

UNIVERSIDAD DE SEVILLA

Escuela Técnica Superior de

Ingeniería Informática

PRÁCTICA 2:

COMUNICACIÓN ASINCRONA

**Tecnología Básica de las Comunicaciones
(Ingeniería Técnica en Informática de Sistemas)**



Departamento de Tecnología Electrónica



PRACTICA 2.- COMUNICACIÓN ASÍNCRONA

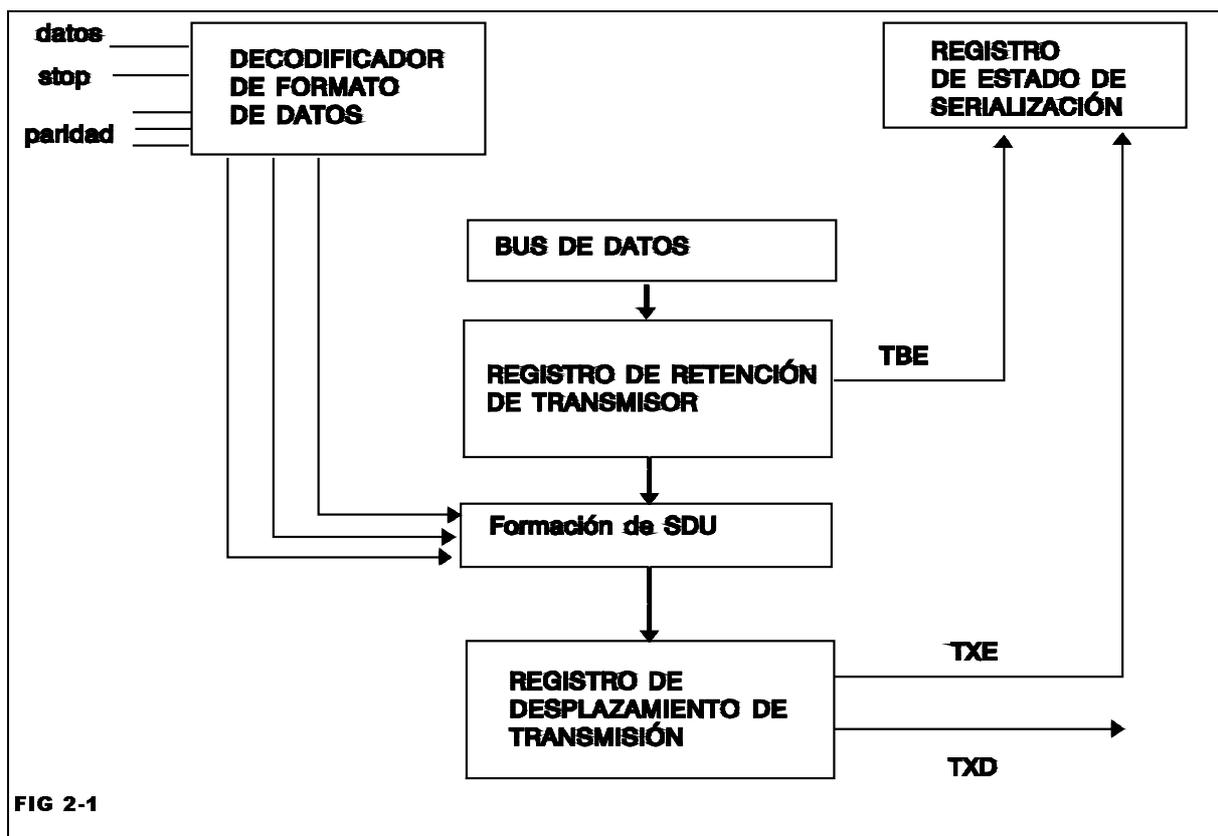
1.-Objetivos

- Realizar varios programas para transmitir datos con diferentes formatos asíncronos y diferentes velocidades, comprobando su correcto funcionamiento en el laboratorio.
- Representar y visualizar la señal que se transmite en el dominio temporal, mediante el osciloscopio, comprobando su velocidad binaria y tiempo de bit.
- Comprobar los niveles de señal que facilita el interfaz eléctrico RS-232 C y razonar que ventajas ofrecen el variar los niveles de tensión en una comunicación de DATOS.

2.-Estudio Teórico

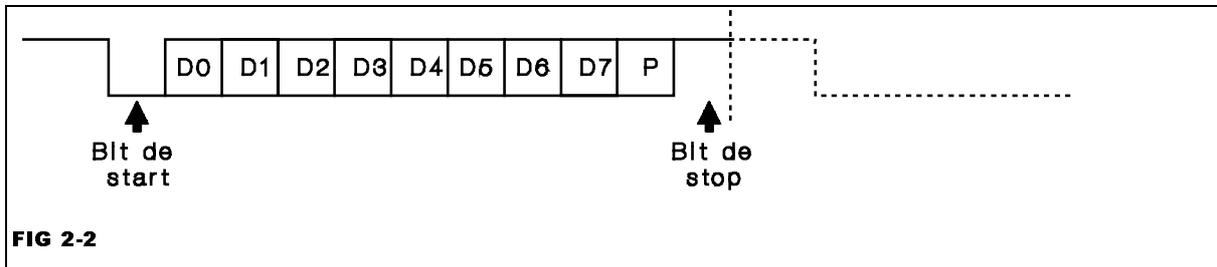
• Descripción de la transmisión serie asíncrona

Un transmisor serie asíncrono debe ser capaz de recibir los datos del procesador a través de un bus y enviarlos de forma serie al exterior. Para ello el componente esencial de un transmisor serie es un registro de desplazamiento. Este registro no es accesible directamente. El procesador carga el byte a transmitir en un registro llamado Registro de Mantenimiento de la Transmisión (THRE), que está conectado a las entradas del registro de desplazamiento (TSR). La carga del dato en el registro de



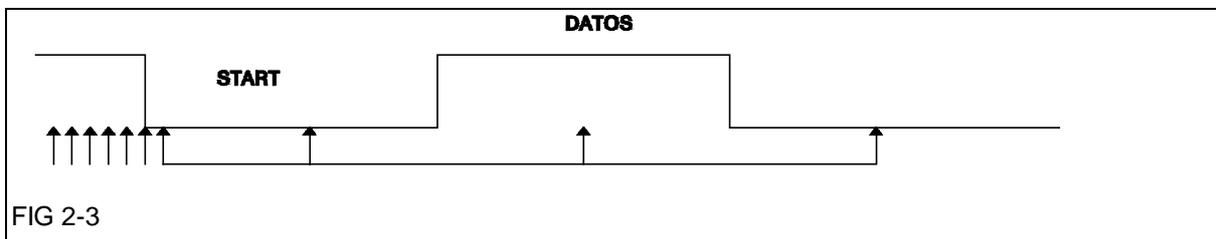
desplazamiento se realiza de forma automática cuando éste se queda vacío (figura 2-1).

El protocolo asíncrono está orientado al carácter, cada byte que se transmite se rodea de un grupo de bits necesarios para la comunicación y que describiremos a continuación. Cuando no hay datos que transmitir la línea o salida del transmisor se encuentra en el estado lógico 1 "MARK". Si existe transmisión de algún carácter la línea podrá tener la forma que se representa en la figura 2-2.



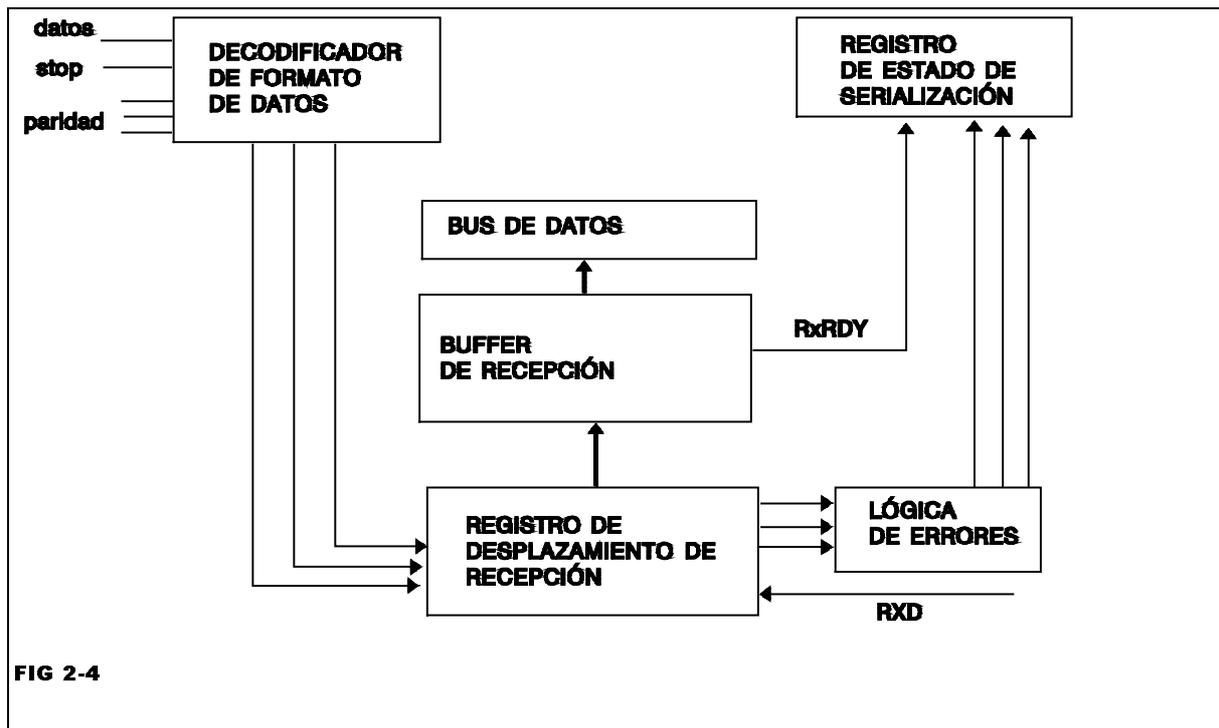
Previo a la transmisión del byte, la línea se pone a 0 lógico durante el tiempo de un bit, esto es el bit de start. Esto se utiliza para avisar y sincronizar al receptor de la existencia de un dato. A continuación van los bits del carácter comenzando por el menos significativo, el bit de paridad (opcional) cuyo objetivo es el control de errores y, en último lugar, el bit de stop que marca el final de un carácter.

En la UART se emplean dos señales de reloj separadas: una para controlar las operaciones internas y otra para controlar las operaciones de desplazamiento en la secciones de transmisión y recepción. La última, dependerá de la velocidad a la que se quiera realizar la transmisión. La señal de reloj interna está diseñada para que su velocidad sea varias veces superior a la velocidad de transmisión (de 16 a 64 veces superior). El detector del bit de START se encuentra muestreando a la frecuencia interna del reloj, cuando detecta una cambio a 0, en el caso de que la velocidad de transmisión sea 16 veces inferior, volverá a comprobar la entrada 8 ticks de reloj más tarde, si la línea se encuentra a 1, esto implica que se trata de ruido, en caso contrario es el bit de start. En tal situación, y a partir de este momento, se muestrea



cada 16 ticks de reloj, lo cual hace que se tomen los valores de entrada en el punto medio de cada bit (figura 2-3).

La estructura general del receptor UART se muestra en la figura 2-4. Cuando todos los bits que constituyen la trama han sido cargados en el registro de desplazamiento, se construye el byte de datos válido y se pasa al buffer de recepción, activándose el flag RxRDY, que indica que el dato ya está listo. En el proceso de recepción pueden producirse errores (lo veremos más adelante en este documento), de los que se informa al registro de serialización.



- UART (Descripción funcional interna)

El nombre de UART corresponde a las siglas de Universal Asynchronous Receiver Transceiver que quiere decir Transmisor Receptor Asíncrono Universal. En los PC XT y AT este dispositivo lo constituye el circuito integrado 8250 (o el actual 16C450) que estudiaremos a continuación. Para el resto de los modelos, `386 y `486, se suelen utilizar otros integrados, pero manteniendo la compatibilidad con el anterior, lo cual justifica que nos centremos en el estudio del 8250. Este circuito contiene un conjunto de registros que gobiernan su funcionamiento y que tienen asignados una dirección específica en el espacio de entrada/salida del procesador. Esto implica que el control sobre la UART y la transmisión de datos, desde el punto de vista del programador, solo será escribir o leer en posiciones de I/O determinadas. Está claro que deberemos estudiar las funciones de cada uno de estos registros y su localización en el mapa de I/O; esta es la tarea que se desarrolla a continuación.

El PC, por defecto, posee dos puertos para la comunicación serie: el COM1, y el COM2. En la siguiente tabla se muestran los registros internos de la UART y su dirección correspondiente al COM1 y al COM2

COM1	COM2	ACCESO	REGISTROS	DLAB
0X3F8	0X2F8	W	TX BUFFER	0
0X3F8	0X2F8	R	RX BUFFER	0
0X3F8	0X2F8	R/W	DIVISOR LSB	1
0X3F9	0X2F9	R/W	DIVISOR MSB	1
0X3F9	0X2F9	R/W	IER	0
0X3FA	0X2FA	R/W	IIR	X
0X3FB	0X2FB	R/W	LCR	X
0X3FC	0X2FC	R/W	MCR	X
0X3FD	0X2FD	R/W	LSR	X
0X3FE	0X2FE	R/W	MSR	x

• **Registro de Control de Línea(LCR)**

Este registro gobierna el formato de la comunicación asíncrona.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DLAB	BRK	SP	EPS	PEN	STB	WLS1	WLS0

Bit 0 y 1.- Estos dos bits especifican el número de bits en cada carácter recibido o transmitido. El número de bits dependerá de WLS1 y WLS0 según muestra la siguiente tabla

WLS1	WLS0	
0	0	5 BITS
0	1	6 BITS
1	0	7 BITS
1	1	8 BITS

Bit 2.- Este bit especifica el número de bits de stop que serán transmitidos o comprobados en la recepción. Si STB=0, se envía o se comprueba en recepción un bit de stop. Si STB=1 y el tamaño del carácter es de 5bits, se envían y comprueba 1'5 bits de stop, en caso contrario 2 bits.

Este es el bit de habilitación de paridad. Si PEN= 1 lógico se generará el bit de paridad para los datos a transmitir y se chequeará para los datos recibidos.

Bit 4.- Este es el bit de selección de tipo de paridad. Si PEN = 1 lógico y este bit está a 0 lógico, se genera y chequea el bit de paridad impar; si EPS=1 se genera y chequea el bit de paridad par.

Bit 5.- Este bit permite transmitir el bit de paridad siempre a 1 o a 0. Si SP=1 y PEN=1, se transmite o chequea un 0 si EPS=1 o un 1 si EPS=0.

SP	EPS	PEN	OPERACIÓN
X	X	0	Sin operación
1	1	1	Siempre 0
1	0	1	Siempre 1
0	0	1	Paridad impar
0	1	1	Paridad par

Bit 6.- Este es el bit de control de Break(ruptura). Cuando está a 1 lógico la salida de transmisión se pone a 0 lógico independientemente de cualquier otra actividad. Si el bit de 6 se pone a 0 lógico desaparecerá la condición de Break. La utilidad de esto reside en la posibilidad que tiene el procesador emisor de alertar al sistema remoto.

Bit 7.- Este es el bit de acceso al latch divisor. Debe ser 1 para acceder a los latches divisores del generador de velocidad. Debe ser 0 para acceder al registro de recepción en operación de lectura o al registro de transmisión en operación de escritura.

• Generador de velocidad programable

El 8250 posee un generador de velocidad programable capaz de dividir la frecuencia del reloj de entrada (1,8432 Mhz) mediante un divisor, desde 1 a 216 - 1. La frecuencia de salida del generador es de 16 x velocidad en b.p.s. escogida. Por tanto el contenido del latch divisor será:

$$divisor = \frac{frecuencia(1,8432Mhz)}{16 * velocidad}$$

Existen dos latches de 8 bits para almacenar el contenido del divisor. Para poder acceder al latch más significativo tendremos que poner a 1 el bit DLAB del registro de control de línea y leer o escribir en la dirección 0x2F9, para el latch de menor peso haremos lo mismo pero en la dirección 0x2F8.

Para generar una velocidad de transmisión de 9600 baudios, tendremos que escribir en el latch divisor

DIVISOR LATCH LSB = 0x0C
DIVISOR LATCH MSB = 0x00

• **Registro de estado de línea (LSR)**

Este registro informa al procesador sobre el estado de la comunicación.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	TSRE	THRE	BI	FE	PE	OR	DR

Bit 0.- (*Data Ready*). Este bit informa al procesador de la existencia de un dato en el buffer de recepción si está a 1 lógico. Este bit se borra cuando se realiza una operación de lectura en el registro de recepción o cuando se escribe expresamente un 0 en esta posición.

Bit 1.- (*Overrun error*). Este bit informa de la sobre escritura de un dato almacenado en el buffer o registro de recepción. Se borra cuando el procesador lee el registro LSR.

Bit 2.- (*Parity error*). Este bit se pone a 1 lógico cuando el carácter recibido no tiene el bit de paridad correcto en función de lo programado en el LCR. Este bit se pone a 0 cuando se lee este registro.

Bit 3.- (*Framming error*). Este bit se pone a 1 lógico cuando el número de bits de stop del carácter recibido no corresponde con el esperado. Este bit se pone a 0 cuando se lee este registro.

Bit 4.- (*Break indication*). Este bit informa de la recepción de una condición de Break del transmisor. Recordamos que la condición de break se genera cuando la línea se encuentra a 0 lógico durante un tiempo superior al tiempo de transmisión del carácter incluido los bits del protocolo asíncrono; esto es, bit start + carácter + bit paridad + bit de stop.

Bit 5.- Este bit es el indicador de buffer de transmisión vacío. Cuando se encuentra a 1 lógico el procesador puede cargar datos en él para su posterior transmisión.

Bit 6.- Este es el indicador de registro de desplazamiento de transmisión vacío. Si está a 1 lógico indica que este registro está en reposo. Se pone a 0 lógico en una transferencia desde el registro de transmisión.

Bit 7.- Siempre a 0.

• **Registro de Control de Modem (MCR)**

Este registro permite el control con un modem o algún dispositivo periférico que emule a un modem.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	-	LOOP	OUT2	OUT1	RTS	DTR

Bit 0.- (*Data Terminal Ready*). Este bit controla la salida /DTR. Si está a 0 lógico la salida será un 0 lógico, si es un 1 la salida corresponderá a un 1.

Bit 1.- (*Request to Send*). Este bit controla la salida /RTS del mismo modo que el descrito anteriormente para RTS.

Bit 2,3.- Controlan las salidas /OUT1 y /OUT2, respectivamente, del mismo modo que los bits anteriores.

Bit 4.- (*Loop*). Este bit provee la posibilidad de autodiagnóstico del circuito. Opera de la siguiente manera: la salida de datos serie del circuito se desconecta. Internamente la salida del registro de Tx se conecta con la entrada del registro de recepción. Asimismo las 4 entradas de control del modem (/CTS,/DTS,/RLSD, y /RI) son desconectadas; las 4 salidas del MCR son conectadas a las cuatro entradas

de control de modem anteriores.

• Registro de Estado de Modem (MSR)

Este registro contiene 4 bits correspondientes al estado de las líneas de control del modem al procesador. Además de estas líneas existen otros 4 bits que muestran información sobre el cambio de las líneas anteriores. Estos bits se ponen a 1 cuando las líneas de entrada cambian de valor y se ponen a 0 lógico cuando el procesador accede a este registro.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CTS	DSR	RI	RLD	DCTS	DDSR	TER	DRLD

Bit 0.- Este bit es el *Delta Clear to Send indicator*. Este bit indica que la entrada CTS ha cambiado de estado desde la última vez que el procesador accedió a este registro.

Bit 1.- Este bit es el *Delta Data Set Ready indicator*. Este bit indica que la entrada DSR ha cambiado de estado desde la última vez que el procesador accedió a este registro.

Bit 2.- Este bit es el *Trailing Edge of Ring Indicator*. Este bit indica que la entrada RI ha cambiado de 1 lógico (ON) a 0 lógico (OFF).

Bit 3.- Este bit es el *Delta Received Line Signal Detector*. Este bit indica que la entrada RLSD ha cambiado de estado.

NOTA: Cuando los bits 0,1,2 o 3 están a 1 lógico se genera una interrupción de estado del modem.

Bit 4.- Este bit es el complemento de la entrada /CTS. Si el bit 4 del MCR está puesto a 1 lógico, este bit es equivalente a RTS en MCR.

Bit 5.- Este bit es el complemento de la entrada /DSR. Si el bit 4 del MCR está a 1 lógico este bit es equivalente a DTR en MCR.

Bit 6.- Este bit es el complemento de la entrada /RI. Si el bit 4 del MCR está puesto a 1 lógico, este bit es equivalente a OUT1 en MCR.

Bit 7.- Este bit es el complemento de la entrada /RLSD. Si el bit 4 del MCR está a 1 lógico este bit es equivalente a OUT2 en MCR.

• **Registro de habilitación de interrupciones (IER)**

Este registro habilita los cuatro tipos de interrupción que permite el 8250 y que generan la activación de la línea INTRPT del dispositivo. Es posible inhabilitar todo el sistema de interrupción o seleccionar los tipos de interrupción deseados. El contenido del registro de interrupción es:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	EMSI	ERLS	ETI	EDAI

Bit 0.- Este bit habilita, cuando está a 1 lógico, la interrupción de dato disponible.

Bit 1.- Este bit habilita, cuando está a 1 lógico, la interrupción de registro de transmisión vacío.

Bit 2.- Este bit habilita, cuando está a 1 lógico, la interrupción de estado de la línea de recepción.

Bit 3.- Este bit habilita, cuando está a 1 lógico, la interrupción de estado del modem.

• **Registro de identificación de interrupción (IIR)**

El 8250 prioriza interrupciones en cuatro niveles: estado de línea del receptor (prioridad 1), dato recibido disponible (prioridad 2), registro de transmisión vacío (prioridad 3) y estado del modem (prioridad 4). El IIR, cuando es accedido libera la interrupción pendiente de mayor prioridad; ninguna otra es reconocida hasta que la interrupción particular sea servida por el procesador. El contenido del IIR se muestra a continuación.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	ID1	ID0	IP

Bit 0.- Este bit se usa para indicar si existe una interrupción pendiente cuando está a 0 lógico. Cuando esto ocurra, el contenido del IIR puede ser utilizado como puntero a la rutina de interrupción apropiada. Cuando este bit se encuentra a 1 lógico no existen interrupciones pendientes.

Bit 1 y 2.- Estos dos bits se utilizan para identificar la interrupción pendiente de mayor prioridad. A continuación se muestra la relación de interrupciones en función del contenido de este registro, las causas que las provocan y los modos de limpiarlas.

Si [IR]=1 no existe ninguna interrupción pendiente y por tanto no se genera la activación de la línea INTRP.

Si [IR]=6 Existe la interrupción de mayor prioridad que es la generada por el estado de la línea del receptor. Las fuentes de esta interrupción son: Error de overrun, error de paridad, error de trama o condición break. Leyendo el registro LSR se elimina la fuente de interrupción.

Si [IR]=4 Existe la interrupción de segundo nivel de prioridad que es provocada por la recepción de datos en el registro de recepción. Leyendo el buffer de recepción se elimina esta interrupción.

Si [IR]=2 Existe la interrupción de tercer nivel de prioridad que es provocada al quedarse vacío el registro de transmisión. Para eliminar la interrupción hay que mandar un dato al registro de transmisión o bien leer el registro IIR.

Si [IR]=0 Existe la interrupción de menor prioridad que es provocada por el cambio en las líneas de entrada de control del modem. Se limpia la interrupción leyendo el MSR.

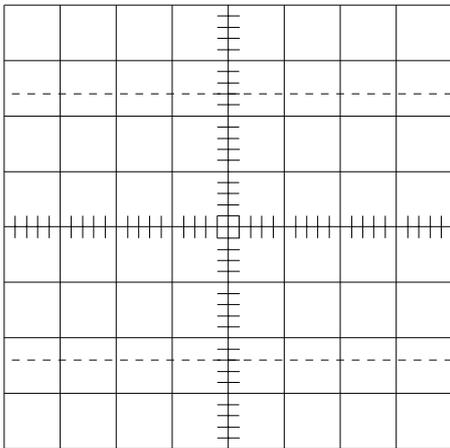
NOMBRE Y APELLIDOS _____

GRUPO _____

3.-Estudio previo

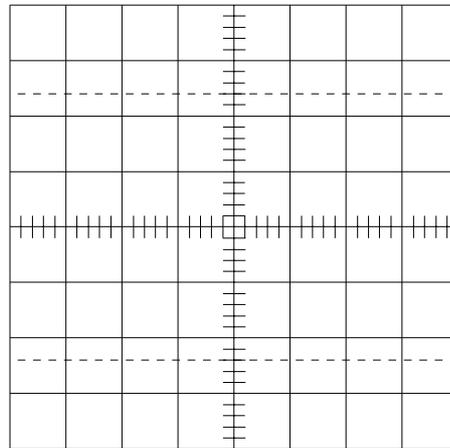
1.- Una determinada UART de un PC está configurada a 600bps, 7 bit, paridad impar, 2 bits de stop y transmitiendo de forma continua el carácter 'A'. Representa en la siguiente pantalla de osciloscopio lo que se vería a la salida del puerto si se conecta la sonda de dicho aparato en él y teniendo en cuenta que un 0 lógico es 12V y un 1 lógico -12V. Determina las unidades para el eje vertical y el horizontal. Realiza la misma operación para velocidades de 1200bps, 2400bps y 4800bps.

Escala eje Y _____ V/div



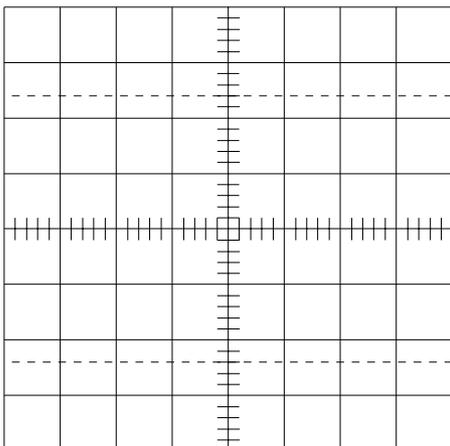
Escala eje X _____ s/div

Escala eje Y _____ V/div



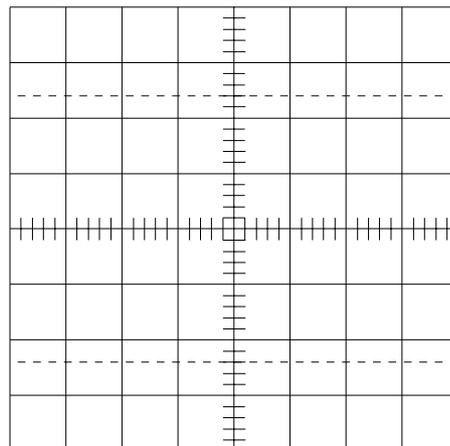
Escala eje X _____ s/div

Escala eje Y _____ V/div



Escala eje X _____ s/div

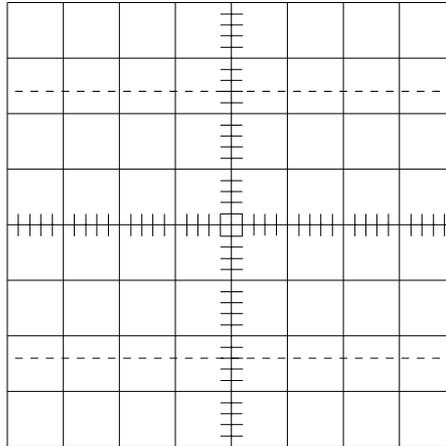
Escala eje Y _____ V/div



Escala eje X _____ s/div

2.- Repite el apartado anterior para 1200 bps, 7 bits de datos, paridad par y 1 bit de stop y transmitiendo de forma continua el carácter 'R'.

Escala eje Y _____ V/div



Escala eje X _____ s/div

3.- Realiza una función en lenguaje de programación C, con la configuración del ejercicio 1, para que transmita y visualice el carácter recibido en pantalla, sabiendo que está en bucle los datos transmitidos con los recibidos en el conector de salida.

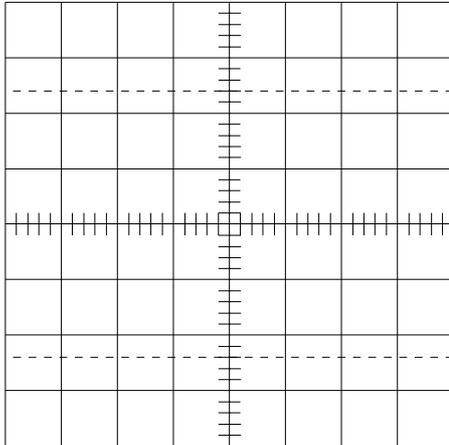
4.- Repite el apartado anterior, con la configuración del ejercicio 2.

4.- Realización práctica

1. Ejecuta el programa propuesto en el punto 3 del estudio previo y comprueba, utilizando un

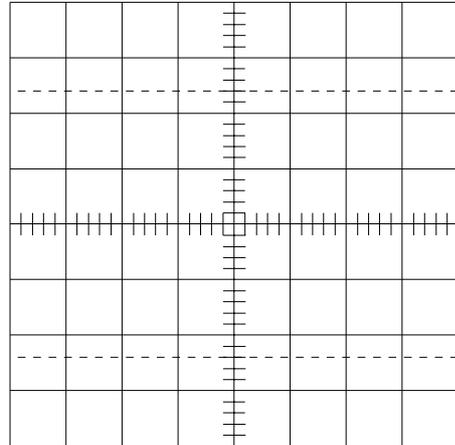
osciloscopio, la salida de la UART para los casos prácticos del apartado 1 del estudio anterior. Además dibuje en las plantillas siguientes, la salida mostrada por el osciloscopio.

Escala eje Y _____ V/div



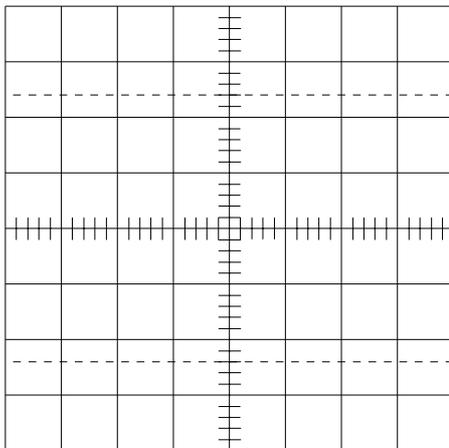
Escala eje X _____ s/div

Escala eje Y _____ V/div



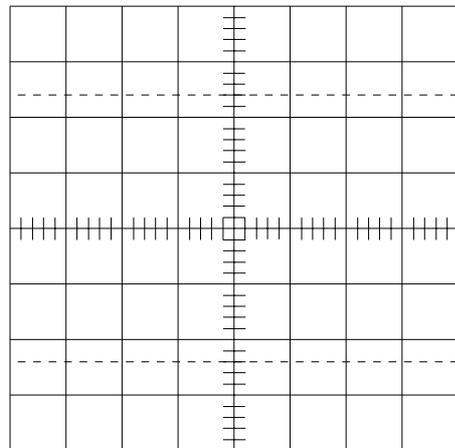
Escala eje X _____ s/div

Escala eje Y _____ V/div



Escala eje X _____ s/div

Escala eje Y _____ V/div



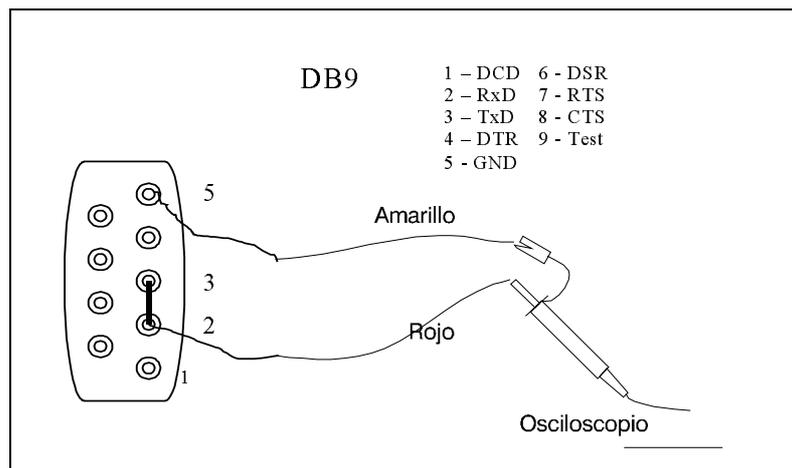
Escala eje X _____ s/div

2. Ejecuta el programa propuesto en el punto 4 del estudio previo y comprueba, utilizando un osciloscopio, la salida de la UART para los casos prácticos del apartado 2 del estudio anterior. Además dibuje en las plantillas siguientes, la salida mostrada por el osciloscopio

Escala eje Y _____ V/div

Departamento de Tecnología Electrónica. Universidad de Sevilla





El cable de la figura, realiza un bucle entre el pin de transmisión, TX, y el de recepción, RX. Esto permite a la UART recibir lo que envía.

5.- Cuestiones

1.- Si transmitimos a la velocidad binaria de 1200 bits/seg, ¿cuál es la velocidad efectiva en el caso de no haber errores (no hay retransmisiones)?, ¿cómo se puede mejorar la velocidad efectiva en la transmisión serie?

2.- A la salida de la UART (niveles TTL), se coloca el interfaz eléctrico. ¿qué niveles de señal has medido para los “0” y los “1” en el apartado 3 de la práctica? ¿qué consigues mejorar con el interfaz eléctrico?

3.- ¿Para que sirve el bit de paridad en la transmisión asíncrona?

4.- ¿Cómo se sincroniza el bit (temporización de reloj) y cómo se sincroniza el carácter?