- Trabajo realizado por el equipo de la Biblioteca Digital de la Fundación Universitaria San Pablo-CEU
- Me comprometo a utilizar esta copia privada sin finalidad lucrativa, para fines de investigación y docencia, de acuerdo con el art. 37 del T.R.L.P.I. (Texto Refundido de la Ley de Propiedad Intelectual del 12 abril 1996)

Capítulo 5

PROTOCOLOS DE CONTROL

CONTROL DE LA COMUNICACIÓN

Como se ha visto anteriormente, el proceso de transmisión de datos conlleva una serie de procedimientos que van desde el nivel físico hasta la presentación de la información en un formato determinado (nivel de aplicación).

Aunque todos ellos son fundamentales, se va a profundizar en el nivel de enlace que es el encargado del control de la comunicación.

Toda comunicación se puede dividir en tres fases:

- **Establecimiento de la comunicación**. En esta fase se establece la conexión física entre los ordenadores y se ponen de acuerdo en cuanto al procedimiento empleado para el intercambio de la información.

- **Transferencia de la información**. Ambos sistemas intercambian datos a través del enlace establecido. En caso de producirse un error en la recepción de los datos, se detecta y se solicita su reenvío.

- **Terminación**. En esta fase se da por finalizada la comunicación.

La forma de establecer y finalizar la comunicación depende de cómo estén conectadas las dos estaciones de trabajo (a través de un cable por la puerta serie o paralelo, a través de una línea punto a punto, a través de un módem por la red telefónica, etc.).

La forma de controlar la transferencia de la información depende exclusivamente del protocolo que se utilice. Este protocolo corresponde al nivel de enlace de datos del modelo *OSI* y deberá realizar las siguientes funciones:

- Sincronización de la comunicación.
- Control de los errores de transmisión.
- Coordinación de la comunicación.
- Recuperación ante los fallos que se produzcan.

Cuando se ha de transmitir una determinada información, la información se distribuirá en bloques de una longitud determinada, dispuesta en un orden determinado y con un control de errores que permitirá comprobar que todos y cada uno de los *bits* enviados sean iguales a todos y cada uno de los *bits* recibidos. De esta forma, si se produjera un error en uno de los bloques, únicamente sería necesario volver a transmitir dicho bloque sin necesidad de repetir toda la transmisión.

Entre los protocolos más adecuados de este nivel, se encuentran:

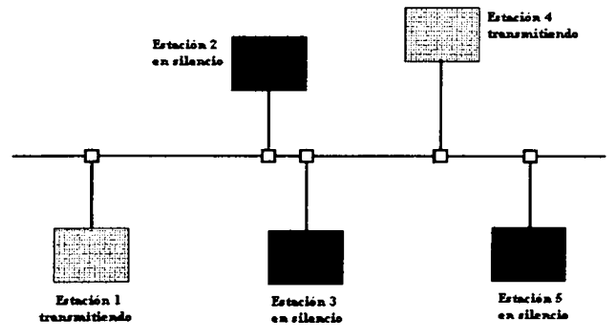
- Contienda.
  - Contienda simple.
  - Acceso múltiple por detección de portadora (*CSMA*).
  - Acceso múltiple por detección de portadora con detección de colisiones (*CSMA/CD*).
  - Acceso múltiple por detección de portadora evitando colisiones (*CSMA/CA*).
- Llamada selectiva (*Polling*).
- Paso de testigo (*Token passing*).

Protocolos de contienda

Se entiende por protocolos de contienda al método de acceso a la línea basado en que el primero que llega a ella es el primero que la utiliza.

CONTIENDA SIMPLE

En este protocolo todas las estaciones comparten el mismo canal de transmisión y los mensajes se envían a través de dicho canal; las estaciones responden únicamente a los mensajes que incluyen su dirección y el resto los ignoran; mientras no reciban un mensaje que incluya su dirección, se encuentran en estado de espera pero escuchando el canal de transmisión.



Protocolo de contienda simple

Por tanto, se pueden dar dos situaciones: que las estaciones se encuentren transmitiendo datos o que se encuentren en estado de espera.

Una estación envía los bloques de datos sin fijarse si el canal está disponible o no. Cuando un bloque de una estación coincide con el de otra se produce una colisión y ambos se destruyen automáticamente. Si éste llega a su destino, la estación receptora envía un mensaje indicando que lo ha recibido. Si la estación emisora, después de un tiempo aleatorio, no ha recibido este mensaje, vuelve a repetir la transmisión del bloque y así sucesivamente, hasta que haya finalizado la transmisión de datos.

Este tipo de protocolo no se utiliza en redes con cargas medias o altas, ya que se estarían produciendo colisiones constantemente y el rendimiento de la red sería muy bajo y con tiempos de espera muy grandes.

ACCESO MÚLTIPLE POR DETECCIÓN DE PORTADORA (CSMA)

En este protocolo también se utiliza un único canal pero una estación no transmite hasta que la línea está libre.

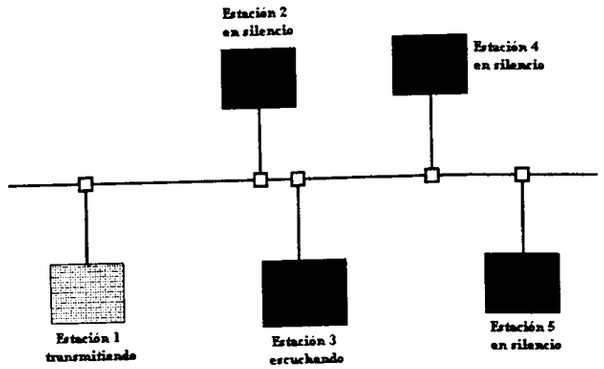
Para ello, la estación emisora se pone a la escucha, en una frecuencia secundaria, para saber si hay otra estación que esté enviando algún bloque de datos.

Mientras se encuentra a la escucha puede actuar de dos maneras distintas:

- Escuchar continuamente a la espera de que quede libre y entonces transmitir (detección continua de portadora).

- Escucha si el canal está ocupado. Si lo está, deja la transmisión un tiempo aleatorio y después vuelve a intentarlo (detección no continua de portadora).

Cuando la línea está libre, envía el bloque de datos y, además, otra señal en la frecuencia secundaria para avisar a las demás estaciones que la línea está ocupada.



Protocolo de acceso múltiple con detección de portadora (CSMA)

Una vez transmitido el bloque de datos, la estación espera hasta recibir el mensaje de que la estación receptora ha recibido el bloque. Si no lo recibe o recibe una señal negativa, la estación supone que se ha producido una colisión (por haber iniciado dos estaciones emisoras un envío simultáneamente), espera un tiempo aleatorio y vuelve a enviar el bloque de datos.

Por tanto, se pueden dar tres situaciones: que las estaciones se encuentren transmitiendo datos, que se encuentren en estado de espera o que se encuentren escuchando la línea.

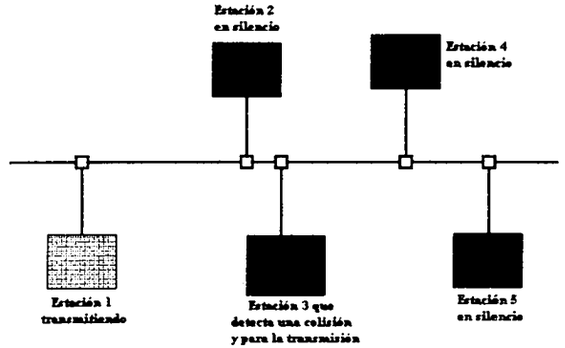
Este protocolo permite una mejora en comparación con el de contienda simple si la carga es baja o media y la red tiene una longitud pequeña, ya que entonces el tiempo que tarda la señal en propagarse es pequeño y el riesgo de que dos estaciones decidan enviar bloques de datos simultáneamente y colisionen, será bajo.

ACCESO MÚLTIPLE POR DETECCIÓN DE PORTADORA CON DETECCIÓN DE COLISIONES (CSMA/CD)

Este protocolo actúa de la misma manera que el anterior, pero, además de comprobar si la línea está libre antes de comenzar la transmisión, se comprueba si se ha producido alguna colisión durante la transmisión.

Si se ha producido alguna colisión, se detiene la transmisión y se vuelve a enviar el bloque de datos después de un tiempo de espera aleatorio.

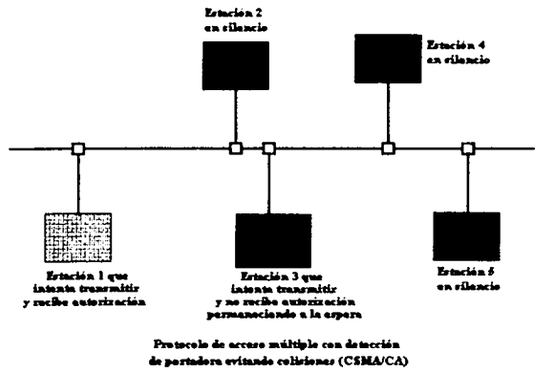
El rendimiento de este tipo de protocolo es mayor que en los dos anteriores, por ello es recomendable para cargas de tipo bajo o medio y para una longitud media de la red.



Protocolo de acceso múltiple con detección de portadora con detección de colisiones (CSMA/CD)

ACCESO MÚLTIPLE POR DETECCIÓN DE PORTADORA EVITANDO COLISIONES (CSMA/CA)

En este tipo de protocolo cuando una estación va a enviar un bloque de datos comprueba que la línea está libre y, cuando verifica que lo está, indica que tiene intención de transmitir.



Protocolo de acceso múltiple con detección de portadora evitando colisiones (CSMA/CA)

Si hay varias que se encuentran esperando, la transmisión se realiza por turno. En este turno se tiene en cuenta la prioridad de la estación y el orden en que se ha indicado que se desea transmitir, por tanto, primero transmitirá la que lo haya solicitado primero entre la que tienen la máxima prioridad y no la que lo haya solicitado primero si tiene una prioridad baja.

El rendimiento de este tipo de protocolo es mayor que en los tres anteriores, por ello es recomendable para cargas de tipo medio o alto y para una longitud media de la red.

Llamada selectiva (*Polling*)

Para poder utilizar este protocolo es necesario que la red disponga de dos tipos de estaciones: la estación principal y las secundarias.

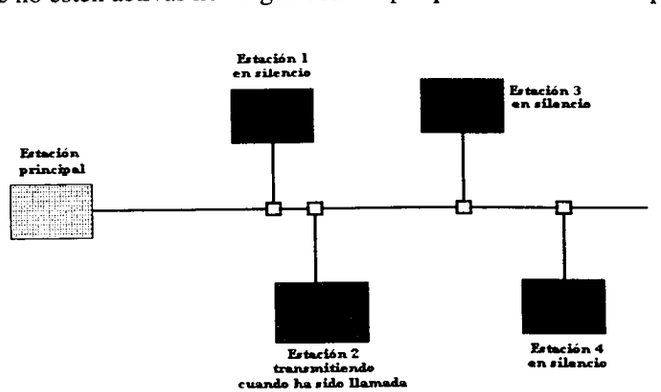
Cada estación secundaria dispone de una zona de almacenamiento temporal, donde envía el bloque de datos que desea transmitir.

La estación principal comprueba en cada una de las secundarias si alguna tiene algún bloque de datos para transmitir. Si en alguna de ellas encuentra uno, se autoriza a dicha estación para que lo transmita de forma inmediata o al cabo de un determinado tiempo. Si no tiene ningún bloque de datos pasa a revisar la estación siguiente y así sucesivamente.

Los bloques de datos se pueden enviar de dos formas distintas:

- Pasando por la estación principal, la cual los reenvía a la estación destino.
- Enviándolos directamente a la estación destino.

Se puede indicar que el control sobre las estaciones secundarias tenga el mismo nivel de prioridad para todas o bien que las estaciones que cuentan con una mayor actividad tengan una prioridad más alta. También se puede indicar que las estaciones que no estén activas no tengan control por parte de la estación principal.



Protocolo de llamada selectiva (Polling)

Este tipo de protocolo cuenta con algunas ventajas con respecto a los de contienda:

- La longitud de los bloques es superior.
- Soporta un mayor volumen de carga en la red.
- Permite trabajar con longitudes de red mayores.

Está recomendado para redes con carga media y para una longitud media o grande de la red.

Paso de testigo (*Token passing*)

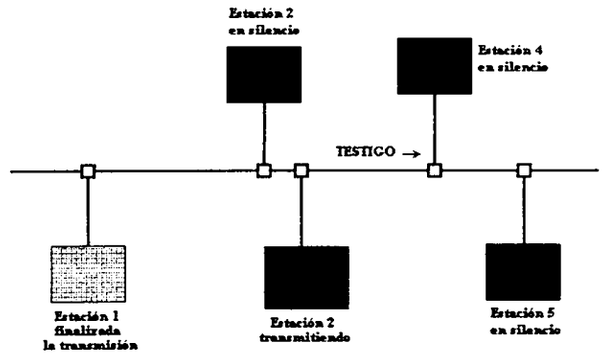
Este protocolo hace circular continuamente un grupo de *bits* (testigo) por la red. Este testigo está formado por una cabecera, un campo de datos y un campo final.

CABECERA	CAMPO DE DATOS	CAMPO FINAL

Cuando una estación quiere transmitir ha de esperar a que llegue hasta ella el testigo vacío. En ese momento le añade unos datos, quedando el testigo formado por: la cabecera, la dirección destino, la dirección origen, el camino que ha de seguir para llegar a su destino y el bloque de datos, y lo envía a su destino.

CABECERA	DIRECCIÓN DESTINO	DIRECCIÓN ORIGEN	CAMINO A SEGUIR	BLOQUE DE DATOS

Si la estación no desea transmitir, pasa el testigo vacío a la siguiente estación y así sucesivamente.



Protocolo de paso de testigo (Token passing)

El testigo ocupado llega a la estación destino que recoge el bloque de datos, pone una marca en el testigo indicando si lo acepta o lo rechaza por venir con errores, y lo devuelve a la estación que lo ha enviado.

Cuando llega a la estación que lo envió, ésta lo reenvía si llega con la marca de rechazado, envía el siguiente bloque de datos o vacía el testigo para que pase a la estación siguiente.

Este protocolo cuenta con bastantes ventajas:

- Elimina por completo el riesgo de colisiones.
- Puede emplear mensajes muy largos.
- El volumen de carga es bastante alto.
- El tamaño de la red puede ser grande.

Está recomendado para redes con volumen de carga medio o alto y para una longitud media o grande de la red.

CONTROL DE ERRORES

Debido a las interferencias, ruidos y distorsiones que aparecen en la línea, los datos al llegar a la estación destino pueden haber sufrido alguna modificación y no corresponder exactamente con los que fueron emitidos.

Para detectar estos errores se emplean diversas técnicas, que dependen del protocolo elegido.

Los métodos más utilizados para el control de los errores son:

- Método de paridad.
- Método de redundancia cíclica.

Método de paridad

Este método, también llamado geométrico, consiste en añadir un *bit* (*bit* de paridad) a cada uno de los caracteres enviados. Este *bit* debe tener el valor cero o uno, y será tal que haga que el carácter, contando el *bit* de paridad, tenga un número par de *bits* con valor uno (en el caso de la paridad par) o que tenga un número impar de unos (en el caso de la paridad impar).

La estación destino cuenta el número de *bits* uno de cada carácter recibido y, si el valor calculado coincide con el correspondiente a la paridad utilizada, la transmisión ha sido correcta, pero si no ha sido así, solicita a la estación emisora que repita el envío.

Este *bit de paridad* (par o impar) que se añade al final de cada carácter, también recibe los nombres de *bit de paridad transversal*, *bit de paridad vertical* o *comprobación de redundancia vertical* (*VRC*).

He aquí un ejemplo de paridad par en el que se indican en cursiva los *bits de paridad*:

11100010
00011101
11001001
01101100

Este método cuenta con el problema de que únicamente puede detectar el error si se ha modificado un solo *bit*. Pero si se modifica un número par de *bits*, no se detectará el error. Para evitar este problema, se puede incluir al final de cada paquete un *bit* de comprobación de error que hará que la suma de unos de cada columna de *bits* corresponda con la paridad par o impar que se está utilizando.

11100010
00011101
11001001
01101100
01011010

A este *bit* se le denomina *carácter de comprobación horizontal, suma de comprobación* (*checksum*), *paridad horizontal* o *comprobación de redundancia horizontal* (*LRC*).

Si se emplean la paridad vertical y la horizontal, se podrían llegar a detectar todos los errores de un *bit* que se produzcan.

Método de redundancia cíclica

Este método consiste en que la estación emisora agrega al final de cada bloque de datos, una información calculada de acuerdo con una fórmula polinómica, cuyas variables son los ceros y unos enviados en el bloque de datos (se divide el valor binario numérico total por un valor constante definido por el protocolo, se desecha el cociente y es el resto lo que se añade al final del bloque de datos).

La estación destino realiza el mismo cálculo. Si le produce el mismo resultado la transmisión es correcta, pero si no ha sido así, solicita a la estación emisora que repita el proceso.

Este método recibe el nombre de *Código de Redundancia Cíclica* (*CRC*) y a los valores añadidos al bloque de datos se les denomina *Carácter de Comprobación de Bloque* (*BCC*) o simplemente *Redundancia*.

La ventaja de este método estriba en que el número de *bits* que se añade a cada bloque de datos es mucho menor que el del método anterior.

Retransmisión de bloques erróneos

Normalmente, la estación destino no corrige los bloques de datos erróneos, sino que se limita a detectar la existencia del error, pidiéndole a la estación emisora que vuelva a emitir dicho bloque de datos. Para la retransmisión del bloque de datos erróneo existen dos técnicas:

- Parada y espera.
- Envío continuo.

PARADA Y ESPERA

Esta técnica consiste en que la estación emisora, después de enviar el bloque de datos, espera a recibir una contestación de confirmación o error del envío.

Si la transmisión es correcta, la estación receptora envía un mensaje de confirmación (*ACK*) y, si la transmisión es errónea, envía un mensaje de rechazo (*NAK*). Al recibir el mensaje de rechazo, la estación emisora procede a retransmitir el bloque de datos erróneo.

El inconveniente de esta técnica es el tiempo que pierde la estación emisora en esperar el mensaje de la estación receptora antes de proceder a un nuevo envío.

ENVÍO CONTINUO

Esta técnica consiste en que la estación emisora está enviando bloques de datos continuamente sin tener que permanecer a la espera de la confirmación de la estación receptora. Para poder realizarlo, identifica a cada bloque de datos con un código numérico.

Cuando se produce un error, la estación receptora solicita el reenvío del bloque erróneo y se pueden producir dos modalidades:

- **Envío continuo no selectivo**. En este modo, la estación emisora vuelve a retransmitir todos los bloques enviados desde aquel en el que se produjo el error. Esto provoca el reenvío de bloques que se podían haber recibido correctamente.

- **Envío continuo selectivo**. En este modo, la estación emisora vuelve a retransmitir únicamente aquel bloque en el que se produjo el error.

RECUPERACIÓN ANTE FALLOS

En el caso de que se produjera un envío de un bloque de datos, la estación emisora estará esperando el mensaje de confirmación o error, si la estación receptora se desconectara o se perdiera dicho mensaje, la estación emisora estaría esperando indefinidamente dicha contestación.

En ese caso, el protocolo debería proceder de la siguiente manera:

- Establecer un tiempo de espera de dicha contestación.

- Solicitar una nueva respuesta cuando haya finalizado dicho tiempo de espera.

- Limitar el número de intentos, después de los cuales el fallo se da por irrecuperable y finalizaría la transmisión de datos con dicha estación.