



Redes de Computadores

Grado en Ingeniería Informática.

Curso 2014/2015

Alumno:

Grupo:

PROBLEMA EVALUABLE

Sea un protocolo de comunicación bidireccional en la capa n de una arquitectura de red. Se trata de un protocolo de intercambio de datos con envío continuo y numeración de 2 bits, además de confirmación de errores (NACK) de los datos.

El elemento emisor del protocolo se encuentra inicialmente a la espera de solicitudes de envío de datos por parte del nivel $n+1$. Cada vez que el nivel $n+1$ solicita el envío de un bloque de datos, el emisor transmitirá la información en cuatro bloques independientes numerados de 0 a 3. Cada paquete se envía con un intervalo de tiempo entre ellos, de manera que existe un temporizador de envío en el emisor y cada vez que expira se envía uno de los cuatro bloques. Después de cada transmisión se inicia el temporizador de envío. Mientras el emisor espera la expiración del temporizador para cada envío, éste puede recibir un paquete NACK numerado (0-3) del receptor, indicando que un bloque numerado (0-3) no llegó correctamente, por lo que el emisor reenvía ese bloque y vuelve a iniciar el temporizador de envío. Cuando se transmite una secuencia completa de bloques (0-3), el emisor pasa a esperar un ACK del receptor confirmando que la secuencia de envío de datos se ha completado correctamente. Cuando el emisor recibe el ACK de confirmación, informa al nivel $n+1$ de la transmisión correcta y pasa a esperar una nueva petición del nivel superior.

El elemento receptor del protocolo se encuentra inicialmente a la espera de un bloque de datos con numeración 0. Cada vez que recibe un bloque de datos numerado en la secuencia 0 a 3, comprueba si los datos han sufrido algún error. En caso de detectar un error en un bloque de datos, lo descarta y envía un NACK numerado al emisor, indicando la secuencia con error, volviendo a esperar la misma secuencia de datos. Si el bloque de datos es correcto, almacena el bloque y pasa a esperar el siguiente, exceptuando que sea la última secuencia (3). En ese caso, si los datos son correctos, enviará al nivel superior el bloque completo de datos recibido e informará al emisor de la correcta recepción con una confirmación ACK. Pasará así de nuevo al estado inicial.

Determina los estados, eventos de entrada y salida, y la MEF que describe el funcionamiento del elemento RECEPTOR del protocolo.

Estados Receptor

- ED0 → Receptor espera datos con numeración 0.
- ED1 → Receptor espera datos con numeración 1.
- ED2 → Receptor espera datos con numeración 2.
- ED3 → Receptor espera datos con numeración 3.

Eventos de entrada

D0_IN → Receptor recibe datos correctos con numeración 0.
D1_IN → Receptor recibe datos correctos con numeración 1.
D2_IN → Receptor recibe datos correctos con numeración 2.
D3_IN → Receptor recibe datos correctos con numeración 3.
D0_IN_E → Receptor recibe datos erróneos con numeración 0.
D1_IN_E → Receptor recibe datos erróneos con numeración 1.
D2_IN_E → Receptor recibe datos erróneos con numeración 2.
D3_IN_E → Receptor recibe datos erróneos con numeración 3.

Eventos de salida

NACK0_OUT → Receptor envía un NACK con numeración 0.
NACK1_OUT → Receptor envía un NACK con numeración 1.
NACK2_OUT → Receptor envía un NACK con numeración 2.
NACK3_OUT → Receptor envía un NACK con numeración 3.
STORE → Receptor almacena un bloque de datos recibido.
DATA_NS → Receptor envía toda la información recibida al nivel superior (n+1).
ACK_OUT → Receptor envía un ACK informando de la recepción correcta de todos los bloques.

MEF

