

Ejercicios T7 solucionados : control de errores

1. Si todos los protocolos de enlace proporcionaran el servicio de entrega segura, ¿sería redundante que TCP también implementara este servicio? ¿por qué o por qué no?

Solución:

No, no sería redundante. Aunque el servicio de entrega segura a nivel de enlace garantice la fiabilidad de la entrega entre dos nodos consecutivos pueden ocurrir imprevistos que hagan que finalmente el paquete no alcance su destino final. Por ejemplo, debido a un problema en el encaminamiento el paquete puede ser descartado en un router tras agotar su tiempo de vida. En este caso el nivel de enlace habría cumplido su tarea de entregar de forma fiable la trama al router y no detectaría el descarte posterior del paquete. Otra situación de fallo posible es que tras recibir correctamente la trama y reconocerla, el nodo tenga un problema de funcionamiento que le impidan transmitir la trama al nodo siguiente, por ejemplo, un fallo en la alimentación del sistema.

2. En la arquitectura de protocolos TCP/IP se pueden implementar técnicas de control de flujo tanto en el nivel de transporte como en el de enlace de datos. ¿Qué sentido tiene el que se hagan en los dos niveles? ¿implica en los dos niveles a los mismos emisores y receptores?

Solución:

En el nivel de transporte se trabaja con los elementos finales que se comunican, esto es las aplicaciones. Por tanto el control de flujo que aplica el protocolo TCP se hace entre dichos elementos. Esto permite dar a la aplicación un servicio orientado a la conexión y fiable. Teniendo en cuenta, además, que dichos elementos finales pueden estar en diferentes redes, y los datos que se envían entre ellos además pueden atravesar diferentes topologías y tecnologías de red.

En el nivel de enlace no todas las redes dan un servicio de control de flujo, esto dependerá del tipo de red. Por tanto, en este nivel el control de flujo se realiza entre elementos adyacentes, esto es pertenecientes a una misma red.

Además, en el nivel de transporte hay protocolos como el UDP que no realiza control de flujo y los segmentos que se envían a través de TCP pueden atravesar redes que tampoco realicen control de flujo.

Resumiendo: No es redundante el control de flujo en ambos niveles dado que implica a distintos elementos y se puede dar casos en que se aplique en el nivel de transporte (TCP) y no en el de enlace (si todas las redes que atraviesa son por ejemplo ethernet) y viceversa.

3. Indica qué dos estrategias se utilizan para la corrección de errores. Describe brevemente en que consiste cada una de ellas.

Solución:

Para poder corregir los errores se pueden emplear dos estrategias: FEC (*Forward Error Correction*) y ARQ (*Automatic Repeat Request*)

FEC añade información que permitirá al receptor reconstruir la información correcta (detección + recuperación), mientras que en ARQ el emisor tiene que retransmitir la información dañada (detección + reenvío)

4. Indica tres métodos de detección de errores. Compáralos (NO expliques en qué consisten) en términos de capacidad de detección y facilidad de implementación en software y hardware.

Solución:

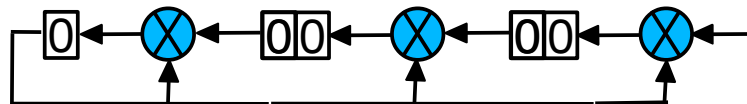
Protocolo IP Checksum y Ethernet CRC (Código de Redundancia Cíclica). Ambos son más eficientes que el de paridad, pero el CRC es el más eficiente de los tres en términos de capacidad de detección, dado que el Checksum determinadas combinaciones pueden dar un mismo resultado. En lo que respecta a la implementación el Checksum es fácilmente implementable por software y difícil de implementar por hardware. En cambio el CRC es complejo de implementar por software pero muy sencillo de implementar por hardware, con un registro de desplazamiento y puertas XOR.

5. Se quieren enviar los datos $D = 1010001101$. Dibuja la implementación hardware para calcular el CRC que tendríamos que añadir a dichos datos si emisor y receptor han acordado utilizar el generador 110101 .

Solución:

El generador en forma polinomial es: $x^5 + x^4 + x^2 + 1$

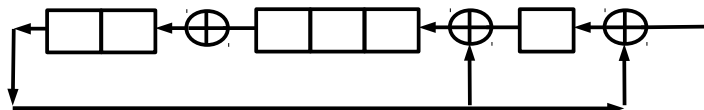
por lo que su implementación hardware para calcular el CRC sería:



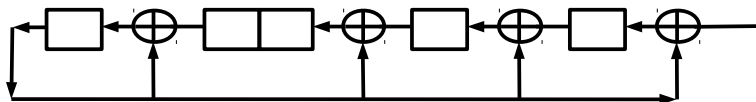
6. Se quieren enviar los datos 0100111011011001 usando un sistema de detección de errores basado en CRC. Dicho sistema va a usar el polinomio generador $x^6 + x^4 + x + 1$. Dibuja el circuito necesario para calcular el CRC que habrá que añadir a los datos.

Solución:

Como el polinomio es de grado 6, el circuito tendrá un registro de desplazamiento de 6 posiciones. Además, las posiciones que corresponden a coeficientes del polinomio que están a 1 ($1x^6 + 0x^5 + 1x^4 + 0x^3 + 0x^2 + x + 1$) estarán precedidos por una puerta XOR exceptuando el último término (x^6) que se manifiesta como la realimentación a todas las puertas del circuito.



7. A partir del circuito de la figura, obtén el polinomio generador que se está utilizando para calcular y verificar el CRC.

**Solución:**

El circuito tiene un registro de desplazamiento de 5 bits, luego el polinomio generador es de grado 5. Para obtener los términos con coeficiente 1 hay que fijarse en las posiciones de las puertas XOR:

antes de x^0 , x^1 , x^2 y x^4 . Por lo tanto, el generador en forma polinomial es: $x^5 + x^4 + x^2 + x + 1$

8. Calcula el CRC correspondiente al mensaje M=1011011101 empleando el polinomio generador $G(x) = x^4 + 1$. Dibuja el circuito que verificaría el mensaje y comprueba mediante una traza del mismo que el CRC calculado es correcto.

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

El CRC calculado es 0010. Por tanto, los datos a transmitir serían 1011011101**0010**

El circuito y su traza serán:

A	B	C	D	Datos
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	1	0	1	1
1	0	1	1	0
0	1	1	1	1
1	1	1	1	1
1	1	1	0	1
1	1	0	0	0
1	0	0	1	1
0	0	1	0	0
0	1	0	0	0
1	0	0	0	1
0	0	0	0	0
0	0	0	0	

