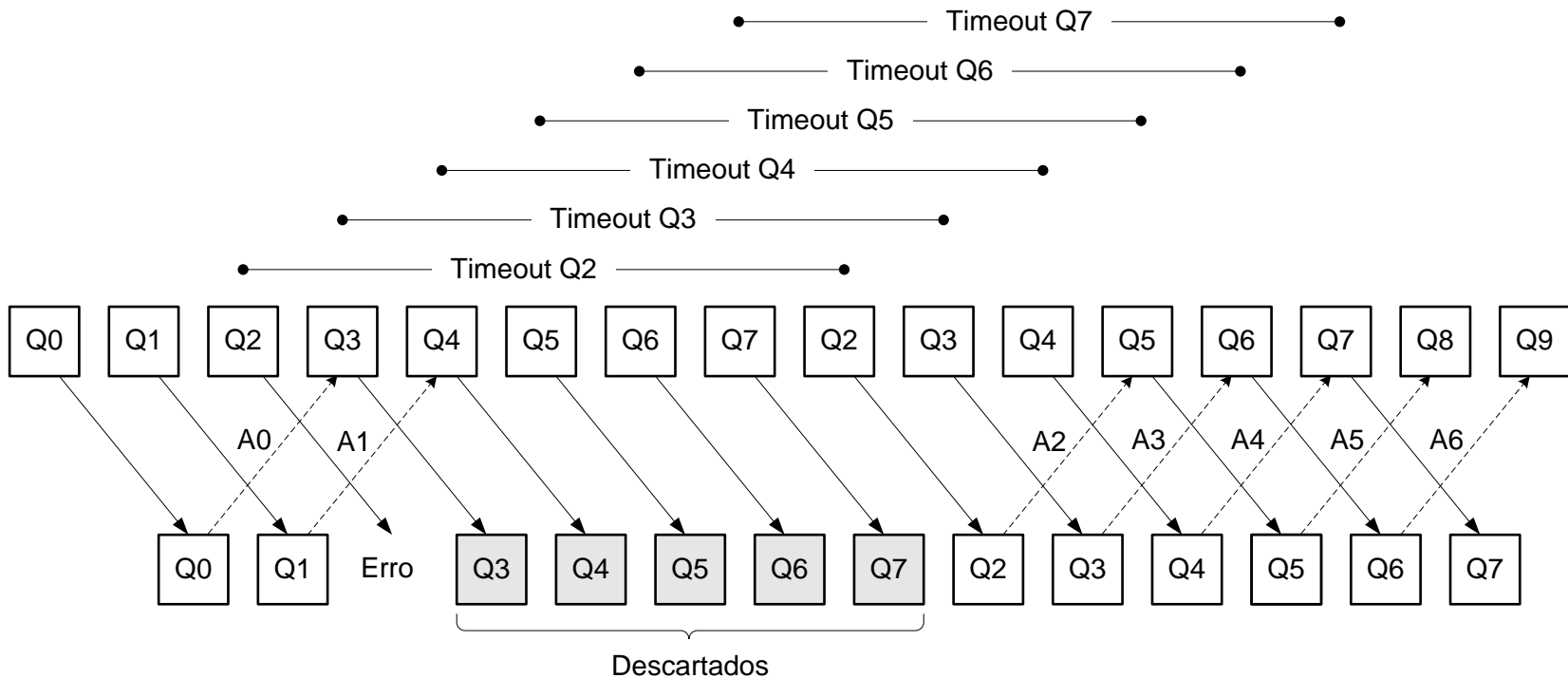
# Janela de transmissão



## OBSERVAÇÕES:

- JT = 4
- Quadros numerados de 0 a 3
- QNT  
*Quadros Não Transmitidos*
- QNR  
*Q. Transmitidos e Não Reconhecidos*
- QTR  
*Q. Transmitidos e Reconhecidos*

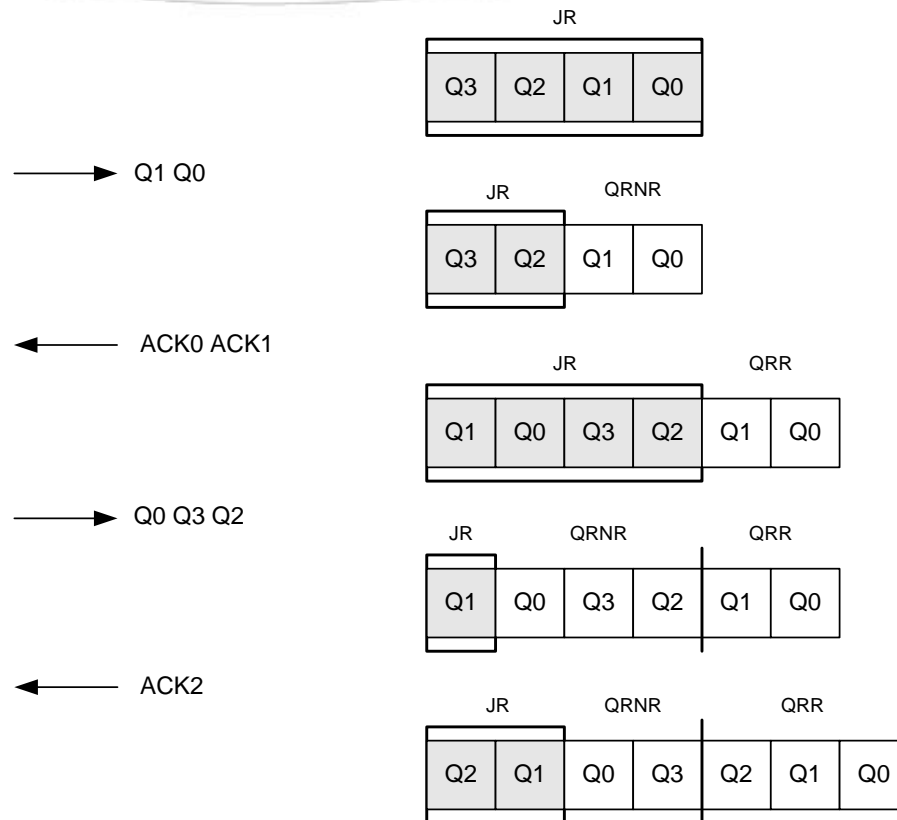
# Retransmissão integral



# Retransmissão Seletiva

- ◆ Neste caso, um problema em um quadro não implica a retransmissão integral dos frames já transmitidos e não reconhecidos, como no protocolo anterior.
- ◆ A retransmissão seletiva implementa uma JANELA DE RECEPÇÃO que permite controlar os quadros que ainda podem ser recebidos
  - ◆ A janela define a capacidade máxima de quadros que a camada de enlace do destino pode gerenciar.
- ◆ O tamanho da janela é uma potência de 2 e os quadros são numerados de **zero** a  $2^n - 1$ ;  $n$  = tamanho máximo da janela.

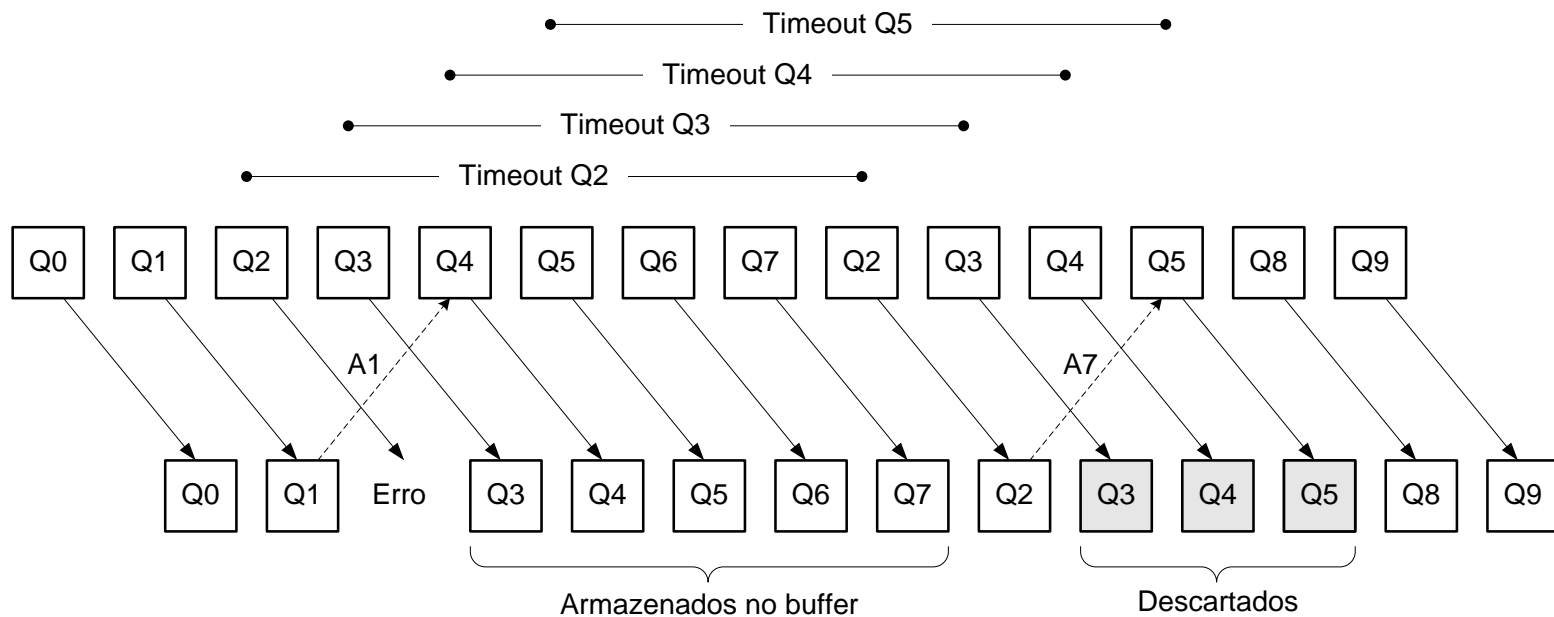
# Janela de recepção



# Janela de recepção

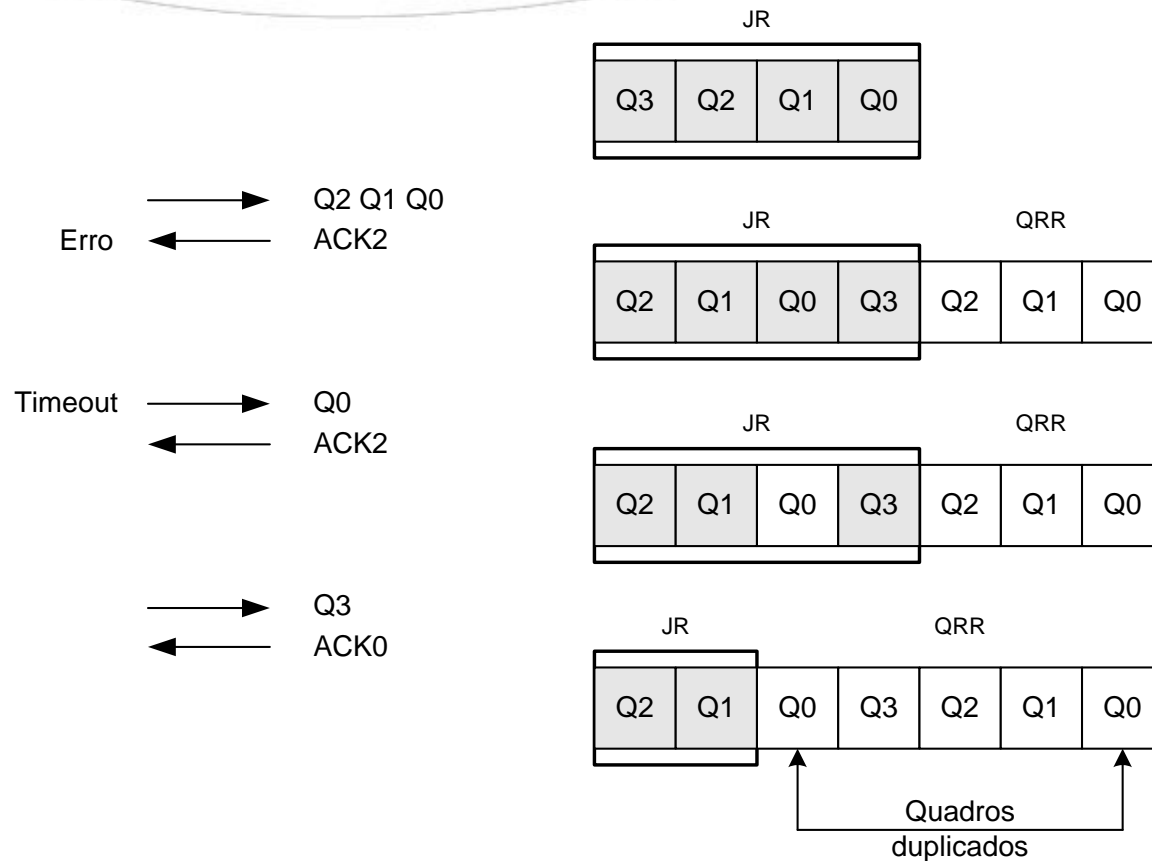
- ◆ Uma técnica usada para melhorar o funcionamento do protocolo de retransmissão seletiva é conhecida como *reconhecimento cumulativo*.
- ◆ Neste caso, é enviado o reconhecimento do último quadro recebido corretamente. Ficando implícito o reconhecimento dos quadros anteriores.

# Retransmissão seletiva





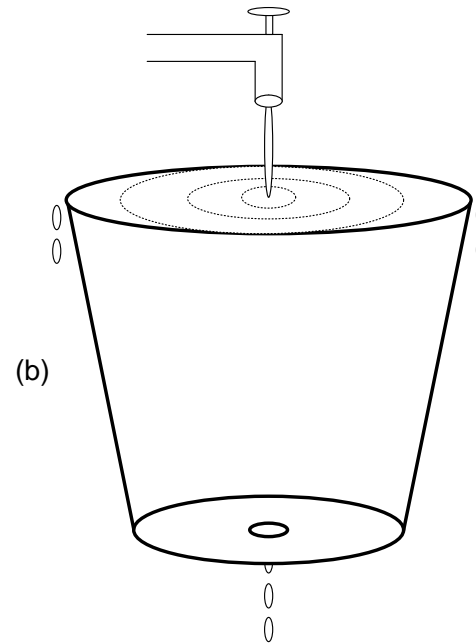
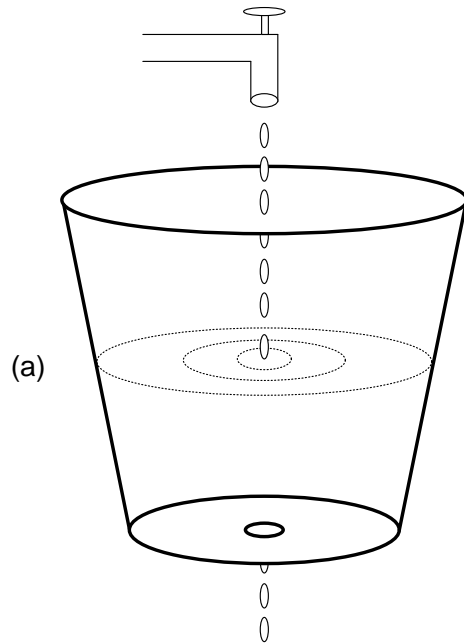
# Problema de sobreposição na janela de recepção



# Controle de Fluxo



# Problema de controle de fluxo



# Problema de controle de fluxo

- ◆ O **CONTROLE DE FLUXO** permite que o dispositivo transmissor regule o volume de dados enviados de forma a não gerar um *overflow* (transbordamento) no receptor.
- ◆ Caso contrário, o destino terá que descartar os dados transmitidos, obrigado à sua retransmissão.

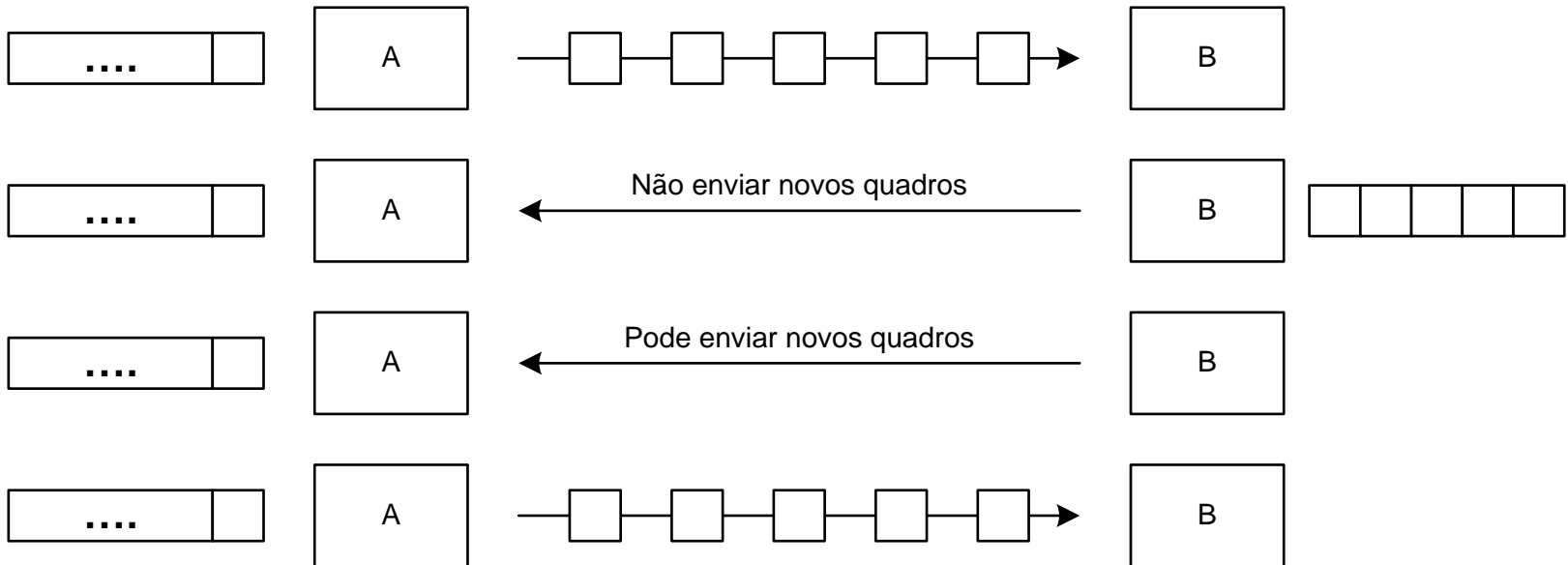
# Problema de controle de fluxo

- ◆ A forma mais simples de controle de fluxo é chamada de *stop-and-wait* (pare e espere).
- ◆ O transmissor, após enviar um quadro, aguarda pela confirmação do recebimento para poder enviar o próximo.
  - ◆ O problema do *overflow* fica eliminado.
  - ◆ Apresenta o mesmo problema de subutilização do canal apresentado pelo bit alternado.

# Problema de controle de fluxo

- ◆ Uma outra técnica bastante utilizada pelos protocolos da camada de enlace é fazer com que o receptor envie mensagens para o transmissor informando se está apto ou não a receber novos quadros.

# Controle de fluxo



# Controle de fluxo

- ◆ O controle de fluxo é geralmente resolvido com o receptor informando ao transmissor sua capacidade de recepção de dados (tamanho do buffer de recepção).
- ◆ Os protocolos que utilizam a técnica de janela deslizante podem facilmente implementar a função de controle de fluxo apenas ajustando o tamanho da janela de transmissão.



# Controle de fluxo

- ◆ Outra forma:
  - ◆ O receptor informa o tamanho de sua janela de recepção a cada reconhecimento.
  - ◆ Se a janela de recepção ficar vazia (i.e, sem espaço para novos quadros), o receptor informa ao transmissor uma janela de recepção de tamanho zero. O transmissor para de enviar quadros.
  - ◆ E, voltará a transmitir apenas quando receber uma janela maior que zero.