



FUNDAMENTOS DE REDES

CAPA DE ENLACE

Daniel Barragán C.

daniel.barragan@correounivalle.edu.co

Lunes y Miércoles 3:00 pm a 5:00 pm – Edificio 331 Oficina 2114

Capa de enlace

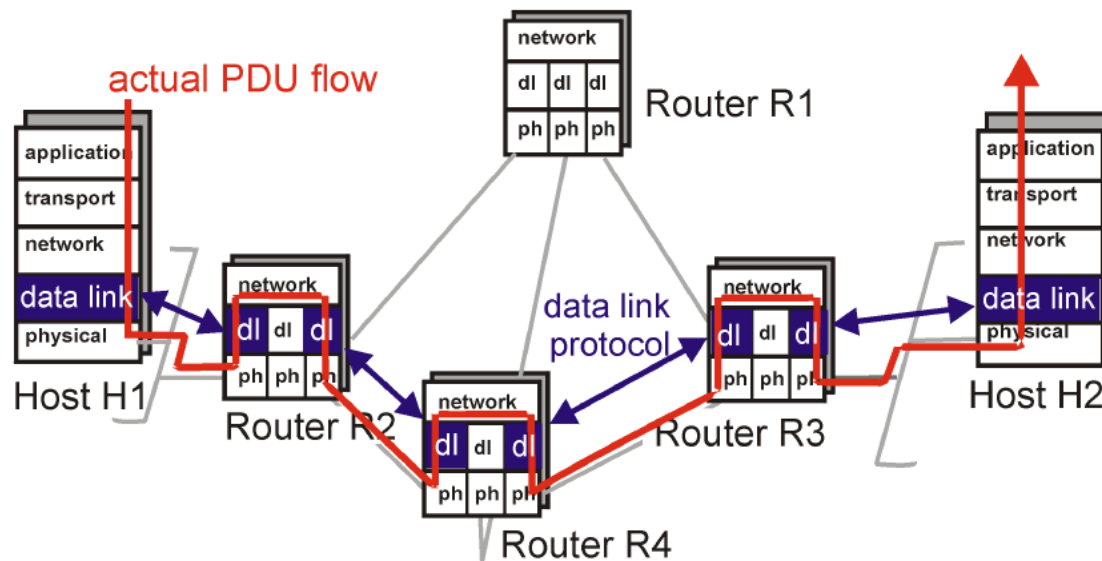


Agenda

- Introducción
- Servicios de la Capa de Enlace
- Corrección/Detección de Error
- Protocolos de Acceso Múltiple
- Protocolos: Ethernet, HDLC

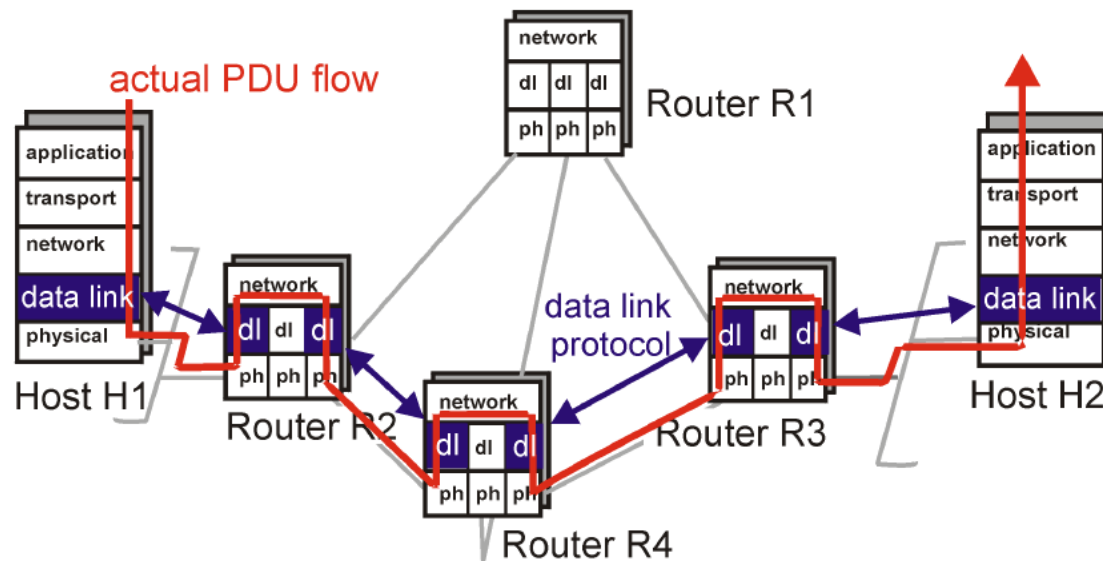
Introducción

Para mover un datagrama desde la fuente (emisor) hasta el destino (receptor), el datagrama debe ser llevado a través de enlaces individuales a lo largo de su camino



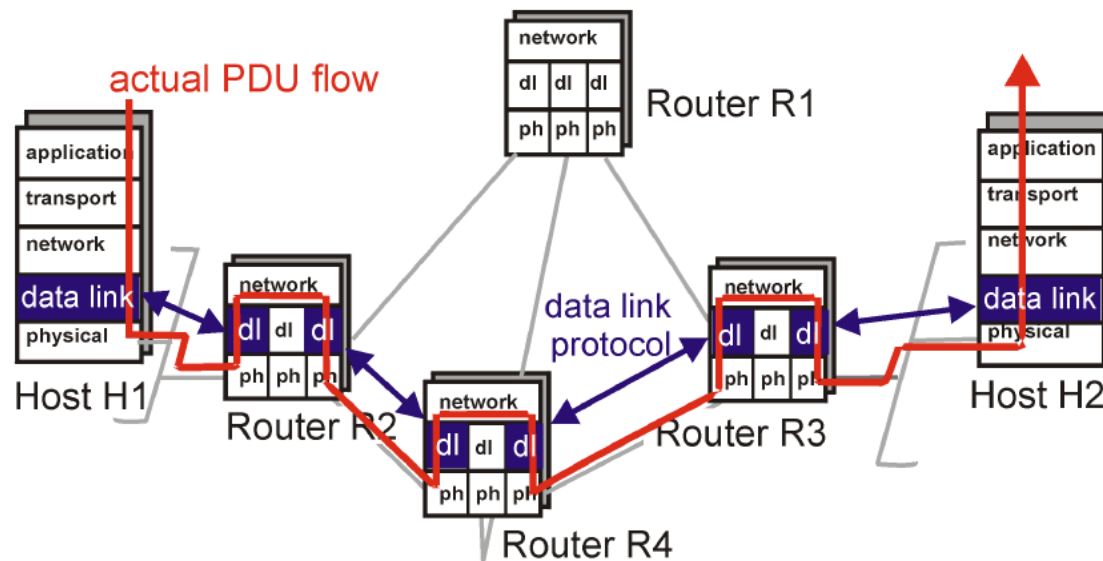
Introducción

En la gráfica se distinguen nodos (enrutadores, dispositivos finales) y enlaces



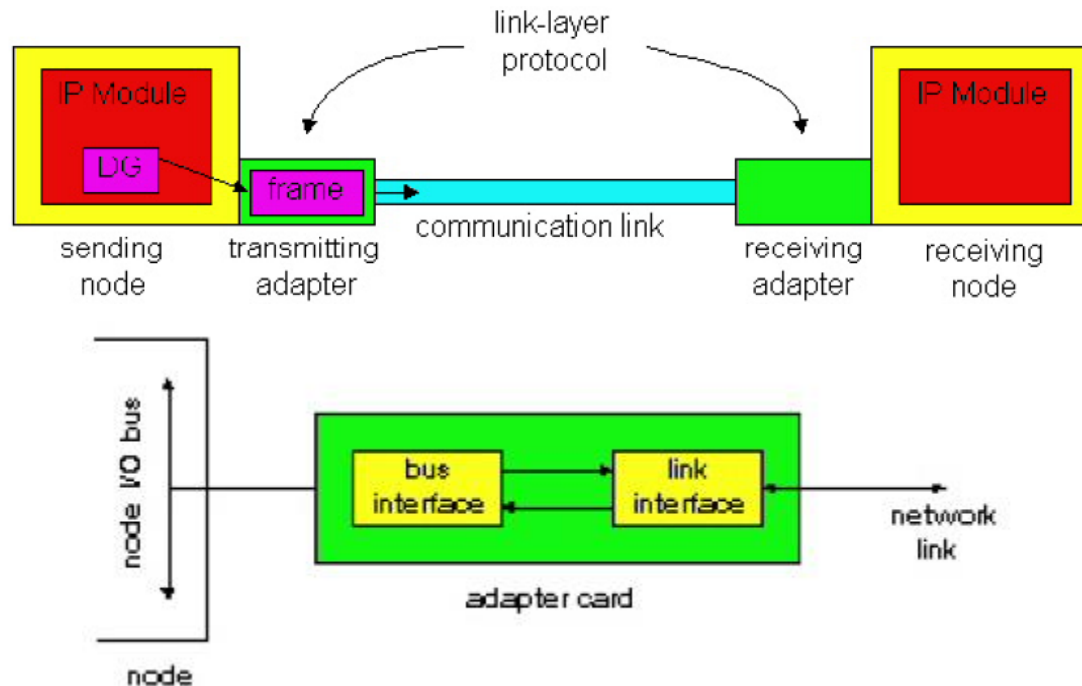
Introducción

Los diferentes enlaces a lo largo del camino del datagrama pueden usar distintos protocolos de capa de enlace



Introducción

En su mayoría los protocolos de la capa de enlace son implementados en los adaptadores de red (**Network Interface Card**)



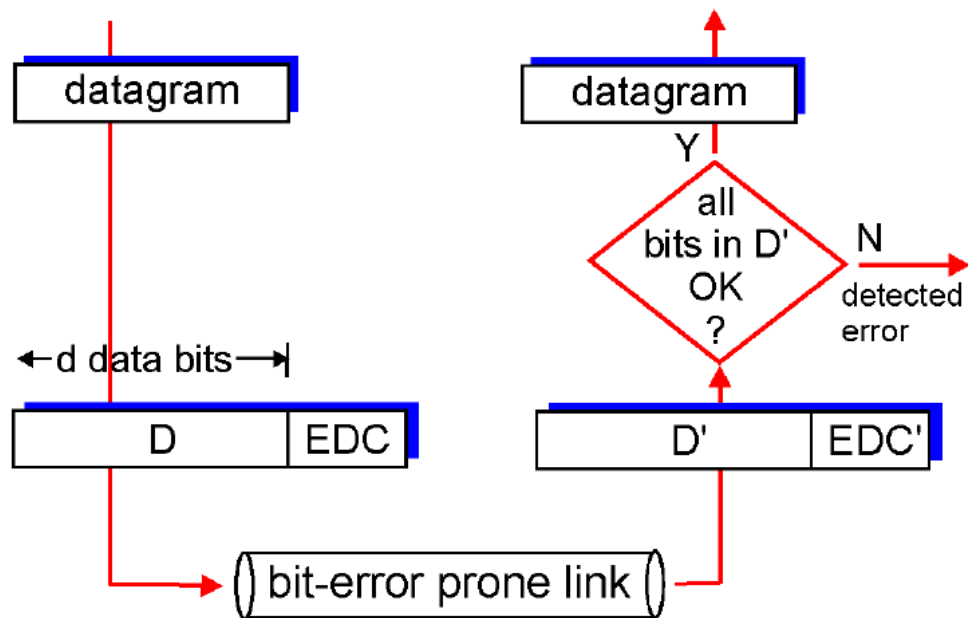
Servicios

La capa de enlace provee los siguientes servicios:

- Control de acceso al medio
- Detección de error : se implementa en hardware
- Corrección de error: se implementa en hardware
- Entrega confiable: confirmaciones de envío y recepción
- Control de flujo: monitoreo de buffers en emisor y receptor
- Full duplex, half duplex

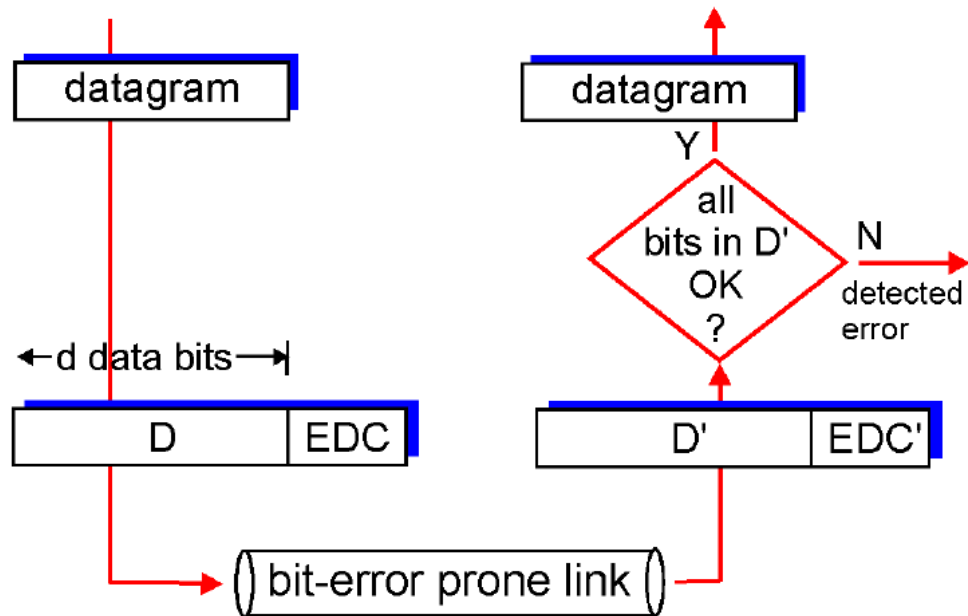
Corrección/Detección Error

Información (D) a ser protegida, se le añaden bits para la corrección y detección de error (EDC)



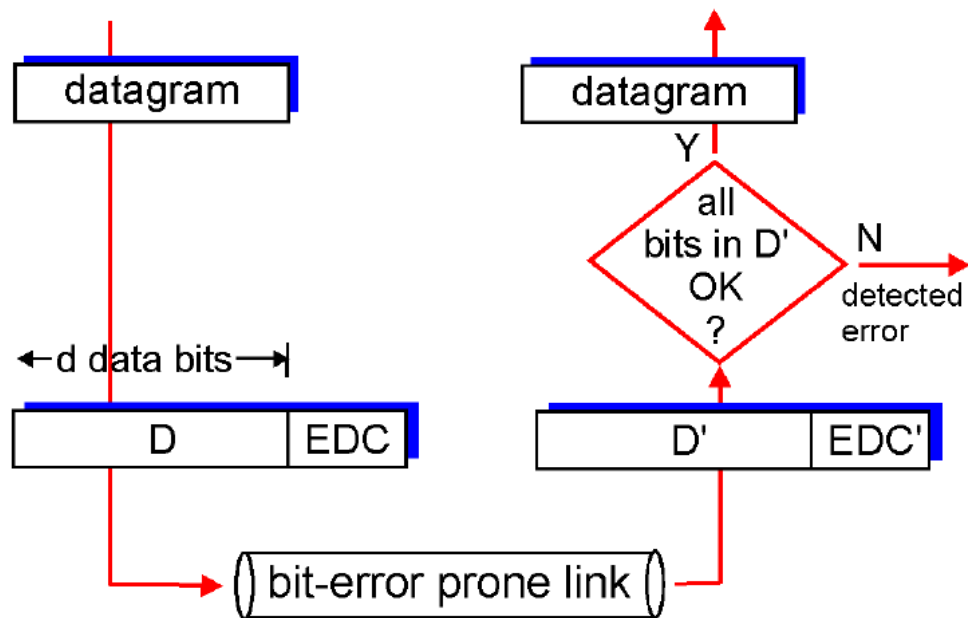
Corrección/Detección Error

El receptor debe determinar si la información recibida (D') es igual a la original (D), para ello se vale de EDC'



Corrección/Detección Error

Existe la posibilidad de que los errores no sean detectados. Por lo general se escoge un esquema de corrección de error donde esta posibilidad sea baja

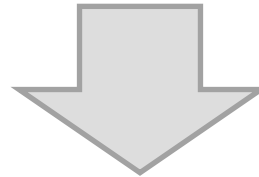


Detección de Paridad

Si la información a enviar es D, en un esquema de paridad par se añade un bit de tal manera que la cantidad de unos sea par

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

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



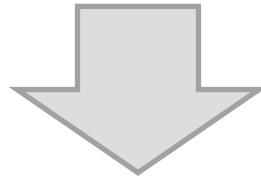
1	1	0	0	0	1	1	1	0	1
---	---	---	---	---	---	---	---	---	---

Detección de Paridad

Si la información a enviar es D, en un esquema de paridad impar se añade un bit de tal manera que la cantidad de unos sea impar

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

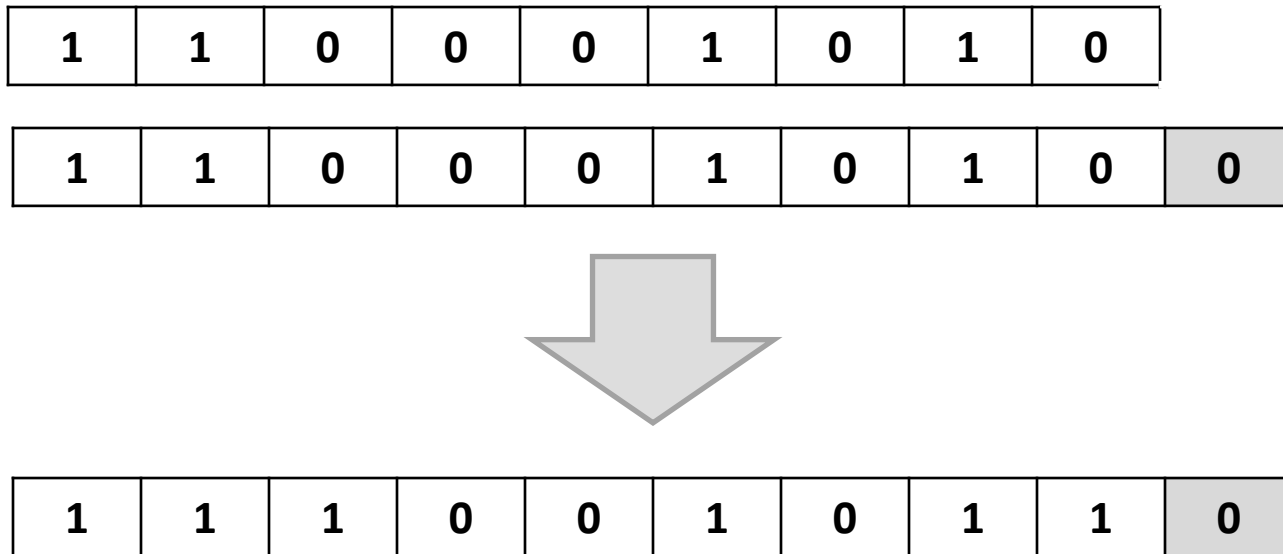
1	1	0	0	0	1	1	1	0	0
---	---	---	---	---	---	---	---	---	---



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

Detección de Paridad

¿Para el siguiente esquema con paridad par que sucede si en la transmisión cambian dos bits simultáneamente?



Detección de Paridad

Con el siguiente esquema de paridad bidimensional se puede detectar y corregir un error

$d(1,1)$...	$d(1,j)$	$d(1,j+1)$
$d(2,1)$...	$d(2,j)$	$d(2,j+1)$
...
$d(i,1)$...	$d(i,j)$	$d(i,j+1)$
$d(i+1,1)$		$d(i+1,j)$	

Detección de Paridad

Con el siguiente esquema de paridad bidimensional se puede detectar y corregir un error

1	0	1	0	1
1	1	1	1	0
0	1	1	1	0

Detección de Paridad

Con el siguiente esquema de paridad bidimensional se puede detectar y corregir un error

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	

Detección de Paridad

Con el siguiente esquema de paridad bidimensional se puede detectar y corregir un error

1	0	1	0	1	1
1	1	1	1	0	0
0	1	0	1	0	1
0	0	1	0	1	

Detección de Paridad

Con el siguiente esquema de paridad bidimensional se puede detectar y corregir un error

1	0	1	0	1	1
1	1	1	1	0	0
0	1	0	1	0	1
0	0	1	0	1	

Código Hamming

Es un detector y corrector de errores. Este esquema intercala bits de paridad entre los bits de datos

Se emplean la siguiente estructura de acuerdo a la cantidad de bits a calcular su equivalente Hamming:

$p_1 - p_2 \ d_1 - p_3 \ d_2 \ d_3 \ d_4 - p_4 \ d_5 \ d_6 \ d_7 \ d_8 \ d_9 \ d_{10} \ d_{11} - \dots$

Código Hamming

Ejemplo:

Para el caso de $D = 0110101$. Determinar los bits a ser transmitidos empleando código Hamming

$p_1 - p_2 d_1 - p_3 d_2 d_3 d_4 - p_4 d_5 d_6$ **d_7**

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1											
p_2											
p_3											
p_4											
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1			0		1		0		1		1
p_2											
p_3											
p_4											
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1			0		1		0		1		1
p_2			0			1	0			0	1
p_3											
p_4											
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1			0		1		0		1		1
p_2			0			1	0			0	1
p_3					1	1	0				
p_4											
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1			0		1		0		1		1
p_2			0			1	0			0	1
p_3					1	1	0				
p_4									1	0	1
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1	1		0		1		0		1		1
p_2			0			1	0			0	1
p_3					1	1	0				
p_4									1	0	1
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1	1		0		1		0		1		1
p_2		0	0			1	0			0	1
p_3					1	1	0				
p_4									1	0	1
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1	1		0		1		0		1		1
p_2		0	0			1	0			0	1
p_3				0	1	1	0				
p_4									1	0	1
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1	1		0		1		0		1		1
p_2		0	0			1	0			0	1
p_3				0	1	1	0				
p_4								0	1	0	1
Datos (con paridad):											

Código Hamming

Solución:

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7
Datos (sin paridad):			0		1	1	0		1	0	1
p_1	1		0		1		0		1		1
p_2		0	0			1	0			0	1
p_3				0	1	1	0				
p_4								0	1	0	1
Datos (con paridad):	1	0	0	0	1	1	0	0	1	0	1

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1													
p_2													
p_3													
p_4													

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1	1		0		1		0		1		1		
p_2													
p_3													
p_4													

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1	1		0		1		0		1		1		
p_2		0	0			1	0			0	1		
p_3				0	1	1	0						
p_4								0	1	0	1		

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1	1		0		1		0		1		1	OK	0
p_2		0	0			1	0			0	1		
p_3				0	1	1	0						
p_4								0	1	0	1		

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1	1		0		1		0		1		1	OK	0
p_2		0	0			1	0			0	1	OK	0
p_3				0	1	1	0					OK	0
p_4								0	1	0	1	OK	0

	p_4	p_3	p_2	p_1	
Binario					
Decimal					$\Sigma = ?$

Código Hamming

Solución (Sin Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	1		
p_1	1		0		1		0		1		1	OK	0
p_2		0	0			1	0			0	1	OK	0
p_3				0	1	1	0					OK	0
p_4								0	1	0	1	OK	0

	p_4	p_3	p_2	p_1	
Binario	0	0	0	0	
Decimal	0	0	0	0	$\Sigma = 0$

Código Hamming

Solución (Con Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	0		
p_1													
p_2													
p_3													
p_4													

Código Hamming

Solución (Con Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	0		
p_1	1		0		1		0		1		0		
p_2		0	0			1	0			0	0		
p_3				0	1	1	0						
p_4								0	1	0	0		

Código Hamming

Solución (Con Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	0		
p_1	1		0		1		0		1		0	-	1
p_2		0	0			1	0			0	0	-	1
p_3				0	1	1	0					OK	0
p_4								0	1	0	0	-	1

	p_4	p_3	p_2	p_1	
Binario					
Decimal					$\Sigma = ?$

Código Hamming

Solución (Con Error):

	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	P	C
Datos recibidos	1	0	0	0	1	1	0	0	1	0	0		
p_1	1		0		1		0		1		0	-	1
p_2		0	0			1	0			0	0	-	1
p_3				0	1	1	0					OK	0
p_4								0	1	0	0	-	1

	p_4	p_3	p_2	p_1	
Binario	1	0	1	1	
Decimal	8	0	2	1	$\Sigma = 11$

Checksum

El **checksum** consiste en una suma de comprobación sobre la totalidad del paquete de información o sobre la cabecera e información por separado

El receptor realiza nuevamente la suma de comprobación y comprueba que sea igual al **checksum** recibido en el paquete de información

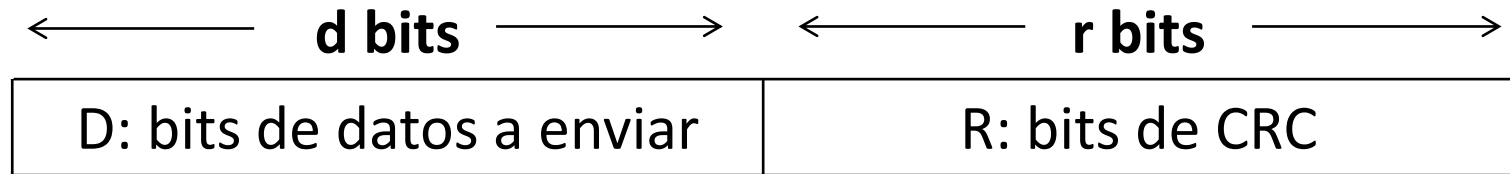
Control Redundancia Cíclica

Los códigos CRC son también llamados códigos polinomiales.

El modo de operación de CRC es el siguiente:

- El emisor y receptor deben conocer un patrón de $r+1$ bits llamado generador **G**
- El bit mas significativo de **G** debe ser **1**
- Para una información **D** (compuesta por **d** bits), el emisor concatena **r** bits adicionales de tal manera que al dividir los bits recibidos en el receptor **d + r** entre **G** el residuo es cero

Control Redundancia Cíclica



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

Control Redundancia Cíclica

Ejemplo:

Para el caso de $D = 101110$, $d = 6$, $G = 1001$ y $r = 3$. Determinar los bits a ser transmitidos empleando CRC

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

				1	0	1						
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			

[illegible]

[illegible]

				1	0	1	0					
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		

				1	0	1	0					
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0		

				1	0	1	0					
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	

				1	0	1	0	1				
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	
								1	0	0	1	

				1	0	1	0	1				
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	
								1	0	0	1	
									1	0	1	

				1	0	1	0	1				
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	
								1	0	0	1	
									1	0	1	0

				1	0	1	0	1	1			
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	
								1	0	0	1	
									1	0	1	0
									1	0	0	1

				1	0	1	0	1	1			
1	0	0	1	1	0	1	1	1	0	0	0	0
				1	0	0	1					
						1	0	1				
						0	0	0				
						1	0	1	0			
						1	0	0	1			
								1	1	0		
								0	0	0		
								1	1	0	0	
								1	0	0	1	
									1	0	1	0
									1	0	0	1
								R	->	0	1	1

Control Redundancia Cíclica

Solución:

Se transmite

← d bits →	← r bits →
D: bits de datos a enviar	R: bits de CRC
101110	011

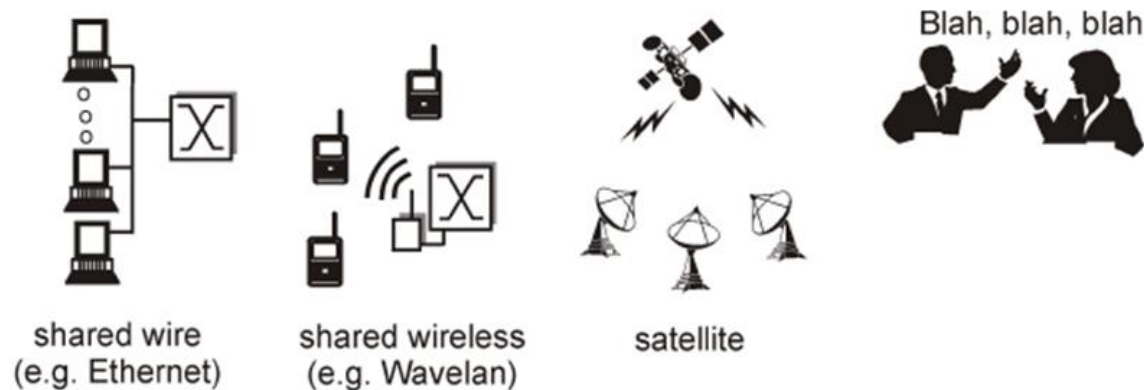
Control Redundancia Cíclica

Los patrones generadores **G** que se usan en **CRC** están definidos en estándares internacionales. IEEE por ejemplo define el siguiente patron **G** para el CRC de 32 bits

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

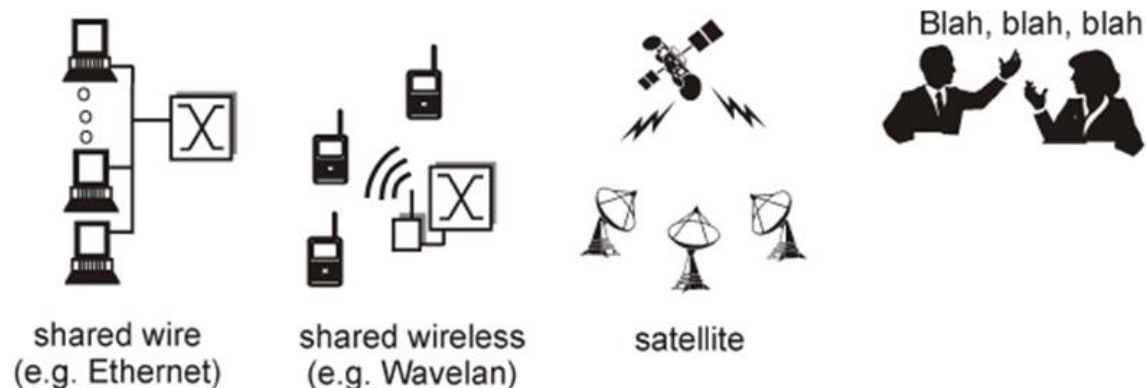
Protocolos de Acceso Múltiple

En esta sección se trata el problema de coordinar el acceso de múltiples nodos al canal de comunicación



Protocolos de Acceso Múltiple

Existen dos tipos de enlace: los enlaces punto a punto y los enlaces de difusión (**broadcast**). Los enlaces de difusión pueden tener múltiples emisores y receptores compartiendo el mismo canal de comunicación



Protocolos de Acceso Múltiple

El trabajo de coordinar el acceso por parte de múltiples emisores y receptores al canal lo realizan los protocolos de acceso al medio.

Los protocolos se clasifican en las siguientes categorías:

- Protocolos de partición de canal (FDM, TDM, SDMA, CDMA)
- Protocolos de acceso aleatorio (Aloha, CSMA)
- Protocolos de turnos

CSMA/CD

CSMA son las siglas de **Carrier Sense Multiple Access**. **CD** son las siglas de **Collision Detection**

Carrier Sense hace referencia a que un nodo que desea transmitir primero escucha si el canal esta libre para iniciar la transmisión de lo contrario espera un tiempo aleatorio y vuelve a intentarlo

Collision Detection hace referencia a que si un nodo mientras esta transmitiendo, escucha a otro transmitir, para de transmitir y espera un tiempo aleatorio antes de volver a transmitir

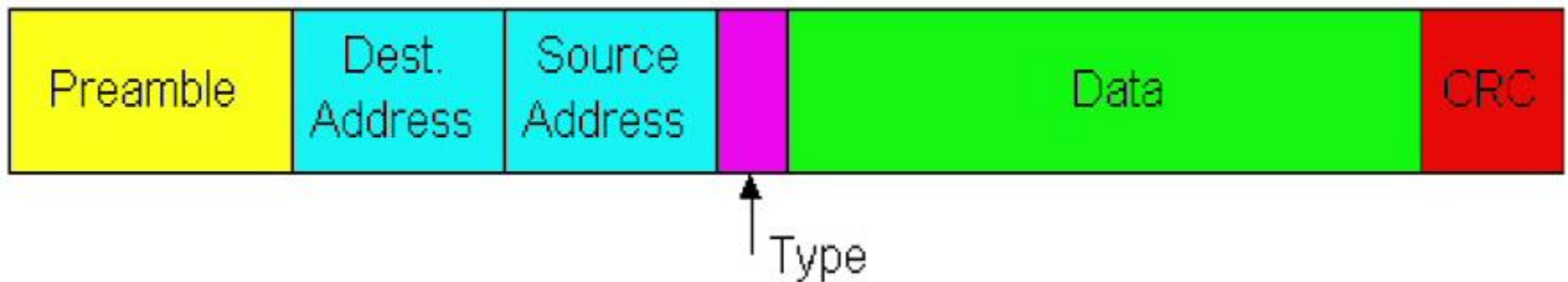
Protocolo Ethernet

Introducción

Ethernet fue la primera tecnología ampliamente difundida para la capa de enlace, de allí su popularidad hasta hoy en día

Protocolo Ethernet

Ethernet presenta la siguiente estructura, la cual aplica para todas las velocidades y medios físicos donde este implementada la tecnología



Protocolo Ethernet

Data Field (46 a 1500 bytes): Esta campo lleva el datagrama IP. Condiciona el tamaño del datagrama IP

Destination Address (6 bytes): Contiene la dirección destino. Si esta dirección coincide con la del equipo destino, se pasa el contenido del campo **data** a la capa de red

Source Address (6 bytes): Contiene la dirección fuente

Type Field (2 bytes): Un dispositivo final podría usar un protocolo distinto a IP en la capa de red, esta campo permite identificar el protocolo que se este usando

Cyclic Redundancy Check (4 bytes): Usado para detectar errores

Protocolo Ethernet

Preamble (8 bytes): Los primeros 7 bytes del preamble son 10101010 y el último byte es 10101011. Se usan para sincronizar el emisor y receptor ya que los adaptadores presentan pequeñas variaciones en su velocidad (100Mbps, 1Gbps, etc)

Protocolo Ethernet

Características

Ethernet provee un servicio no orientado a conexión. No hay handshaking

Ethernet provee un servicio de comunicación no confiable. No hay confirmaciones de recepción. Al detectar un error se descarta la trama sin comunicarlo al emisor, esto produce brechas en el flujo de datagramas que se entrega a la capa de red

Protocolo Ethernet

Características

Ethernet no tiene conocimiento de cuando retransmite una trama, para Ethernet cada trama es una trama nueva

Ethernet emplea CSMA/CD como protocolo de acceso múltiple

Protocolo Ethernet

Codificación

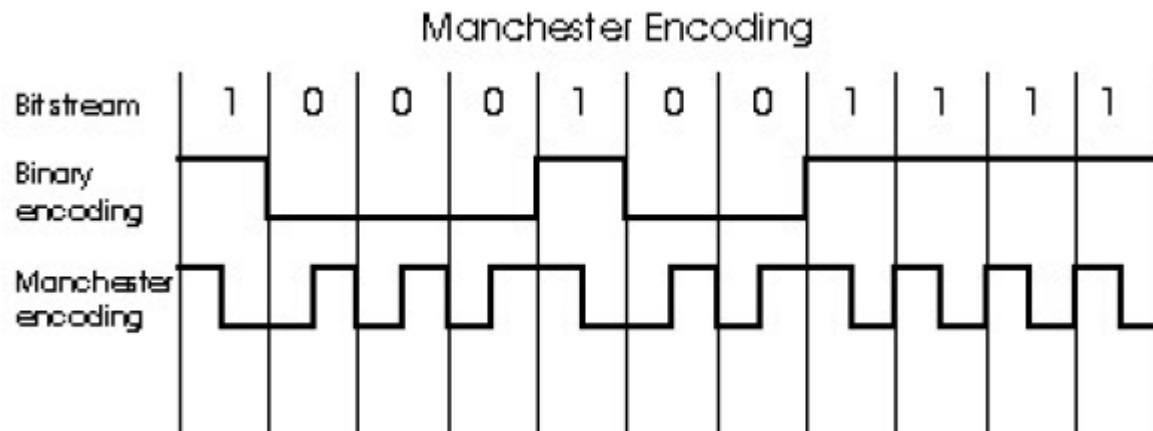
Ethernet envía una señal digital al medio de transmisión (transmisión en banda base).

Se emplea codificación **Manchester** (10Base2), codificación **4B5B** (1000BaseT) ó codificación **8B10B** (1000BaseS).

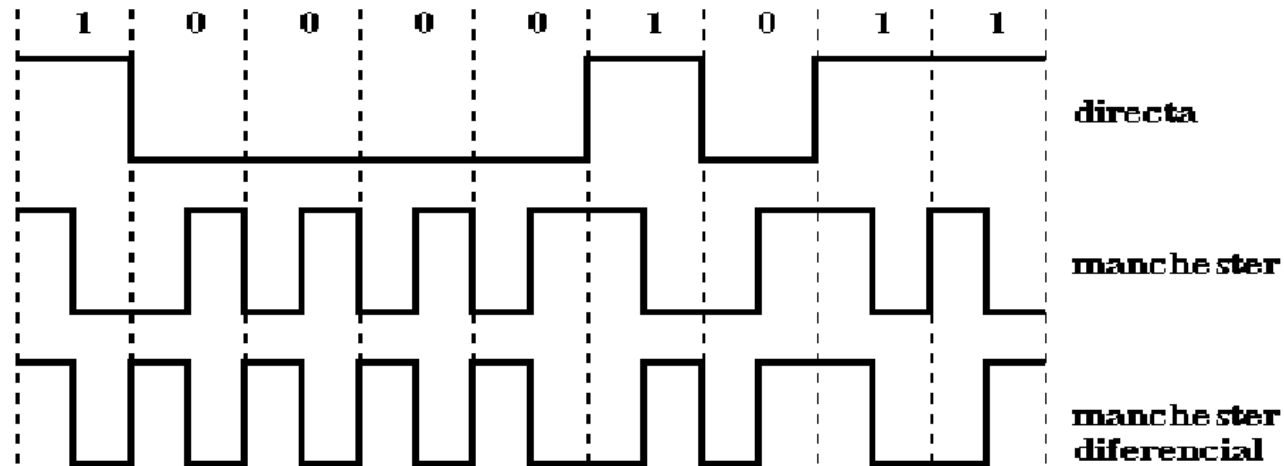
Protocolo Ethernet

Codificación

En la codificación Manchester cada bit tiene una transición. Un 1 tiene una transición de 1 a 0 y un 0 tiene una transición de 0 a 1. Esta codificación permite sincronizar al emisor y receptor



Protocolo Ethernet



manchester : 1 = alto en la primera mitad
0 = alto en la segunda mitad

manchester diferencial: 1 = ausencia de transición al principio
0 = transición al principio

Protocolo Ethernet

Tecnologías Ethernet

Las tecnologías mas populares de Ethernet son:

100BASE-T: para conexión de computadores a switches
1000BaseT o 1000BaseS: para conexión de switch a switch y de switch a servidor

1000 indica una velocidad de 1Gbps, T indica el uso de par trenzado (Cable UTP categoría 5), S indica el uso de fibra óptica

Protocolo Ethernet

Tecnologías Ethernet

La máxima longitud para 100BaseT que puede emplearse entre un adaptador de red y un switch es de 100 metros

La topología mas usada es la topología en estrella

Protocolo HDLC

HDLC son las siglas de Control de Enlace de Datos de Alto Nivel. Es un protocolo punto a punto y multipunto que opera en la capa de enlace

HDLC Proporciona recuperación de errores en caso de pérdida de paquetes (provee un servicio confiable)

HDLC define tres tipos de estaciones, tres configuraciones del enlace y tres modos de operación para la transferencia de los datos.

Protocolo HDLC

Los tipos de estaciones son: Primaria (controla el enlace), Secundaria (recibe ordenes de la primaria), Combinada

Los tipos de configuración del enlace son: No balanceada (una estación primaria y una o más secundarias), Balanceada (dos estaciones combinadas)

Los tres modos de transferencia de datos son: Modo de respuesta normal, Modo de respuesta asíncrono, Modo balanceado asíncrono

Bibliografía

Computer Networking: A Top-Down Approach

Sexta Edición (2012)

James F. Kurose and Keith W. Ross

Using Snort and Ethereal to Master The 8 Layers Of An Insecure Network

Primera Edición (2006)

Michael Gregg, Stephen Watkins, George Mays, Chris Ries, Ronald M. Bandes, Brandon Franklin

Asesorías

daniel.barragan@correounivalle.edu.co

Edificio 331 – Oficina 2114

Lunes y Miércoles 3:00 pm – 5:00 pm

