

# Capa de enlace

Álvaro González Sotillo

November 23, 2017

## Contents

<b>1</b>	<b>Introducción</b>	<b>1</b>
<b>2</b>	<b>Tramas</b>	<b>2</b>
<b>3</b>	<b>Control de errores</b>	<b>2</b>
<b>4</b>	<b>Corrección de errores</b>	<b>4</b>
<b>5</b>	<b>Referencias</b>	<b>6</b>

## 1 Introducción

### 1.1 La capa de enlace

- Es la capa 2 de la arquitectura OSI.
- Se encarga de conseguir que la comunicación de datos se produzca correctamente a través de un medio físico de transmisión.
- Para lograr que dos dispositivos adyacentes se comuniquen, se necesita un control del intercambio de datos: el control del enlace.
- La capa de enlace proporciona a la capa de Red un servicio de transporte de bits fiable (asegura que los bit se transmiten correctamente por el medio físico).
- El bloque de datos transmitido se denomina TRAMA.

### 1.2 Funciones de la capa de enlace

- Sincronización a nivel de trama.
- Control de flujo: las estaciones deben ponerse de acuerdo en el ritmo de transmisión de datos.
- Control de errores: los enlaces no son perfectos. Hay que controlar que no haya errores en la transmisión.
- Direccionamiento: si hay varios posibles destinos, es necesario identificar a quien va dirigida la trama.
- Gestión del enlace:

- 
- Inicio de la transmisión
  - Mantenimiento de la transmisión
  - Finalización de la transmisión

### 1.3 MAC y LLC

- En la arquitectura IEEE 802, el nivel de enlace se divide en dos subcapas:
  - LLC: se encarga de las funciones comunes de la capa independientemente del medio físico usado
    - \* Control de errores
    - \* Direccionamiento
    - \* Sus funciones han sido definidas por el subgrupo 802.2.
  - MAC: se encarga del acceso al medio (gestión del enlace)

## 2 Tramas

- Una trama es un bloque de bits agrupados que son enviados por la línea.
- El tamaño de la trama depende del tipo de red.
- Agrupar los bits en tramas facilita:
  - la detección y corrección de errores
  - la compartición del medio.
- Una trama se compone de tres partes
  - Información sobre la trama
  - Datos.
  - Redundancia.

## 3 Control de errores

- Consiste en enviar algunos bits añadidos a los datos con información que permita detectar o corregir los errores.
- El porcentaje de redundancia se calcula como

$$\frac{\text{bits de control}}{\text{bits totales}} \times 100$$

- Los errores pueden
  - Detectarse
  - Adicionalmente, corregirse

---

### 3.1 Detección de errores

- ECO
  - El receptor envía una copia exacta de la información recibida al emisor.
  - El emisor confirma con otra trama que la información es correcta
- Paridad lineal. Se añade un bit extra, indicando si el número de bits con valor a 1 es par o impar.
  - 100100, con paridad par, se envía como 100100 0
  - 100100, con paridad impar, se envía como 100100 1
  - Problema: ¿Qué pasa si cambia un número **par** de bits?

#### 3.1.1 Paridad de bloque

- Paridad de bloque. Se distribuyen los datos en una tabla y se calcula paridad por cada línea y columna.
  - Mensaje: 1100101 0110110 1011010 1001111 0111001 1100111 1010000, con paridad par
  - Se envía 1100101 0 0110110 0 1011010 0 1001111 1 0111001 0 1100111 1 1010000 0 1001000 0

	Datos	Paridad lineal
	1100101	0
	0110110	0
	1011010	0
	1001111	1
	0111001	0
	1100111	1
	1010000	0
Paridad de bloque	1001000	0

### 3.2 Actividad

- Calcular la paridad bidimensional del siguiente mensaje:
  - 1001101, 1111010, 1100110, 1110001, 1101001, 1110111, 0010111

### 3.3 Distancia de Hamming

- Cuando se produce un error, cambian algunos bits
- Según la codificación utilizada, no todas las combinaciones de 0s y 1s son posibles
  - Ejemplo: 4B/5B
- La **distancia de Hamming** de un código es la cantidad de bits que hay que cambiar en una combinación válida para llegar a otra combinación válida
- Cuanto mayor sea la distancia, más robusto es el código frente a errores
- ¿Cuál es la distancia de Hamming de una transmisión con paridad?

---

### 3.4 CRC

- Al principio de la comunicación, emisor y receptor acuerdan un Polinomio Generador.
- Al iniciar la transmisión se añaden un número predeterminado de ceros a la información a enviar y se divide utilizando el polinomio generador.
- El receptor realiza nuevamente una división sobre los datos recibidos y si el resto es 0 indica que la trama se ha recibido sin errores.
- Finalmente se descartan los bits añadidos en el transmisor para quedarnos con el mensaje original.

#### 3.4.1 ¿Por qué CRC?

- Hay versiones de CRC para diferentes longitudes de polinomio: CRC16, CRC32,...
- Los errores se producen típicamente a ráfagas
- Para un CRC de  $n$  bits
  - Se detectan todos los errores de ráfagas de menos de  $n$  bits incorrectos
  - Se detecta una fracción de las ráfagas más largas  $(1 - 2^{-n})$

Longitud de crc	Porcentaje de detección de ráfagas mayores
8	99.609375
16	99.998474
32	99.999999767169

## 4 Corrección de errores

- La detección de errores es el primer paso
- Una vez detectado:
  - Se puede ignorar (las capas más altas deben arreglar el error)
  - Se puede corregir
- Ethernet no corrige errores, pero veremos algunas técnicas que pueden usar otras capas 2

### 4.1 Retransmisión

- Es el método de corrección más sencillo.
- Se detecta el error y se pide al emisor que vuelva a enviar la trama.
- Se tienen que memorizar las tramas enviadas hasta la recepción de un ACK que confirme que el envío de información fue exitosa.

### 4.2 Corrección: Código Hamming

- Codificación que permite la detección y la corrección de un bit

- 
- Su distancia de Hamming es 1
  - Se incluyen bits de paridad de la siguiente forma:
    - Los bits de las posiciones  $s = 2^{p-1}$  son de paridad: 1, 2, 4, 8...
    - El resto son de datos
    - El bit de la posición  $s$  se incluye en el bit de paridad  $p$  si la expresión de  $s$  en binario tiene a 1 el bit  $p$

	p1 s1	p2 s2	d1 s3	p3 s4	d2 s5	d3 s6	d4 s7	p4 s8	d5 s9	d6 s10	d7 s11
p1	x		x		x		x		x		x
p2		x	x			x	x			x	x
p3				x	x	x	x				
p4								x	x	x	x

#### 4.2.1 Ejemplo Hamming

- Para transmitir 0110101
- Se transmite 10001100101

	p1	p2	d1	p3	d2	d3	d4	p4	d5	d6	d7
Datos			0		1	1	0		1	0	1
p1		1	0		1		0		1		1
p2		0	0			1	0			0	1
p3				0	1	1	0				
p4								0	1	0	1
Datos (con paridad):	1	0	0	0	1	1	0	0	1	0	1

#### 4.2.2 Detección de un error con Hamming

- Se recibe 11001100101

	p1 s1	p2 s2	d1 s3	p3 s4	d2 s5	d3 s6	d4 s7	p4 s8	d5 s9	d6 s10	d7 s11	Paridad
Recibida:	1	1	0	0	1	1	0	0	1	0	1	
p1	1		0		1		0		1		1	0
p2		1	0			1	0			0	1	1 (error)
p3				0	1	1	0					0
p4								0	1	0	1	0

- Hay un error, y se localiza en la posición 0010: s2 (p2)

#### 4.2.3 Ejercicio

- Decide si las siguientes palabras de código Hamming son correctas. Si no son correctas, corrígelas.

- 
- 10001100101
  - 00100110010
  - 01110111001

#### 4.2.4 ¿Y si hay más de 1 error?

- Su distancia de Hamming es 1, así que no se puede detectar
- En Hamming extendido se añade un bit de paridad adicional
  - Permite detectar errores de dos bits, pero no corregirlos

### 4.3 Ejercicio

- Calcula el porcentaje de redundancia de:
  - Tramas de 1000 bytes con crc32
  - Tramas de 100 bytes con crc16
  - Hamming
  - Hamming extendido
  - Paridad lineal: un bit de paridad cada 7 de datos
  - Paridad de bloque: un bit de paridad lineal gcada 7 de datos, bloques de 49 bits de datos

## 5 Referencias

- Formatos:
  - [Transparencias](#)
  - [PDF](#)
- Creado con:
  - [Emacs](#)
  - [org-reveal](#)
  - [Latex](#)