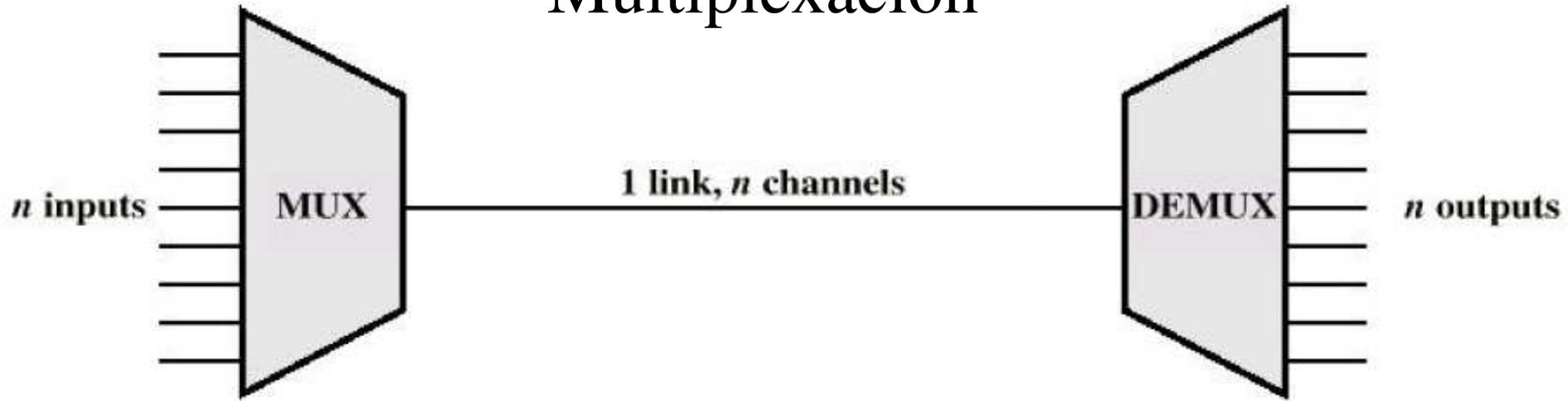
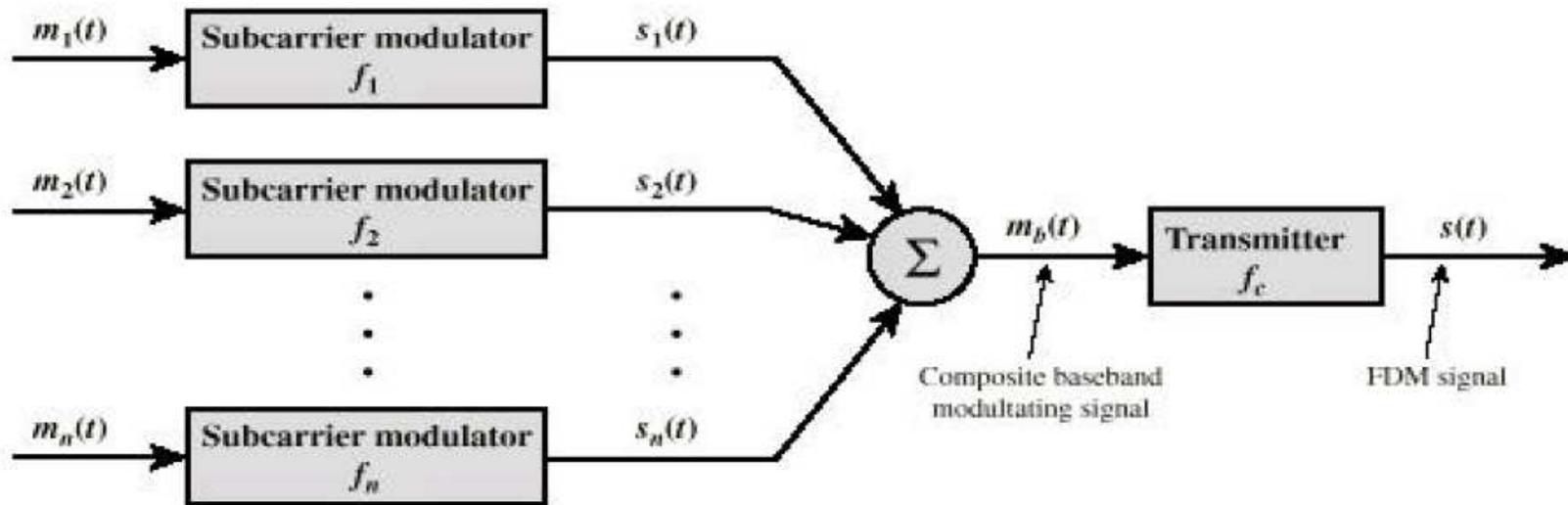


Multiplexación



- Objetivos -> compartir el medio
 - ♦ Un solo cable frente a muchos cables
 - ♦ Posibilidad de transmisión de varias señales donde de otro forma no se podría (p.e. por el aire)
 - ♦ Aprovechamiento del ancho de banda
- Tipos
 - ♦ Multiplexación por división en frecuencias (FDM).
 - ♦ Multiplexación por división en tiempo (TDM síncrona).
 - ♦ Multiplexación estadística por división en el tiempo (TDM estadística, asíncrona o inteligente).

Modulación por división en frecuencia (MDF)



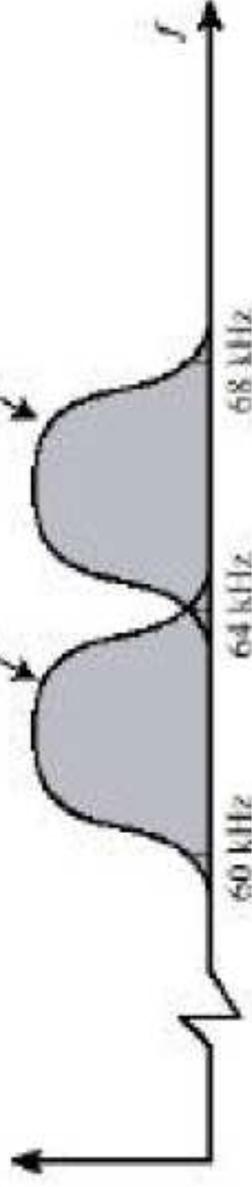
- Modulación \rightarrow desplazamiento de la señal a frecuencias altas
- Multiplexación \rightarrow suma de varias señales moduladas a frecuencias distintas
- Señales limitadas en banda \Rightarrow no hay solapamiento
- Válido para transmisión analógica y digital
- Ancho de banda total = suman anchos de banda

Señal en banda base



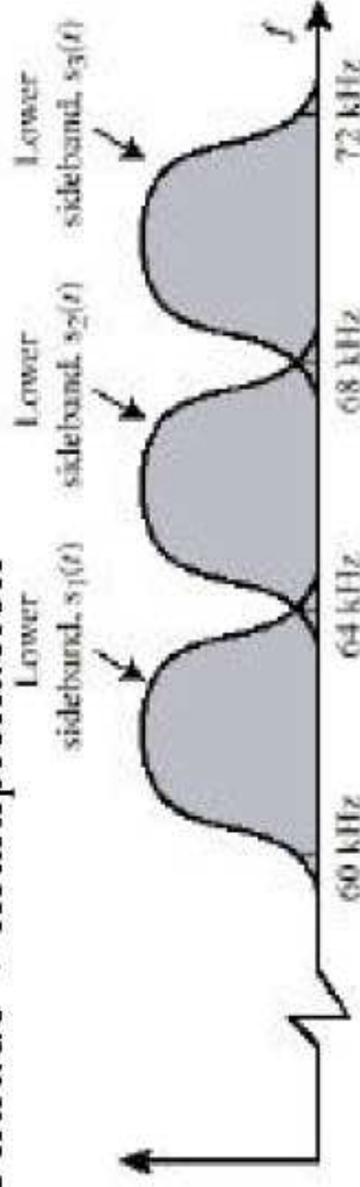
(a) Spectrum of $m_1(t)$, positive f

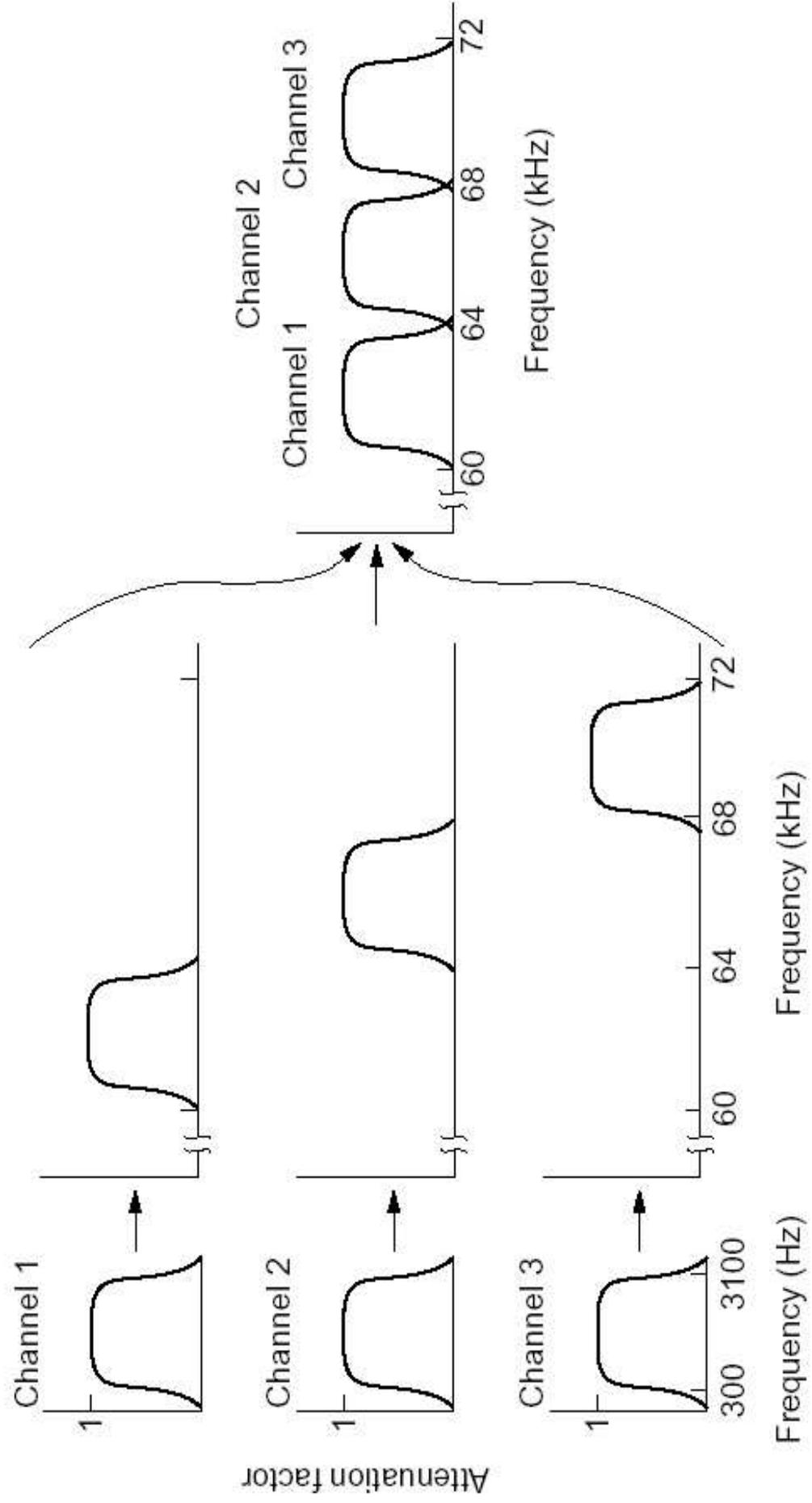
modulación



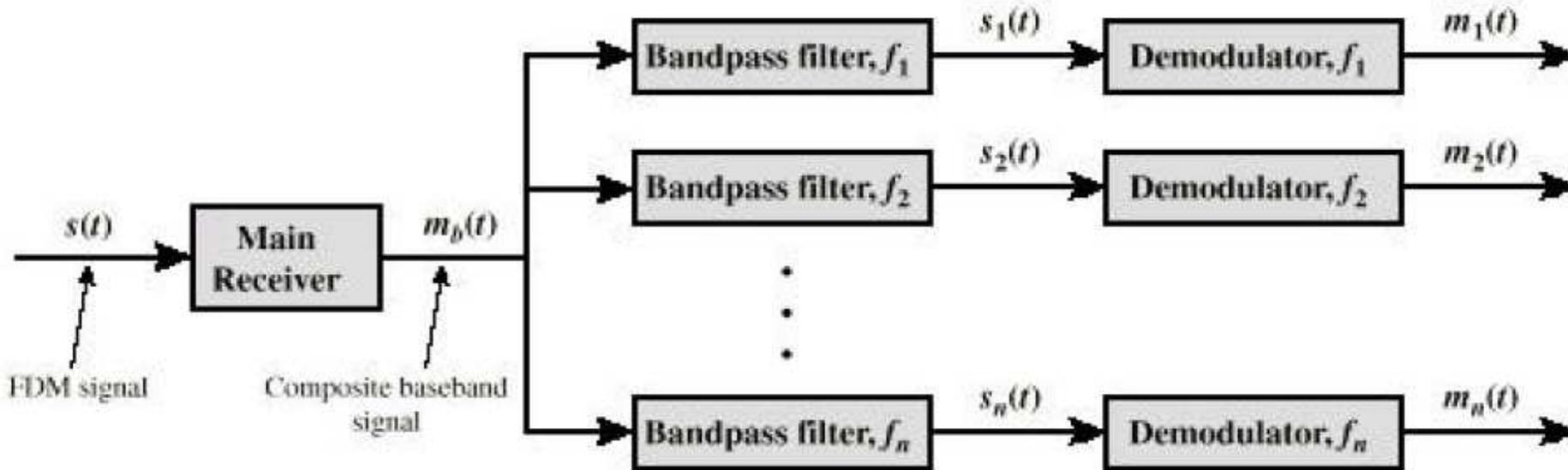
(b) Spectrum of $s_1(t)$ for $f_1 = 64$ kHz

Filtrado + multiplexación



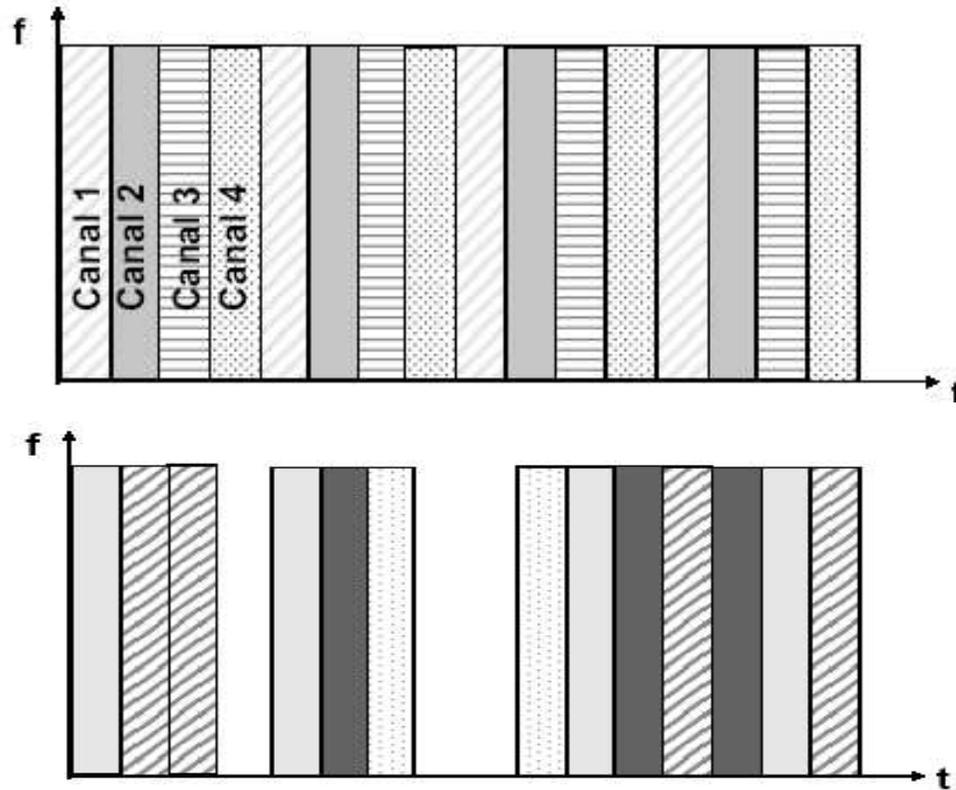


Recuperación de la señal



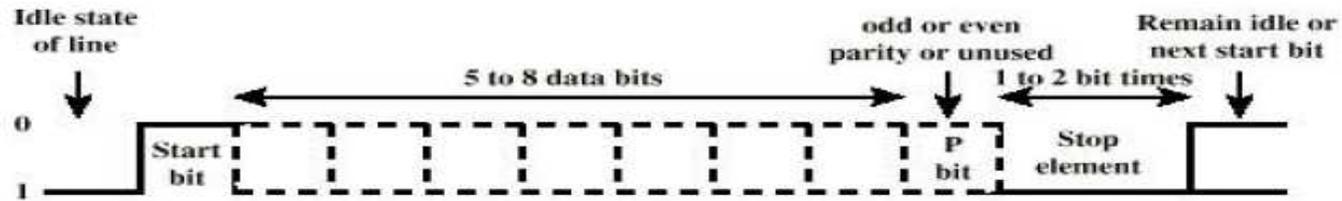
- Filtro P.Banda \rightarrow elimina todo menos un canal
- Demodulador \rightarrow desplaza a frecuencia baja \Rightarrow banda de base
- Problemas
 - ♦ Diafonía si los espectros de señales adyacentes se solapan demasiado.
 - ♦ Intermodulación en enlaces largos. Los amplificadores de un canal podrían generar frecuencias en otro canal.

Multiplexación por división en el tiempo (MDT)

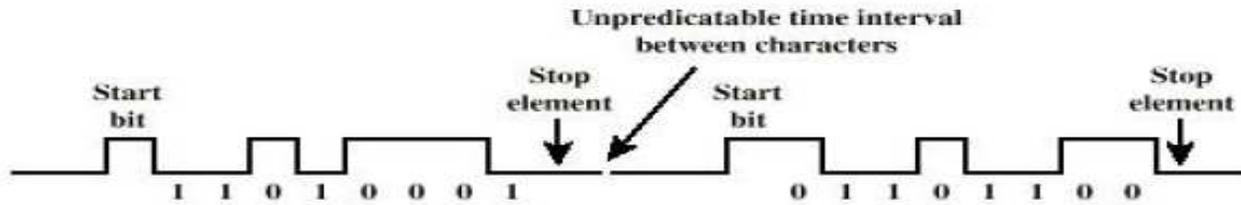


- ♦ Asignación de intervalos de canal (slots) a los distintos canales.
- ♦ MDT síncrona -> asignación fija de intervalos de canal => desperdicio de ancho de banda
- ♦ MDT asíncrona -> asignación variable según las necesidades => hay que identificar canales

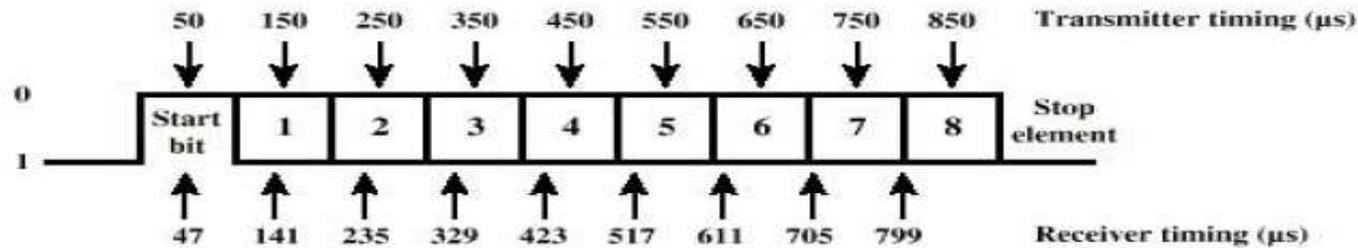
Transmisión asíncrona



(a) Character format



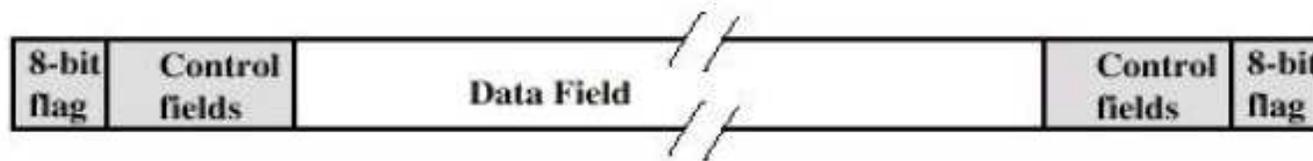
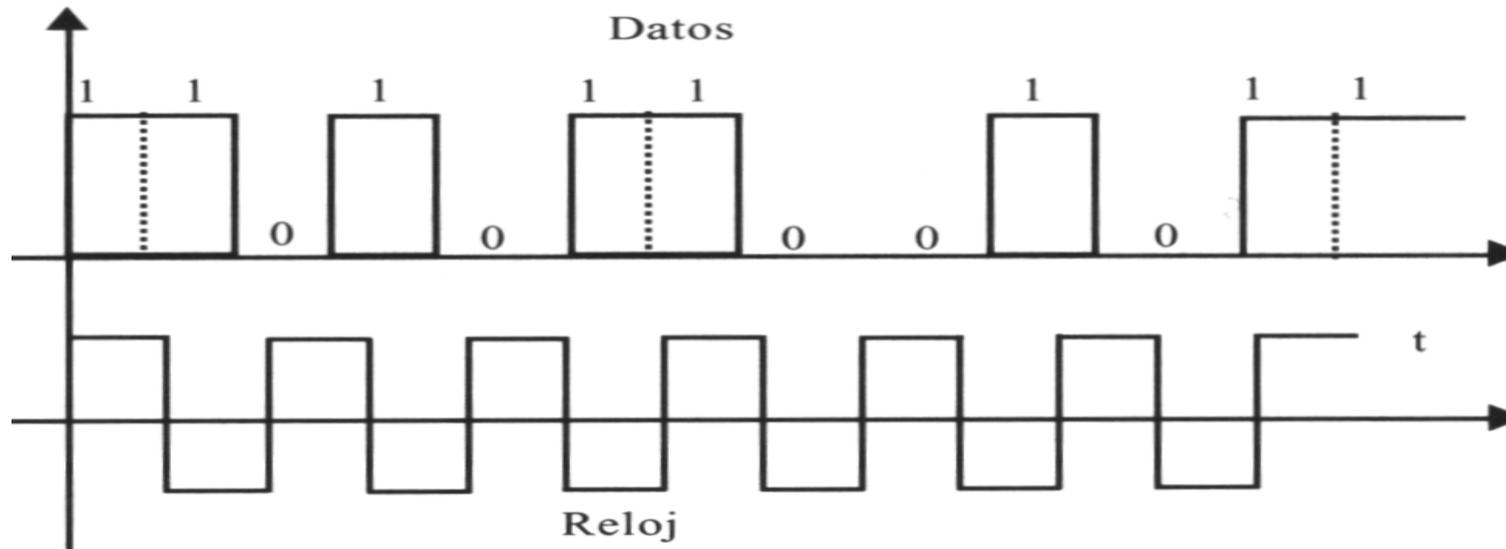
(b) 8-bit asynchronous character stream



(c) Effect of timing error

- Relojes distintos
- Errores de sincronización -> cadenas cortas.

Transmisión síncrona

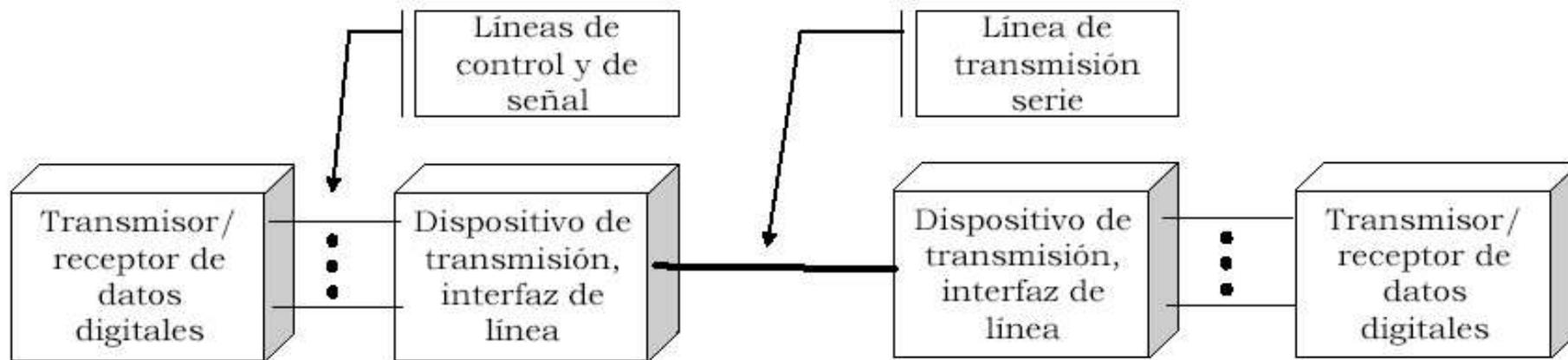


- Reloj
 - ♦ Por línea aparte
 - ♦ Incluido en la codificación (p.e. manchester)
- Menor sobrecarga de bits de control que en t. asíncrona.

Transmisión serie/paralelo

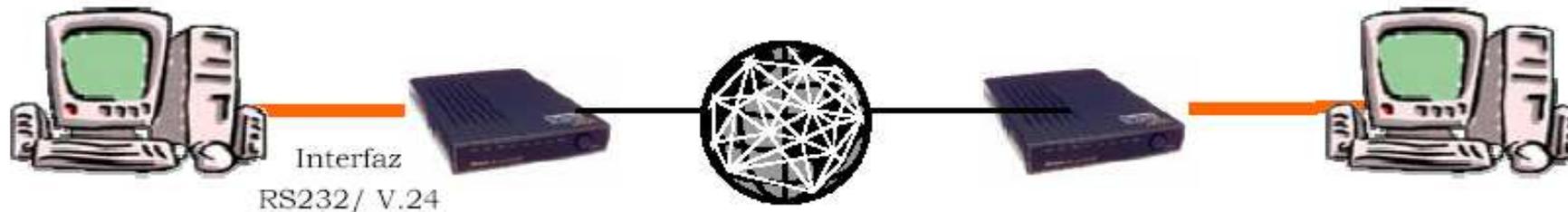
- Transmisión serie
 - ♦ Bit a bit
 - ♦ Menos hilos
 - ♦ Mayor complejidad: necesidad de un protocolo
 - ♦ Transmisión a larga distancia
- Transmisión paralelo
 - ♦ Varios bits a la vez
 - ♦ Mayor número de hilos
 - ♦ Más simple, sin protocolo o protocolo más sencillo
 - ♦ Transmisión a corta distancia

Interfaces para las comunicaciones de datos



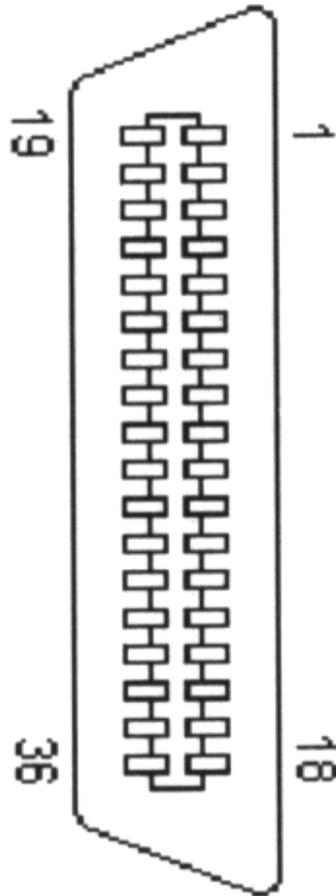
DTE: equipo terminal de datos

DCE: equipo terminación de circuito de datos



Interfaz Centronics

- 19 (R) DATA STROBE
- 20 (R) DATA BIT 1
- 21 (R) DATA BIT 2
- 22 (R) DATA BIT 3
- 23 (R) DATA BIT 4
- 24 (R) DATA BIT 5
- 25 (R) DATA BIT 6
- 26 (R) DATA BIT 7
- 27 (R) DATA BIT 8
- 28 (R) ACKNOWLEDGE
- 29 (R) BUSY
- 30 (R) INIT PRINTER
- 31 INIT PRINTER
- 32 ERROR
- 33 SIN DEFINIR (GND)
- 34 SIN DEFINIR
- 35 SIN DEFINIR
- 36 SIN DEFINIR



- 1 DATA STROBE
- 2 DATA BIT 1
- 3 DATA BIT 2
- 4 DATA BIT 3
- 5 DATA BIT 4
- 6 DATA BIT 5
- 7 DATA BIT 6
- 8 DATA BIT 7
- 9 DATA BIT 8
- 10 ACKNOWLEDGE
- 11 BUSY
- 12 PAPER END
- 13 SELECT
- 14 AUTO FEED
- 15 SIN ASIGNAR
- 16 LOGIC GND
- 17 CHASSIS GND
- 18 + 5v

(R) Retorno de la Señal

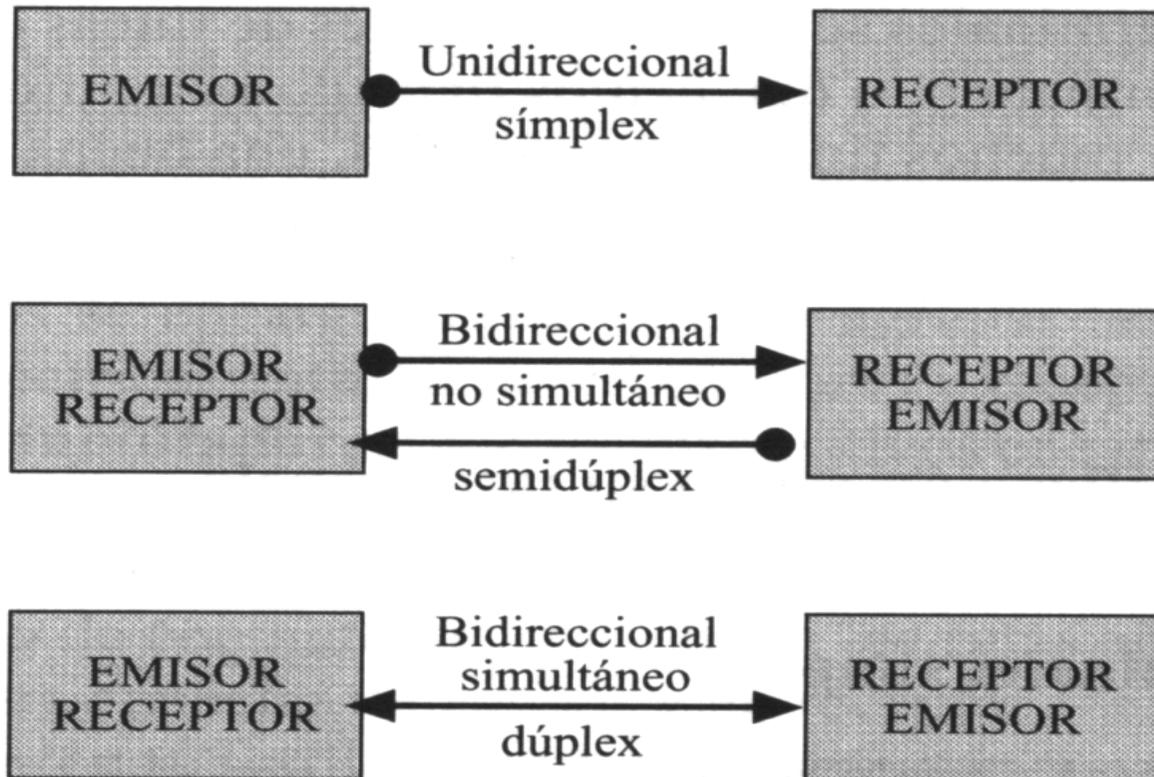
Interfaz RS-232



Señales RS-232 en un conector PC de 9 pines

PIN	SEÑAL	NOMBRE	FUNCIÓN
1	DCD	Data Carrier Detect	Detección de portadora
2	RD	Received Data	Entrada de datos en el DTE
3	TD	Transmitted Data	Salida de datos del DTE
4	DTR	Data Terminal Ready	DTE preparado y listo. Pone en funcionamiento al módem
5	GND	Masa	Masa del circuito
6	DSR	Data Set Ready	ETCD está listo para comunicar con DTE
7	RTS	Request To Send	DTE desea cambiar a modo de transmisión
8	CTS	Clear To Send	ETCD está listo para transmitir
9	RI	Ring Indicator	Aviso de llamada detectada

Modos de diálogo



Protocolos de comunicación

- Protocolo = conjunto de normas que hacen posible la comunicación entre dos o más nodos.
- Funciones más importantes de un protocolo:
 - ♦ Establecimiento y fin de la comunicación
 - ♦ Sincronización de la comunicación -> a nivel de bit, de palabra y de trama.
 - ♦ Direccionamiento -> identificación de los nodos
 - ♦ Control de flujo y de congestión -> permitir a la red compartir sus recursos entre varios nodos dando servicio a todos.
 - ♦ Control de errores -> códigos y sistemas para la detección y recuperación de errores.
 - ♦ Estrategias de encaminamiento -> utilización de los recursos de la red de forma óptima, caminos alternativos, etc.
- Arquitectura de protocolos
 - ♦ Procesos independientes
 - ♦ Implementación por software o hardware
 - ♦ Estructura en capas.

Clasificación de los protocolos

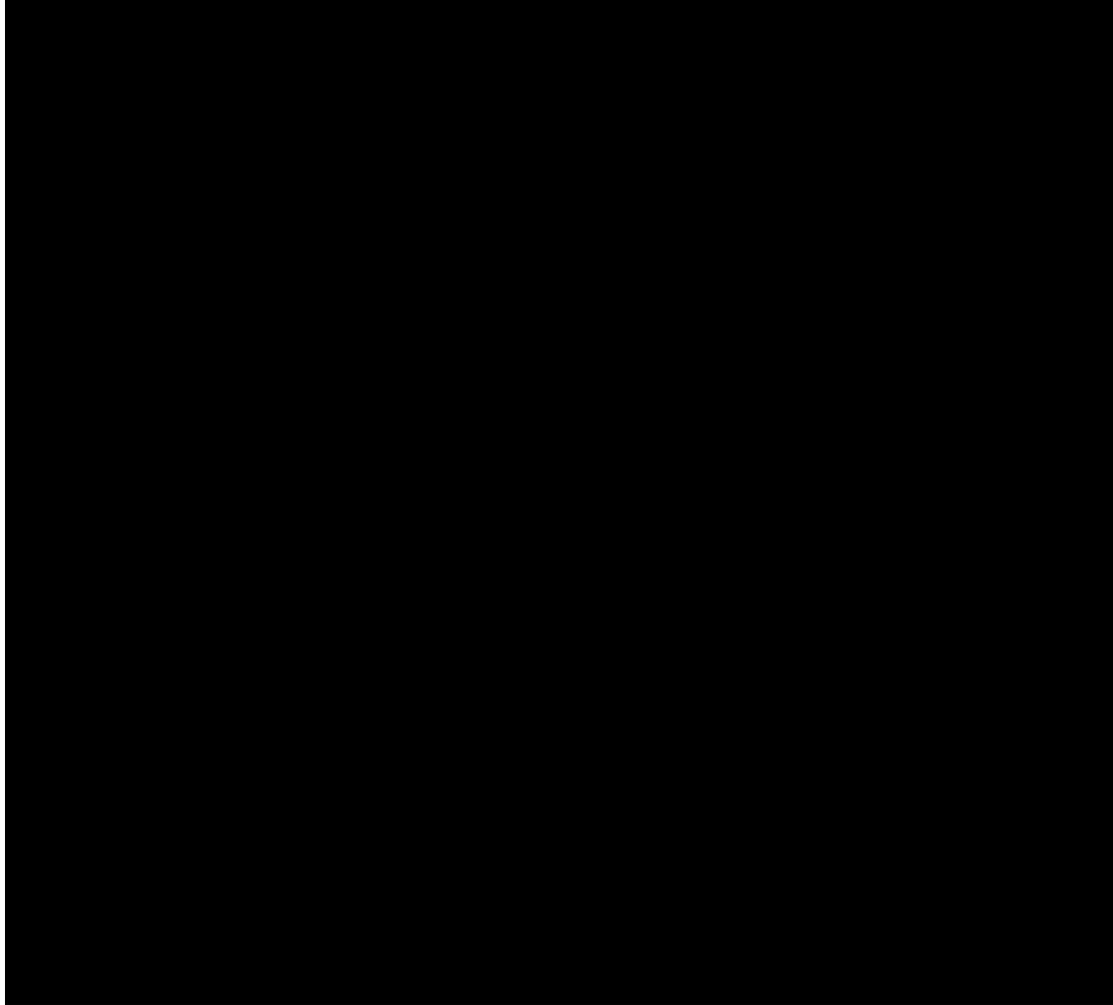
- Según las unidades de datos con las que trabajan
 - ♦ Protocolos orientados a carácter -> década de los 60
 - ♦ Protocolos orientados a bit -> modernos
- Según su forma de sincronización -> síncronos / asíncronos
- Según el control sobre el medio
 - ♦ Balanceados o simétricos:
 - los dos extremos trabajan igual.
 - Cada uno puede tomar la iniciativa de la comunicación
 - ♦ No balanceados a asimétricos
 - Una estación primaria (maestra) y las demás secundarias (esclavas)
 - La estación primaria emite y/o da turnos de palabra para emitir
 - La estación secundaria recibe o espera su turno para emitir
 - ♦ Híbridos

- Según utilicen o no sondeo
 - ♦ Protocolos de sondeo-selección
 - Sondeo = la estación primaria pide información a la secundaria
 - Selección = la estación primaria envía información a la estación secundaria
 - El proceso se controla con señales:
 - ✓ Sondeo = petición de información
 - ✓ Selección = aviso de envío de información
 - ✓ ACK = validación
 - ✓ NAK = no validación
 - ✓ EOT = fin de transmisión
 - ♦ Protocolos sin sondeo: no realizan sondeo
 - Control de flujo hardware: RTS/CTS
 - Control de flujo software: XON/XOFF
 - ✓

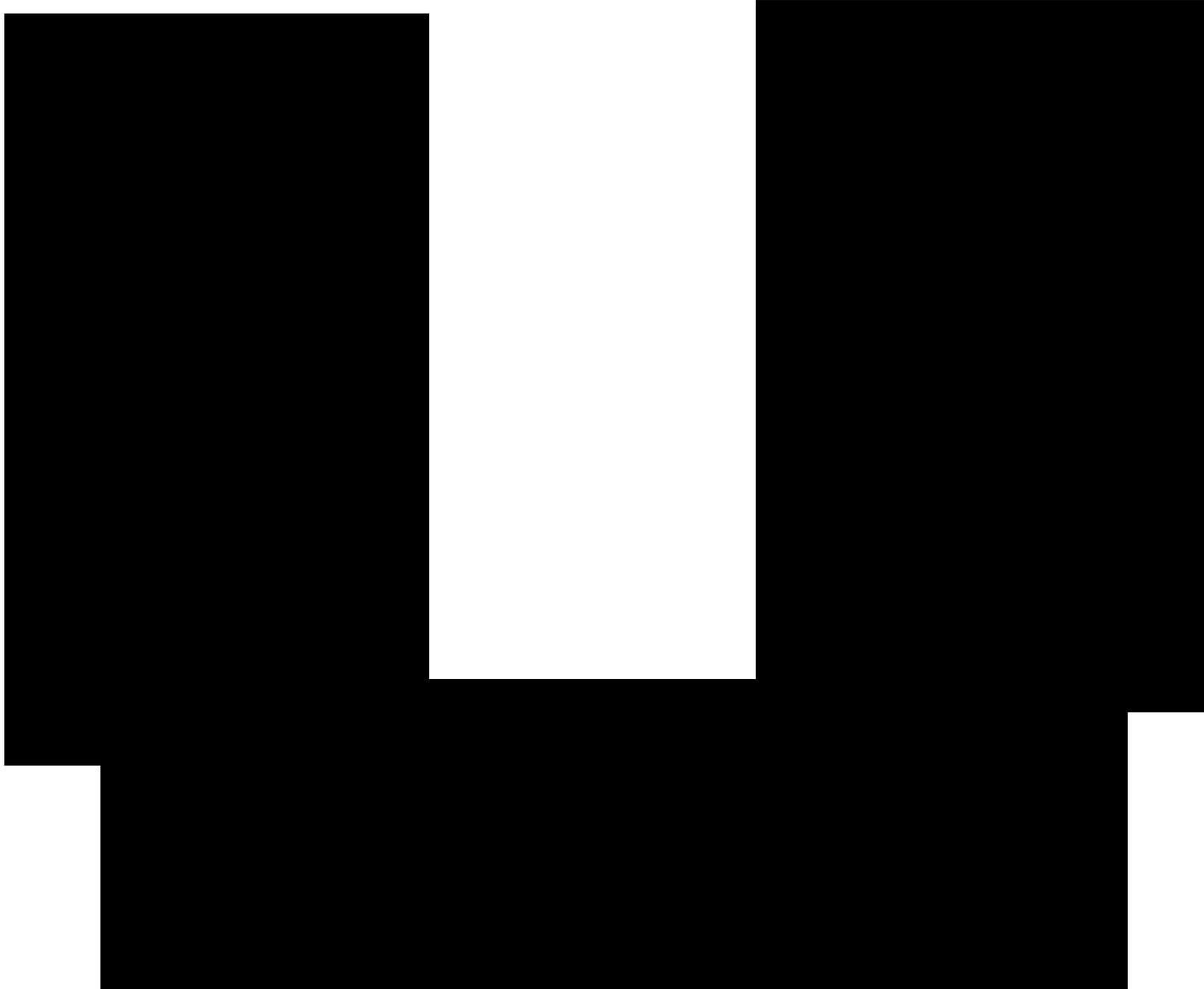
- Según utilicen o no prioridades
 - ♦ Sistemas sin prioridad
 - MUX-MDT (Multiplex por división en el tiempo)
 - ✓ El canal se divide en intervalos de tiempo
 - ✓ Se asigna un intervalo a cada estación
 - CSMA/CD (acceso múltiple por detección de portadora y detección de colisiones)
 - ✓ Todas las estaciones pueden utilizar el canal cuando está libre
 - ✓ Una estación escucha a ver si el canal está libre, y si está libre transmite
 - ✓ Si dos estaciones empiezan a emitir a la vez se produce una colisión. Cada estación corta el envío y espera un tiempo aleatorio antes de empezar a enviar de nuevo
 - ✓ El rendimiento se degrada en sistemas con mucho tráfico por el aumento de las colisiones
 - Paso de testigo
 - ✓ Se transmite por la red un testigo
 - ✓ Sólo la estación que tiene el testigo puede transmitir

- ♦ Sistemas con prioridad
 - CSMA/CD con prioridad
 - ✓ El tiempo de espera después de una colisión no es aleatorio sino que se fija para cada estación, menor cuanto mayor sea la prioridad de la estación
 - Paso de testigo con prioridad
 - ✓ El paso del testigo no se hace por turnos, sino que se puede reservar por las estaciones según su prioridad
- Protocolos de ventana deslizante
 - ♦ En protocolos normales (parada y espera) el canal permanece sin utilizar mientras se espera la validación del receptor
 - ♦ Los protocolos de ventana deslizante permiten enviar varias tramas sin esperar validación y validarlas luego todas a la vez
 - ♦ Llevan un contador de tramas transmitidas

Desperdicio de tiempo de canal en protocolos de parada y espera



Protocolos con ventana deslizante



- Clasificación según el nivel (OSI):
 - ♦ Protocolos de nivel físico (1)
 - ♦ Protocolos de nivel de enlace (2)
 - ♦ Protocolos de nivel de red (3)
 - ♦ Protocolos de nivel de transporte (4)
 - ♦ Protocolos de nivel de sesión (5)
 - ♦ Protocolos de nivel de presentación (6)
 - ♦ Protocolos de nivel de aplicación (7)

Protocolo XMODEM

- Protocolo para transferencia de archivos entre PC's a través del módem
- Protocolo de parada y espera, serie, asíncrono
- Tramas de longitud fija:

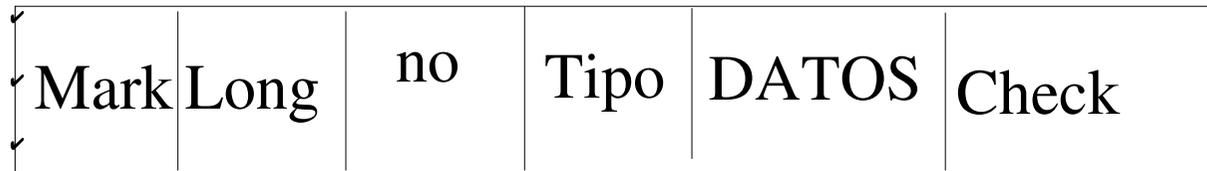
✓ SOH ✓	No	C1 no	DATOS	Checksum
---------------	----	----------	-------	----------

- SOH = cabecera (carácter 1 ASCII)
- no (1byte)= número de secuencia del paquete
- C1 no (1byte)= n° secuencia en complemento a 1
- DATOS (128 bytes)
- Checsum (1 byte) = suma de todos los bytes de datos

- Inicio de la transmisión -> receptor envía ACK indicando que está preparado para recibir
- Transmisión:
 - El emisor envía un dato
 - Si el receptor lo recibe bien envía ACK
 - Si hay error de secuencia en vía CAN -> corta la transmisión
 - Si hay otro error envía NACK -> el emisor reenvía la trama
- Fin de la transmisión -> el emisor envía EOT

Protocolo kermit

- Protocolo para transferencia de archivos entre ordenadores (no PCs) a través del módem
- Protocolo de parada y espera, serie, asíncrono
- Tramas de longitud variable:



- mark (1 byte) = cabecera (secuencia irrepetible)
- long (1 byte) = longitud de la trama
- no (1byte)= número de secuencia de la trama
- tipo = tipo de trama
- DATOS (longitud variable)
- Check (1,2,3 byte) = puede ser check o CRC

- Permite la transferencia de archivos entre diferentes sistemas
- Sólo presupone que los sistemas son capaces de enviar caracteres imprimibles (20h-7Fh ASCII)
- Los "códigos de control" son tramas en vez de caracteres
- Tramas de longitud variable
- El protocolo incluye el nombre del fichero
- Permite negociar parámetros de la comunicación
- Permite versiones de ventana deslizante (nº de secuencia en tramas ACK y NACK)
- Permite transferir múltiples ficheros

Protocolo HDLC

- Protocolo orientado a bit, síncrono, punto a punto o multipunto, de ventana deslizante.
- Estandar ISO.
- Permite explotación duplex del enlace.
- Permite la transmisión de cualquier tipo de datos.
- Permite enlaces equilibrados y no equilibrados.
- Trama:



- ♦ Bandera = 01111110
- ♦ Dirección (8bits) = identifica estación (multipunto)
- ♦ Control (8bits) = tipo de trama, etc
- ♦ DATOS = cualquier número de bits
- ♦ FCS (16 bits) = control de errores