

PTC 2550 - Aula 06

5.1 Introdução à camada de enlace

5.2 Detecção, correção de erros

5.3 Protocolos de acesso múltiplo

(Kurose, p. 321 - 333)

(Peterson, p. 57-79)

29/03/2017

Capítulo 2

Camada de enlace

Capítulo 2: Camada de enlace

nossos objetivos:

- ❖ entender princípios por trás dos serviços da camada de enlace:
 - detecção, correção de erros
 - compartilhamento de um canal *broadcast*: múltiplo acesso
 - endereçamento na camada de enlace
 - redes de área local: Ethernet, VLANs
- ❖ exemplificação, implementação de várias tecnologias da camada de enlace

Camada de enlace, LANs: estrutura

2.1 introdução, serviços

2.2 detecção, correção de erros

2.3 protocolos de múltiplo acesso

2.4 LANs

- endereçamento, ARP
- Ethernet
- *switches*
- VLANs

2.5 virtualização de enlace: MPLS

2.6 redes de *data centers*

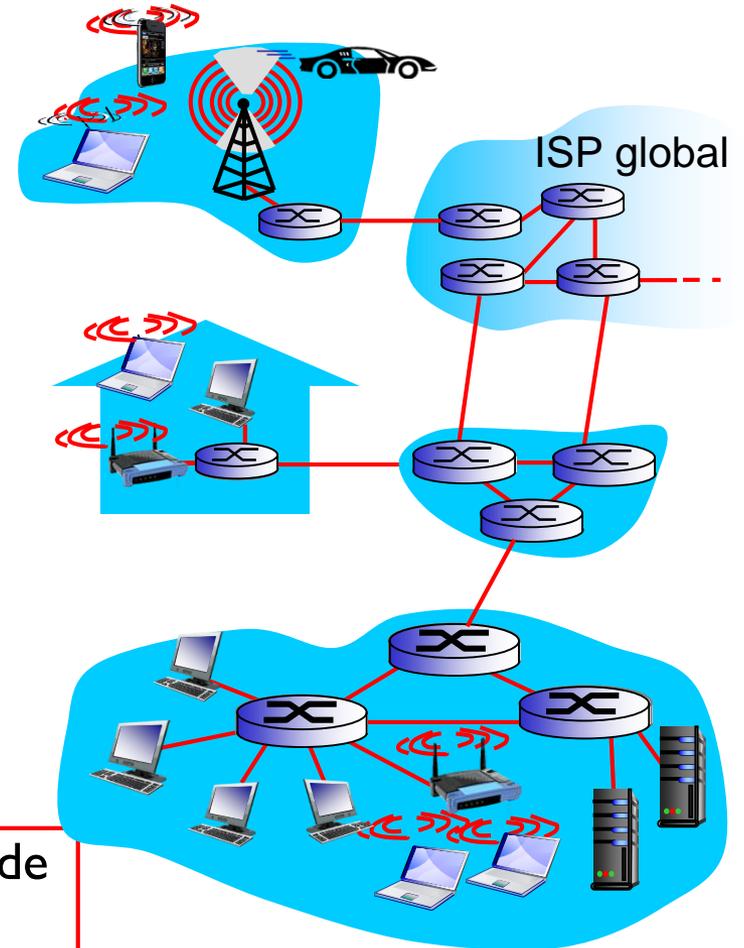
2.7 resumo: como funciona um pedido *web*

Camada de enlace: introdução

terminologia:

- ❖ **nós:** dispositivos que rodam protocolos da camada de enlace (camada 2) - hosts, roteadores, switches, pontos de acesso WiFi, etc.
- ❖ **enlaces:** canais de comunicação que conectam nós adjacentes ao longo do caminho de comunicação
 - ❖ enlaces com fio
 - enlaces sem fios
 - LANs
- ❖ **quadro:** pacote da camada-2 - encapsula datagrama

camada de enlace de dados tem a responsabilidade de transferir datagramas de um nó a um nó *fisicamente adjacente* através de um enlace



Camada de enlace: contexto

- ❖ datagrama transferido por diferentes protocolos de enlace sobre diferentes enlaces:
 - e.g., Ethernet no primeiro enlace, *frame relay* sobre enlaces intermediários, 802.11 no último enlace
- ❖ cada protocolo de enlace provê diferentes serviços
 - e.g., pode ou não prover transporte de dados confiável sobre enlace

analogia com transporte:

- ❖ viagem de São Paulo a Saint-Malo
 - táxi: São Paulo a GRU
 - avião: GRU a Paris
 - trem: Paris a Saint-Malo
- ❖ turista = **datagrama**
- ❖ trecho de transporte = **enlace de comunicação**
- ❖ modo de transporte = **protocolo da camada de enlace**
- ❖ agente de viagem = **algoritmo de roteamento**

Serviços da camada de enlace

❖ *enquadramento, acesso ao enlace:*

- encapsulamento do datagrama em quadro, adicionando cabeçalho, *trailer*
- acesso a canal se meio compartilhado
- endereços “MAC” (*Medium Access Control*) usados nos cabeçalhos de quadro para identificar fonte e destino
 - diferente dos endereços IP!

❖ *entrega confiável entre nós adjacentes:*

- já aprendemos como fazer isso (capítulo 3)!
- raramente usado em enlaces com baixa taxa de error de bit (BER) (fibra, alguns pares trançados)
- enlaces sem fio: altas taxas de erro
 - **Q:** por que tanto confiabilidade em nível de enlace quanto fim-a-fim?

Serviços da camada de enlace (mais)

❖ *controle de fluxo:*

- ritmo entre nós adjacentes enviando e recebendo

❖ *detecção de erro:*

- erros causados por atenuação do sinal, ruído.
- receptor detecta presença de erros:
 - solicita retransmissão ou descarta quadro

❖ *correção de erros:*

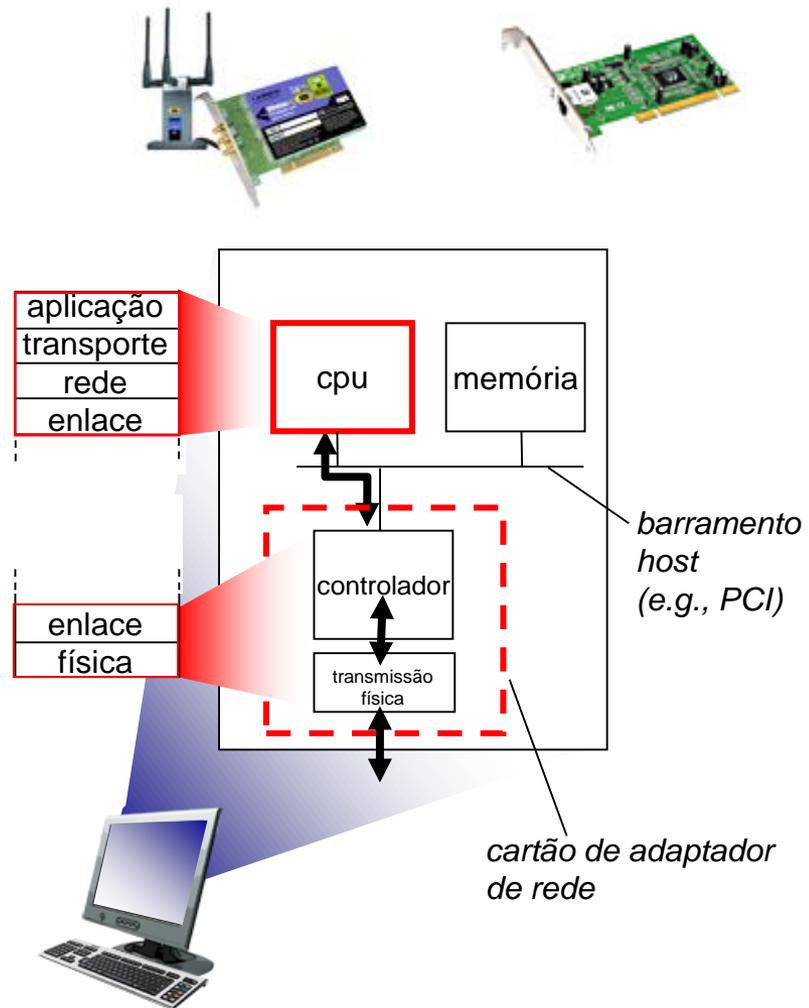
- receptor identifica *e corrige* erro(s) em bit(s) sem necessitar de retransmissão

❖ *half-duplex e full-duplex*

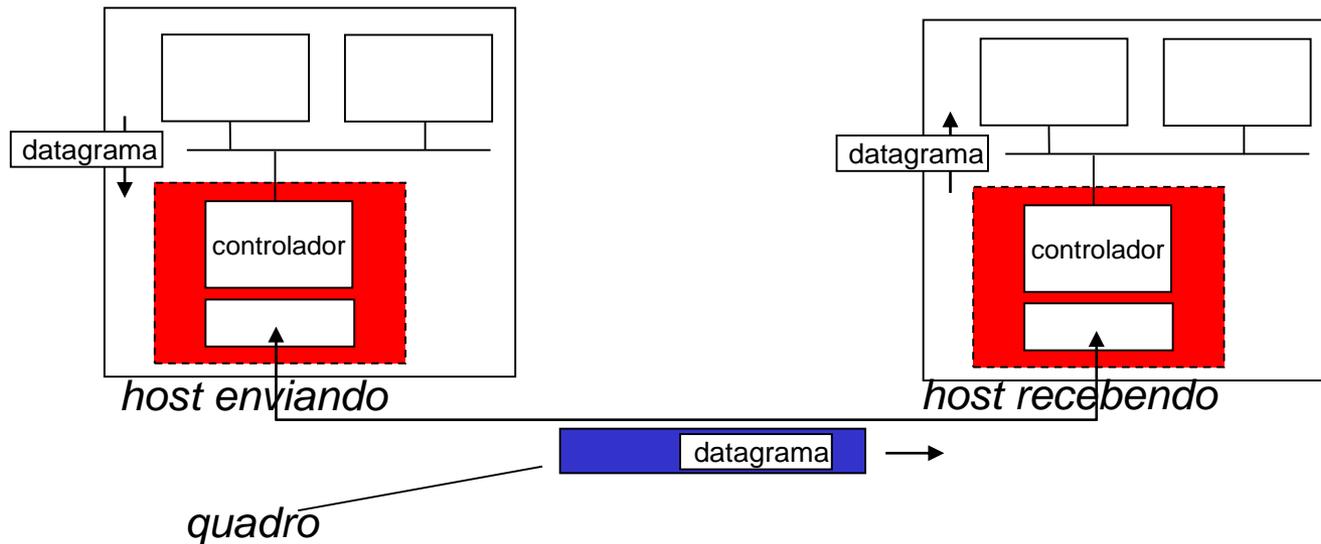
- com *half-duplex*, nós de ambos os lados do enlace podem transmitir, mas não ao mesmo tempo

Onde a camada de enlace é implementada?

- ❖ em cada e todos os *hosts*
- ❖ camada de enlace implementada em “adaptador” (ou *cartão de interface de rede* NIC) ou na placa-mãe (*on board*)
 - cartão Ethernet (Intel 8254x), cartão 802.11; *chipset* Ethernet
 - implementa camadas de enlace e física
 - Exemplo
- ❖ afixa-se no sistema de barramentos do *host*
- ❖ combinação de *hardware*, *software*, *firmware*



Comunicação entre adaptadores



❖ lado enviando:

- encapsula datagrama em quadro
- adiciona bits verificadores de erro, rdt, controle de fluxo, etc.

❖ lado recebendo:

- procura por erros, rdt, controle de fluxo, etc
- extrai datagrama, passa para camada superior no lado recebendo

Camada de enlace, LANs: estrutura

2.1 introdução, serviços

2.2 detecção, correção
de erros

2.3 protocolos de acesso
múltiplo

2.4 LANs

- endereçamento, ARP
- Ethernet
- *switches*
- VLANs

2.5 virtualização de enlace:
MPLS

2.6 redes de *data centers*

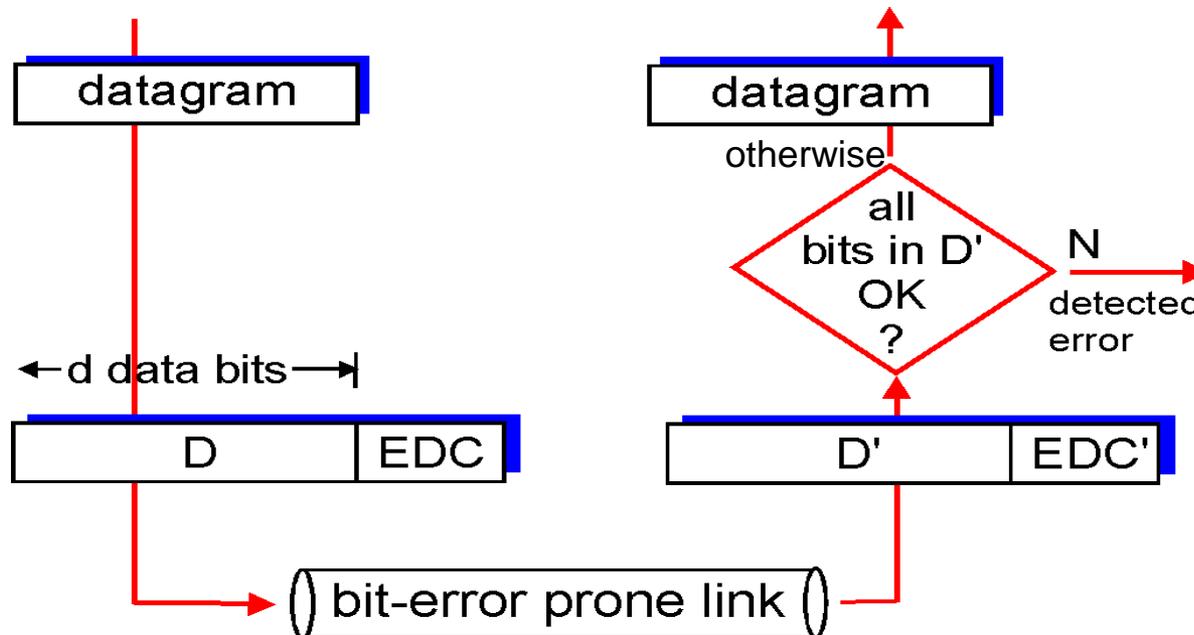
2.7 resumo: como
funciona um pedido
web

Detecção de erros

EDC = *Error Detection and Correction bits* (redundância)

D = Dados protegidos por verificação de erros, podem incluir campos de cabeçalho

- Detecção de erros não é 100 % confiável!
 - protocolo pode deixar passar alguns erros, mas raramente
 - campo EDC maior permite melhor detecção e correção
- Vamos rever verificação de paridade (básico), *checksum* (transporte) e CRC (enlace)
- *Mais detalhes nos cursos de comunicações!!*

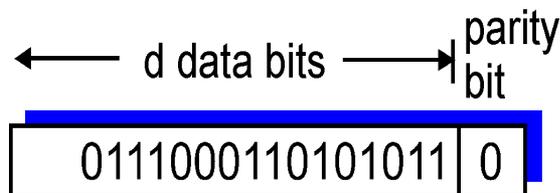


Verificação de paridade

paridade com bit

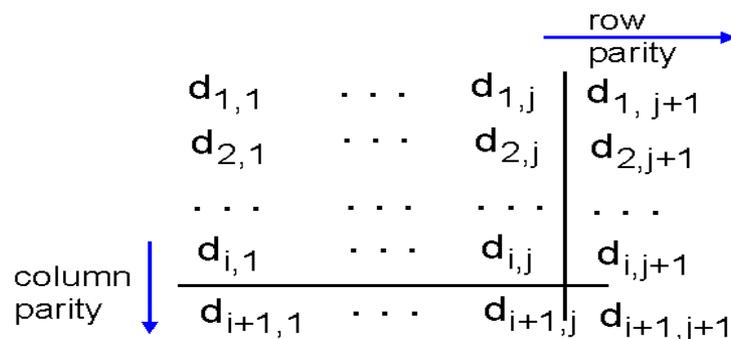
único:

- ❖ detecta erro em um único bit



paridade de bit bidimensional:

- ❖ detecta e corrige erro em um único bit
- ❖ *Forward error correction (FEC) – Ex. CDs*



10101	1
111100	
011101	
10101	0

no errors

10101	1
1 1100	0
011101	
10101	0

parity error

*correctable
single bit error*

* Veja exercícios interativos em:

http://gaia.cs.umass.edu/kurose_ross/interactive/2d_parity.php

Checksum da Internet (revisão)

objetivo: detectar “erros” (e.g., bits invertidos) em pacotes transmitidos (nota: usado nas camadas de transporte e rede)

remetente:

- ❖ trata conteúdo de segmento como sequência de inteiros de 16 bits
- ❖ *checksum*: adição (soma complemento de 1) do conteúdo dos segmentos
- ❖ remetente coloca valor da *checksum* no campo checksum do UDP ou TCP

receptor:

- ❖ calcula *checksum* do segmento recebido
- ❖ verifica se *checksum* calculado é igual ao valor do campo checksum:
 - NÃO - erro detectado
 - SIM- não há erro detectado. Mas talvez ainda hajam erros mesmo assim?

- ❖ *pouco overhead mas pouca proteção comparada com CRC*
- ❖ *Checksum em software CRC em hardware*

Verificação de redundância cíclica (CRC - *Cyclic redundancy check*)

- ❖ Muito usado na prática (Ethernet, 802.11 WiFi, ATM)!
- ❖ conhecidos também como *códigos polinomiais*
- ❖ Interpretar bits de dados, D , como um número binário
- ❖ Emissor e receptor escolhem padrão de $r+1$ bits (gerador), G
- ❖ Fonte: escolhe r bits CRC, R , tais que
 - $\langle D, R \rangle$ é exatamente divisível por G (módulo 2)
- Receptor conhece G , divide $\langle D, R \rangle$ por G . Se resto é não nulo: erro detectado!
 - pode detectar erros em rajada com menos do que $r+1$ bits

← d bits → ← r bits →



$D * 2^r \text{ XOR } R$ *mathematical formula*

Exemplo CRC

- ❖ Operações realizadas módulo-2 sem “vai-um” ou “empresta-1”
- ❖ Assim “+” = “-” = XOR

Como calcular R ?

queremos:

$$D \cdot 2^r \text{ XOR } R = nG$$

equivalentemente:

$$D \cdot 2^r = nG \text{ XOR } R$$

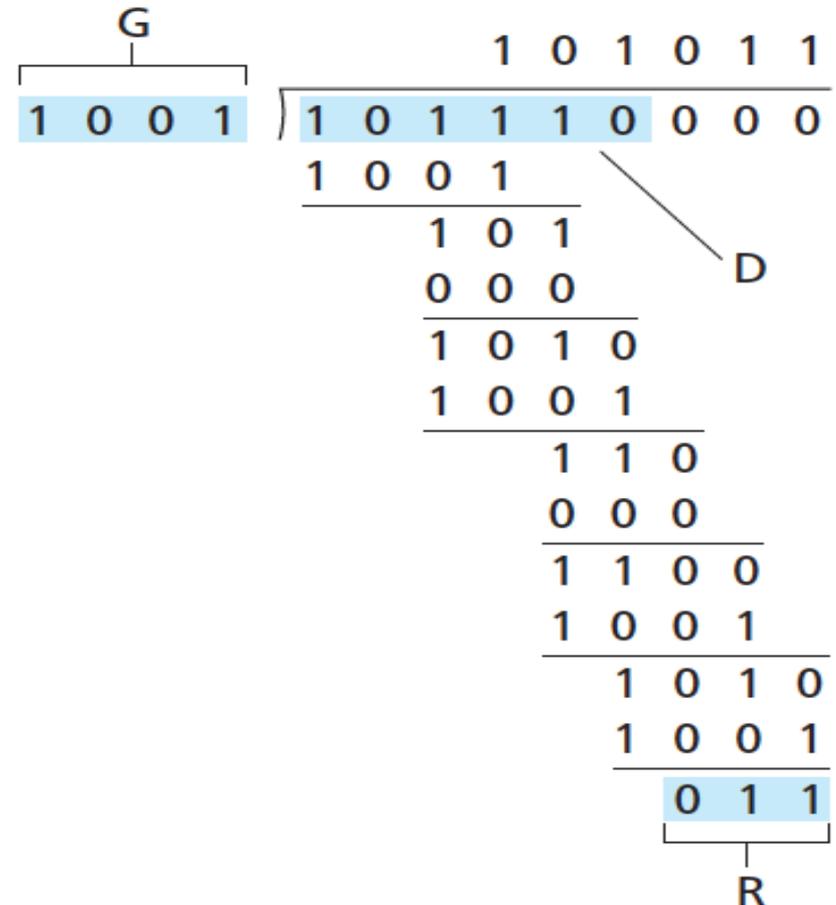
equivalentemente:

se dividirmos $D \cdot 2^r$ por G :

$$R = \text{resto de } \left[\frac{D \cdot 2^r}{G} \right]$$

Exercício:

- ❖ Calcular R para $D = [101110]$, $r=3$, $G = [1001]$. Qual a palavra de código nesse caso?
- ❖ **Resposta $R = 011$, palavra = $[101110011]$**



Camada de enlace, LANs: estrutura

5.1 introdução, serviços

5.2 detecção, correção
de erros

5.3 protocolos de acesso
múltiplo

5.4 LANs

- endereçamento, ARP
- Ethernet
- *switches*
- VLANs

5.5 virtualização de enlace:
MPLS

5.6 redes de *data centers*

5.7 resumo: como
funciona um pedido
web

Enlaces de acceso múltiplo, protocolos

2 tipos de “enlaces”:

❖ **ponto-a-ponto**

- PPP para acesso discado (*dial-up*)
- enlace ponto a ponto entre *switch* Ethernet e *host*

❖ **de difusão ou *broadcast* (fio ou meio compartilhado)**

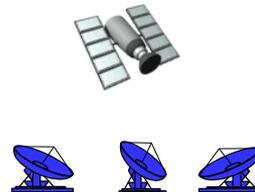
- Ethernet antiga
- *upstream* do HFC (ISP cabo)
- LAN sem fio 802.11



fio compartilhado (e.g., Ethernet cabeada)



RF compartilhada (e.g., 802.11 WiFi)



RF compartilhada (satélite)



pessoas em um coquetel (ar compartilhado, acústico)

Protocolos de acesso múltiplo

- ❖ canal de *broadcast* compartilhado único
- ❖ 2 ou mais transmissões simultâneas de nós: interferência
 - *colisão* se nó recebe 2 ou mais quadros ao mesmo tempo
 - Em geral, nesse caso, o nó recebendo não consegue entender nenhum dos dois quadros: perdidos!
 - Canal desperdiçado durante transmissão dos quadros

protocolo de acesso múltiplo

- ❖ algoritmo distribuído que determina como nós compartilham canal, i.e., determina quando cada nó transmite
- ❖ Assunto de Teleco estudado a mais de 40 anos e com muita atividade ainda!
- ❖ comunicação sobre compartilhamento do canal precisa usar canal também!
 - não há canal *out-of-band* para coordenação

Um protocolo de acesso múltiplo ideal

dado: canal de difusão *broadcast* com taxa R bps

Deseja-se:

1. quando apenas 1 nó quer transmitir, consegue fazê-lo à taxa R .
2. quando M nós querem transmitir, cada pode transmitir à *taxa média* de R/M
3. totalmente descentralizado:
 - não há nó especial para coordenar transmissões
 - não há sincronização de relógios, intervalos
4. simples e barato de implementar

Protocolos MAC (Medium Access Control) : taxonomia

3 grandes categorias:

❖ *particionamento de canal*

- divide canal em “pedaços” menores (intervalos de tempo, frequência, código)
- aloca pedaços para nós para uso exclusivo
- TDMA, FDMA, CDMA
- Telefonia fixa e celular convencional, TV, rádio
- Muito mais detalhes nos cursos de Telecom...

❖ *acesso aleatório*

- canal não dividido, permite colisões
- “recupera-se” de colisões
- Ethernet, Wi-fi

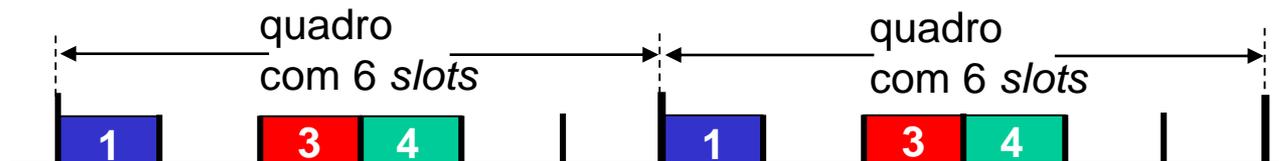
❖ *“revezamento”*

- nós revezam-se, mas nós com mais dados a enviar podem ocupar turnos maiores
- Bluetooth

Protocolos MAC por *particionamento de canal*: TDMA

TDMA: *Time Division Multiple Access*

- ❖ acesso ao canal em “turnos”
- ❖ cada nó obtém intervalo (*slot*) de tempo de duração fixa (duração = tempo de transm. de 1 pacote) a cada turno (quadro)
- ❖ *slots* não usados ficam desocupados
- ❖ exemplo: LAN com 6 nós, 1,3,4 têm pacotes, *slots* 2,5,6 desocupados



Protocolos MAC por particionamento de canal: FDMA

FDMA: *Frequency Division Multiple Access*

- ❖ espectro do canal dividido em faixas de frequências
- ❖ a cada nó é designado uma faixa de frequência
- ❖ faixas de frequência em que não ocorre transmissão ficam desocupadas
- ❖ exemplo: LAN com 6 nós, 1,3,4 têm pcts, faixas de frequência 2,5,6 desocupadas

