

## Tiempo de transmisión

### Ethernet

Se tiene en cuenta el ACK porque es half-duplex sino preguntar en el examen, si dijera full-duplex no se tiene en cuenta el ACK

### UDP

$$T_{transmisión} = \frac{(L_{paquete} + 26(Cabecera Ethernet) + 12(IFG)).8}{V_{transmisión}}$$

### TCP

$$L_{ACK} = 0 + IP + TCP$$

$$T_{transmisión\ datos} = \frac{(L_{paquete} + 26(Cabecera Ethernet) + 12(IFG)).8}{V_{transmisión}}$$

$$T_{transmisión\ ACK} = \frac{(L_{ACK} + 26(Cabecera Ethernet) + 12(IFG)).8}{V_{transmisión}}$$

### Wifi

Se tiene en cuenta el ACK porque es half-duplex sino preguntar en el examen, si dijera full-duplex no se tiene en cuenta el ACK

$$T_{backoff} = \frac{(Valor\ backoff\ 1 + valor\ backoff\ 2)}{2} * slot\ time$$

Preámbulo:

Corto: 72 bits de preámbulo 48 bits de cabecera

Largo: 144 bits de preámbulo y 48 bits de cabecera

$$T_{PC} = T_{preámbulo} + T_{cabecera}$$

$$T_{Datos} = \frac{L_{paquete} + Cabecera\ MAC}{R} + T_{PC}$$

$$T_{ACK} = \frac{14*8}{\min(6, R_b)} + T_{PC}$$

### UDP

$$T_{transmisión} = T_{DIFS} + T_{SIFS} + T_{backoff} + T_{Datos} + T_{ACK}$$

### TCP

$$L_{ACK} = 0 + IP + TCP$$

$$T_{ACK\ TCP} = \frac{L_{ACK} + \text{Cabecera MAC}}{R} + T_{PC}$$

$$T_{transmisión\ datos} = T_{DIFS} + T_{SIFS} + T_{backoff} + T_{Datos} + T_{ACK}$$

$$T_{transmisión\ ACK} = T_{DIFS} + T_{SIFS} + T_{Datos\ ACK} + T_{ACK}$$

## Ocupación

### UDP

$$ocupación = \frac{t_{transmisión}}{frecuencia(us)}$$

### TCP

$$ocupación = \frac{t_{transmisión}}{frecuencia(us)} + \frac{t_{transmisión\ ACK}}{2 * frecuencia(us)}$$

## Número de sensores

$$N = \frac{1}{\sum_{i=0}^n ocupacion_i} \text{ Se redondea para abajo}$$

$$N = \frac{1 - ocupación_j}{\sum_{i=0}^n ocupacion_i} \text{ Se redondea para abajo}$$

## ATM

$$Celdas = \frac{L_{paquete} + 8}{48} \rightarrow \text{El resultado se redondea para arriba}$$

Es full-duplex y no se tiene en cuenta el ACK de TCP

## Velocidad de transmisión (cps)

$$AB_{cps} = \frac{celdas}{Frecuencia(s)}$$

## Velocidad de transmisión (Mbps)

$$AB_{Mbps} = \frac{celdas * 53 * 8}{Frecuencia(us)}$$

## Erlang

	Canales simultáneos
Tráfico total(erlangs)	% pérdida(tabla B) o demora (tabla C)

## Número de ubicaciones

Limita el Giga Ethernet ( 1000Mbps) → mirar limitación mínima de octetos

$$L = L_{paquete} + 26(Cabecera Ethernet) + 12(IFG) > 768$$

Entonces:

$$L = L_{paquete} + 26(Cabecera Ethernet) + 12(IFG)$$

Sino:

$$L = 768$$

$$AB_i = \frac{L \cdot 8}{frecuencia (us)}$$

$$N_{ubicaciones} = \frac{1000}{\sum_{i=0}^n AB_i \cdot N_{sensores_i}} \text{ Se redondea para abajo}$$

## Tiempo activo cuando nos dan Erlang por sensor

$$Erlang = 0,004$$

$$0,04 = \frac{T_{activo}}{3600 s} \quad T_{activo} = 0,004 \cdot 3600 = 14,4s$$

## Total sensores ejercicio de Erlang

$$A = 6,3$$

$$Total\ sensores = \frac{Tráfico\ total}{Tráfico\ sensores} = \frac{6,3}{0,004} = 1575 \text{ Se redondea para abajo}$$

## Ejercicio de asignar ip

Miramos los routers, a los cuales le damos dos ips, definimos las redes en función de los sensores y el número de ubicaciones, y decimos que redes ponemos y si son pocas máquinas damos IP a cada una o solo decir el total de IPs. No asignamos IPs a los switch ni a los APs, ni a los dispositivos que conectan con Wifi.

## Ejercicio de despejar M

Se multiplica M tanto por la trama como por el tiempo, pero no a la cabecera (leer bien el enunciado por si cambia).

## Gráfica temporal

Se calculan todos los enlaces (solo de ida y si lo dice también de vuelta), hay que tener en cuenta que de vuelta el ATM aunque limite por ejemplo a 768B no ponemos 768B después del ATM sino que quitamos el padding que le mete y calculamos con el valor sin el padding. Ej: Tenemos ATM y pasamos a un ethernet de 10Mbps, ¿usamos 768B para el ethernet ? No, usamos el valor real.

#### Modo infraestructura

Tiempo de transmisión x2

#### Modo ad-hoc

Tiempo de transmisión x1

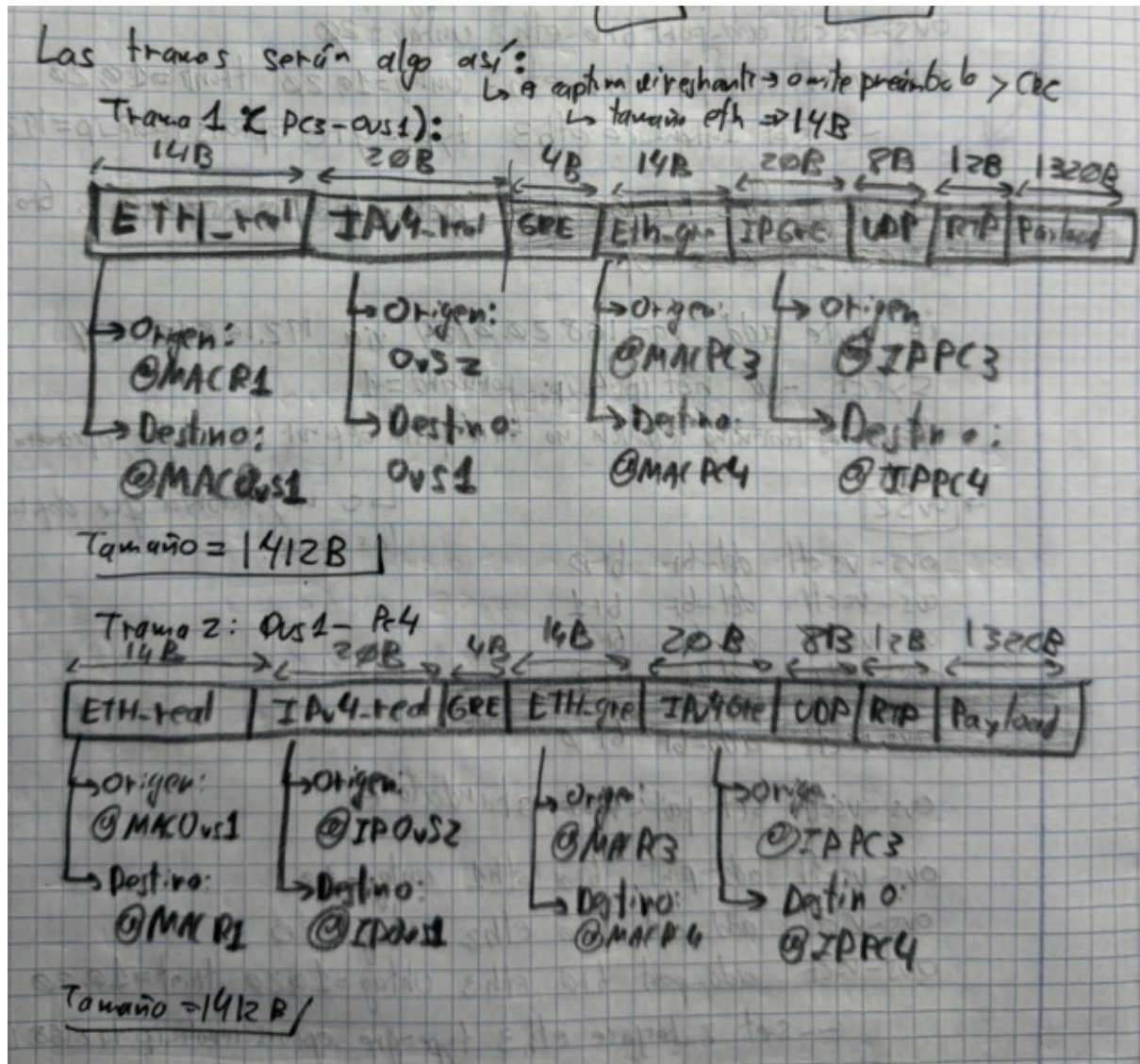
#### Tiempo de propagación

Despreciable

## EXTRAS

wireshark: no tiene en cuenta preámbulo ni crc → cabecera eth = 14B

Gre: En este ejemplo concreto el paquete pasa por un router por lo que son dos paquetes con el mismo destino, el de ida y vuelta no el request-reply



Examen 2022/2023 c2

OVS: Buscar en práctica 3.2

Este ejemplo corresponde a la práctica

Shell

# OVS-1

ovs-vsctl del-br br0

```
ovs-vsctl del-br br1
ovs-vsctl del-br br2
ovs-vsctl del-br br3
ovs-vsctl add-port br0 eth1
ovs-vsctl add-port br0 eth2
ovs-vsctl add-port br0 eth3-- set Interface eth3 type=gre
options:remote_ip=192.168.40.1
ifconfig eth3 192.168.20.1 netmask 255.255.255.0 broadcast 192.168.20.255 up
ovs-vsctl set-fail-mode br0 secure
route add default gw 192.168.20.254 eth3
```

#### # ARP

```
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.3,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.4,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.2,actions=output:2
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.3,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.4,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.1,actions=output:1

ovs-ofctl add-flow br0
in_port=3,dl_type=0x0806,nw_dst=192.168.10.1,actions=output:1
ovs-ofctl add-flow br0
in_port=3,dl_type=0x0806,nw_dst=192.168.10.2,actions=output:2
```

#### # IP

```
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1

ovs-ofctl add-flow br0
in_port=3,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:1
ovs-ofctl add-flow br0
in_port=3,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:2
```

## # OVS-2

```
ovs-vsctl del-br br0
ovs-vsctl add-br br0
ovs-vsctl add-port br0 eth1
ovs-vsctl add-port br0 eth2
ovs-vsctl add-port br0 eth3-- set Interface eth3 type=gre
options:remote_ip=192.168.20.1
ifconfig eth3 192.168.40.1 netmask 255.255.255.0 broadcast 192.168.40.255 up
ovs-vsctl set-fail-mode br0 secure
route add default gw 192.168.40.254 eth3
```

## # ARP

```
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.1,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.2,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0806,nw_dst=192.168.10.4,actions=output:2
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.1,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.2,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0806,nw_dst=192.168.10.3,actions=output:1

ovs-ofctl add-flow br0
in_port=3,dl_type=0x0806,nw_dst=192.168.10.3,actions=output:1
ovs-ofctl add-flow br0
in_port=3,dl_type=0x0806,nw_dst=192.168.10.4,actions=output:2
```

## # IP

```
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:3
ovs-ofctl add-flow br0
in_port=1,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.1,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.2,actions=output:3
ovs-ofctl add-flow br0
in_port=2,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1

ovs-ofctl add-flow br0
in_port=3,dl_type=0x0800,nw_dst=192.168.10.3,actions=output:1
```

```
ovs-ofctl add-flow br0  
in_port=3,dl_type=0x0800,nw_dst=192.168.10.4,actions=output:2
```

valores pre definidos wifi

802.11g

SIFS 10  $\mu$ s

Slot Time 9  $\mu$ s

DIFS 28  $\mu$ s

Backoff [0, 15]