Práctica Número 5:

"Comunicación Serie"

Objetivo

Comprender el principio de funcionamiento del puerto serie asíncrono y los modos de configuración.

Material y equipo para la práctica

1 Multímetro.
1 PC.
1 Tarjeta DEMOJM60.
1 Microcontrolador MC9S08JM60.
1 Adaptador de USB a Serial (TTL a FTDI)

Cuestionario Preliminar

1. ¿Qué es una comunicación serial?
2. ¿Qué es una comunicación asíncrona?
3. ¿Qué es un canal de comunicaciones?
4. ¿Qué es un baudio?
5. ¿Qué es un DCE y un DTE en comunicaciones?
6. ¿Cuáles son las velocidades estándar para transmisión serial?
7. ¿Qué es un bit de paridad?
8. ¿Qué es el código ASCII y para que sirve?
9. ¿Qué es el estándar RS-232?

Introducción

El puerto serie es uno de los módulos más habituales que integran los microcontroladores. Su función básica es la de establecer una comunicación serie asíncrona con otros sistemas digitales, tanto para intercambiar información, como para volcar el código en la memoria flash del microcontrolador.

Funcionamiento

En la comunicación serie los datos se envían y se reciben uno a uno de forma independiente, ver figura 3.38.

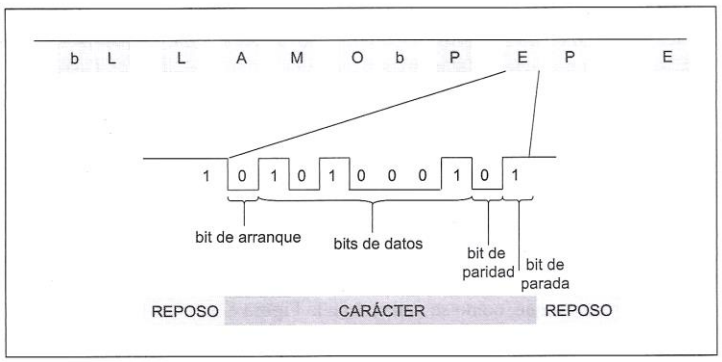


Figura 3.38 Ejemplo de una comunicación serial.

En reposo, el transmisor envía "unos" binarios continuamente. Cuando requiere transmitir un caracter envía un bit de arranque ("cero" binario) y, a continuación, los bits de datos correspondientes al caracter codificado en ASCII. El número de datos depende de la codificación utilizada (7 u 8 son los valores habituales). También se suele incluir un bit adicional de paridad para detectar errores de transmisión. Cuando el receptor recibe el bit de arranque activa un reloj que muestra los bits de datos. Cuando termina la transmisión, el transmisor envía 1 o 2 bits de parada (unos binarios) y vuelve a la condición de reposo. En el ejemplo de la figura 3.38 este procedimiento se repite caracter a caracter hasta que se envía el mensaje entero.

La figura 3.39 muestra un esquema simplificado de un sistema de comunicación asíncrona, hay un transmisor, un canal de comunicaciones y un receptor. En el transmisor, el dato a transmitir se carga en el buffer de transmisión. A continuación, se encapsula en una trama o paquete que añade los bits de control (arranque, paridad y parada) y se transfiere a un registro de desplazamiento, que va generando una secuencia de bits a la velocidad establecida por el reloj de transmisión. Los bits generados se envían a la interfaz de comunicaciones que suele hacer una conversión de niveles de tensión para mejorar las condiciones de transmisión.

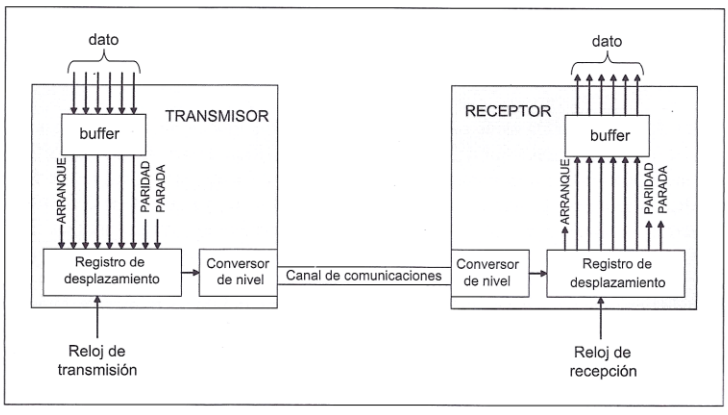


Figura 3.39 Esquema de un sistema de comunicación asíncrona.

En el receptor, se deshace la conversión de nivel y se almacena la trama en el registro de desplazamiento al ritmo de la frecuencia del reloj de recepción. Transmisor y receptor usan el mismo

formato de trama y la misma frecuencia nominal de reloj. Cuando el dato es completamente recibido se extrae del registro de desplazamiento y se transfiere al buffer de recepción.

Un aspecto muy importante es que, en todo este proceso, los bits adicionales de arranque y parada son fundamentales para establecer y mantener una frecuencia de tiempo común entre transmisor y receptor. Esto es necesario para realizar correctamente la comunicación puesto que los relojes de transmisión y recepción, aunque tienen la misma frecuencia nominal, son independientes entre sí. La sincronización de los relojes se lleva a cabo tomando como referencia el flanco de bajada que se produce cuando comienza el bit de arranque, como se muestra en la figura 3.40.

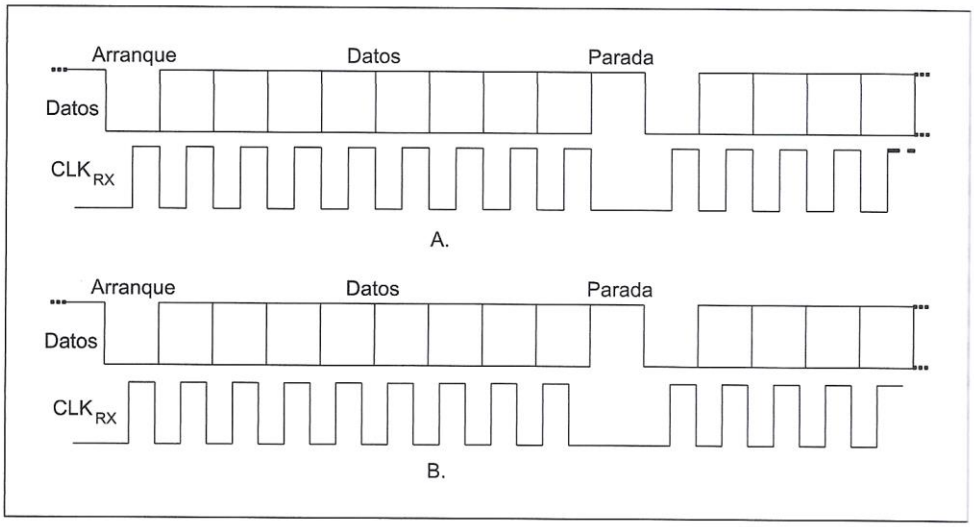


Figura 3.40 Sincronización en una comunicación asíncrona.

En una comunicación asíncrona, es posible que exista una cierta deriva entre los relojes del transmisor y receptor debido a que pueden sufrir cierta desviación con respecto a la frecuencia nominal de funcionamiento. Esto obliga a resincronizar los relojes de forma periódica, si se quiere garantizar un funcionamiento correcto, ya que, de lo contrario, se producirán errores. Un ejemplo en el que el reloj del receptor es ligeramente más rápido que el del transmisor se muestra en la figura 3.40, y como puede observarse, transcurrido un cierto tiempo desde la sincronización, la captura de los bits de datos se realizaría instantes de tiempo en los que su valor es inestable. La longitud de las tramas de datos está pues limitada si se quiere permitir cierta tolerancia en las frecuencias de los relojes de transmisión y recepción, siendo lo habitual utilizar 7 y 8 bits.

La inserción de bits de arranque y parada cada cierto número de bits garantiza la existencia de flancos de bajada independientemente de los datos transmitidos, de forma que el receptor puede mantenerse sincronizado con el trasmisor.

En algunos protocolos de comunicaciones serie asíncrona se utiliza un caracter especial denominado *break,* que habitualmente se utiliza para llamar la atención del receptor. La condición de break se produce cuando la línea permanece al valor del bit de arranque por un tiempo superior al necesario

para la transmisión de un caracter, lo cual es imposible en condiciones normales ya que hay al menos un bit con el valor del bit de parada por caracter transmitido.

En resumen, la necesidad de introducir bits adicionales en las comunicaciones asíncronas hace que éstas no sean adecuadas para altas velocidades, ya que son poco eficientes. Además, como ya se ha comentado, al ser los relojes de transmisión y recepción independientes, el tamaño de los caracteres no puede ser muy largo ya que conforme se incrementa el tamaño del caracter aumenta un posible desfase entre los relojes y, en consecuencia, la dificultad de recibir datos válidos.

La velocidad de transmisión hace uso de valores estándar, algunas velocidades típicas son 110, 300, 1200, 1400, 4800, 9600, 19200 bits por segundo. La unidad usada es bits por segundo, aunque a veces se puede encontrar otra muy utilizada en módems: el baudio. Se entiende por baudio el número de cambios de la señal por segundo; es importante saber que no siempre coincide el número de baudios con el de bits por segundo ya que, en ciertas modulaciones, se transfieren varios bits en un cambio de señal.

El puerto clásico, también llamado UART (*Universal Asynchronos Receiver/Transmitter, Transmisor-Receptor Asíncrono Universa*l), implementa un transmisor, un receptor, generadores de reloj, registros para configurar, entre otras cosas. El MC9S08JM60 está dotado de dos módulos hardware denominados SCI que facilitan la implementación de interfaces de comunicaciones serie asíncronas y ofrecen soporte para los protocolos y modos de operación más habitualmente implementados sobre este tipo de interfaces. Ambos módulos SCI son iguales y pueden ser utilizados de manera independiente, disponiendo cada uno de ellos de un conjunto separado de registros operacionales para su control y configuración.

La figura 3.41, cada módulo SCI está constituido por tres bloques bien diferenciados: el bloque transmisor, receptor y el generador de baudios. Tanto el transmisor como el receptor pueden operar independientemente uno del otro, pero comparten el mismo generador de baudios, por lo que la velocidad (o tasa binaria) será la misma en transmisión y recepción.

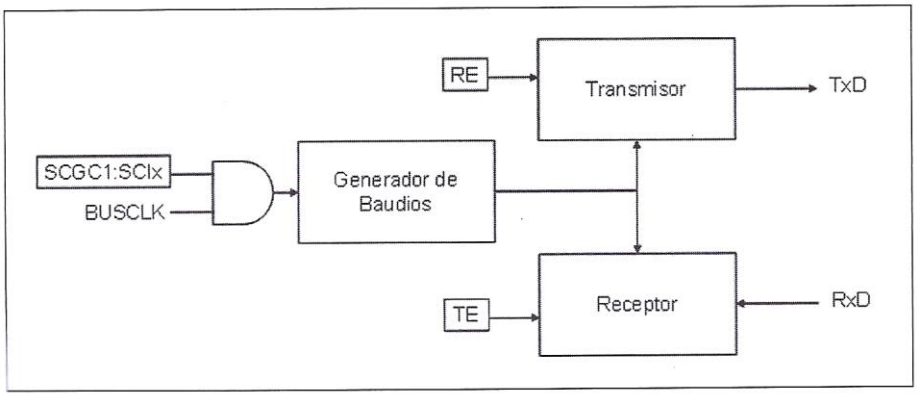


Figura 3.41 Diagrama de bloques del módulo de comunicación serie SCI.

Para el control de estos tres bloques, el microcontrolador dispone de 8 registros que permiten configurar su funcionamiento (registro de control), realizar operaciones (registro de datos) y

verificar los datos (registro de estado). Estos 8 registros están duplicados para el módulo SCI1 y para el SCI2.

Generador de baudios

El generador de baudios es el subsistema encargado de crear los relojes que gobiernan la transmisión y recepción de bits, permitiendo de esta manera configurar la tasa binaria a la que operarán el transmisor y el receptor. Como se muestra en el diagrama de bloques de la figura 3.41**,** básicamente consiste en un contador/divisor de 13 bits programable que divide la frecuencia del reloj BUSCLK por un valor configurable, de forma que pueda ajustarse la tasa de bits que utilizan transmisor y receptor, lo que implica que una modificación en la frecuencia del BUSCLK afecta a la velocidad con la que operan los módulos SCI.

La tasa binaria se ajusta mediante 13 bits situados en los registros SCIxBDH y SCIxBDL, viene dada por la siguiente expresión:

$$BaudRate = \frac{f_{BUSCLK}}{SBR * 16}$$

Dónde:

f_{BUSCLK} : es la frecuencia configurada para el reloj BUSCLK en el subsistema de generación de reloj.

SBR es el valor de 13 bits configurado para el contador/divisor programable, que puede estar comprendido entre 1 y 8191. El valor 0 es especial, ya que detiene el funcionamiento del contador/divisor y, por lo tanto, el funcionamiento de los bloques transmisor y receptor. SCIxSBDH y SCIxSBDL deben ser escritos en este orden, ya que el valor del contador/divisor se actualiza cuando se realiza la escritura de SCIxSBDL.

Puesto que el registro SBR tiene una resolución limitada de 13 bits, es evidente que no pueda conseguirse de manera exacta cualquier valor de tasa binaria deseada. No obstante, el mecanismo de sincronización mediante bits de arranque y parada, utilizados en las interfaces serie asíncronos, permite la existencia de cierta tolerancia en la tasa de los extremos transmisor y receptor sin que se produzcan errores en la comunicación. La tolerancia permitida en el ajuste de la tasa binaria es aproximadamente de un 3 o 4%, que debe incluir tanto las posibles desviaciones de la frecuencia del reloj respecto a la ideal, como error cometido por la resolución del SBR.

Bloque transmisor

El bloque es independiente de los otros dos bloques aunque comparte alguno de los parámetros de configuración con el bloque receptor y puede activarse y desactivarse mediante el bit TE del registro SCIxC2. Una vez habilitado, el transmisor permanece en estado de espera, manteniendo la línea en condición de parada hasta que se envíen datos al buffer de transmisión.

Los datos son enviados al buffer de transmisión, por el código que se ejecuta en el microcontrolador mediante escrituras en el registro SCIxD.

El elemento central es un registro de desplazamiento (registro de transmisión) en el que, en cada transmisión, se almacenan los bits que se van a transmitir por cada caracter a enviar:

- Bit de arranque
- Bits de datos
- Bit de paridad (si procede)
- Bit de parada

El número de bits que se envían por cada caracter depende de las diferentes opciones de configuración seleccionadas mediante los correspondientes registros de control. Cuando el registro de transmisión se encuentra disponible para enviar un nuevo caracter, el valor almacenado en el registro de datos (SCIxD) se transfiere al de desplazamiento, y comienza un ciclo de operación en el cual se va desplazando sincronizadamente con el reloj suministrado por el generador de baudios, de forma que en la línea TxD van apareciendo sucesivamente el bit de arranque, los bits de datos (primero el LSB), y finalmente el bit de parrada, con lógica positiva o negativa, en función del bit TXINV del registro SCIxC3.

Una vez finalizada la operación, se transfiere un nuevo valor desde el registro de datos SCIxD y se comienza nuevamente un ciclo. Cada vez que se transfiere un valor disponible en el SCIxD al registro de desplazamiento, se activa el bit de estado TDRE del registro SCIxS1 para indicar que se puede escribir un nuevo valor en el registro de datos. Si cuando finaliza la transmisión en curso no se ha escrito un nuevo valor en el registro de datos, entonces el transmisor entra en estado de espera y se activa la bandera TC del registro SCIxS1 para indicar que la transmisión ha finalizado. Ambos eventos registro de datos vacio y transmisión finalizada pueden producir la interrupción asociada al transmisor si ella está habilitada mediante los bits TIE y TCIE del registro SCIxC2 respectivamente.

El módulo transmisor del SCI permite configurar el modo de transmisión a 8 o 9 bits por caracter, lo que se controla mediante la bandera M del registro de control SCIxC1. En el caso de transmitir 9 bits, el bit más significativo tiene que ser escrito en el bit T8 del registro SC1xC3 y, posteriormente, los 8 bit menos significativos deben ser escritos en el registro de datos (SCIxD). El transmisor del SCI incluye la posibilidad de generar automáticamente un bit de paridad para mejorar la fidelidad de las comunicaciones asíncronas, permitiendo detectar errores en la transmisión. Para habilitar la paridad es necesario activar el bit PE del registro SCIxC1. El tipo de paridad utilizado (par o impar) puede ajustarse mediante el bit PT del mismo registro. La configuración seleccionada mediante ambos bit afectará tanto al comportamiento del transmisor como al del receptor. En el caso de habilitar la paridad, ésta se generará automáticamente en el bit más significativo, por lo tanto, perderá su capacidad de llevar información, de forma que según se haya seleccionado el modo de transmisión con 8 ó 9 bits, se tendrá 7 u 8 bits de datos respectivamente.

Cuando el transmisor es deshabilitado poniendo a 0 el bit TE del registro SCIxC2, no se desactiva inmediatamente, sino que primero finalizará la transmisión del caracter en curso para evitar la transmisión de un caracter erróneo. Una vez finalizada, la línea de salida pasa al estado de espera.

El transmisor del SCI también incluye facilidades para generar y detectar el caracter especial *break*, que fuerza al nivel de parada en la línea durante un periodo de tiempo superior al de un caracter. La generación de la condición *break* por el transmisor es disparada poniendo a 1 el bit SBK del registro SCIxC2. Para enviar un *break*, el programa normalmente verificara que el registro de datos

se encuentre vacío mediante el bit TDRE y, tras la activación de SBK, un caracter de *break* quedará en la cola para ser enviado en cuando finalice la transmisión en curso.

Si en ese momento SBK permanece aún activo, afecta a la longitud del caracter de *break* que será de 10 y 11 bits respectivamente. Adicionalmente se pueden generar caracteres de *break* de 13 o 14 bits si se activa el bit BRK13 del registro SCIxS2.

Bloque receptor

Puede activarse o desactivarse independientemente de los otros dos bloques mediante el bit RE del registro SCIxC2. Al igual que el transmisor, puede operar con caracteres de 8 o 9 bits. Los valores recibidos por la entrada RxD pueden ser enviados mediante el bit RXINV del registro SCIxS2.

El bloque receptor detecta el flanco provocado en línea por la condición de arranque y lee los bits sucesivos a la velocidad marcada por el generador de baudios, mediante el registro de desplazamiento. Una vez capturado el número adecuado de bits en el registro de desplazamiento, el caracter recibido es transferido al registro datos SCIxD, y el bit RDRF del registro de estado SCIxS1 es puesto a uno para indicar que existe un dato válido en SCIxD que puede ser leído por el programa en ejecución. Si el registro SCIxD ya había un dato disponible de una recepción anterior que no ha sido leído todavía RDRF estaría ya activo antes de recibir el caracter, entonces el nuevo dato recibido se pierde y se pone a uno el bit de estado OR del registro SCIxS1, que indica que se ha producido una pérdida de un caracter. El bit RDRF se desactiva automáticamente cuando se leen SCIxS1 y SCIxD desde el programa en ejecución, lo cual implica que se ha leído el caracter recibido. La activación del bit RDRF puede dar lugar a una interrupción del receptor si se habilita mediante el correspondiente bit RIE en el registro SCIxC2.

El receptor incluye los mecanismos necesarios para conseguir una buena sincronización con los bits que están siendo recibidos, lo cual es importante para garantizar la integridad de la comunicación. Si además de la condición de arranque se detectan algunos flancos de bajada adicionales, el receptor los aprovecha para mantener la sincronización. También se incluyen mecanismos para la detención de flancos espurios cercanos al instante en el que se capturan los bits de entrada, indicando su presencia mediante la activación del bit NF del registro SCIxS1. De la misma forma, se verifica automáticamente la correcta recepción del bit de parada, indicando un error de trama mediante el bit FE de SCIxS1 en caso de que no se reciba correctamente el bit, que indicará un problema con el sincronismo entre el transmisor y el receptor que están llevando a cabo la comunicación. Cuando se produce un error de trama, el receptor es deshabilitado hasta que se borre el bit FE por programa.

El receptor del SCI incluye el hardware necesario para la comprobación del bit de paridad, lo que permite detectar errores en la transmisión de los bits. Para habilitar la verificación de paridad es necesario activar el bit PE del registro SCIxC1. El tipo de paridad utilizando par o impar puede ajustarse mediante el bit PT del mismo registro. Como se mencionó, estos bits afectan tanto a la generación automática del bit de paridad en el transmisor como para sondear el mismo en el bloque receptor. En caso en que se detecte un error de paridad, el receptor activa la bandera PF del registro SCIxS1, pudiendo dar lugar también a una interrupción por hardware.

- ***Registros de tasa de baudios (SCIxBDH:SCIxBDL)***

El registro de configuración de la tasa de baudios del SCI. Se recomienda al usuario escribir primero sobre el registro SCIxBDH y luego sobre el registro SCIxBDH. Para el caso de programación en C, es posible escribir sobre el registro SCIxBD, el cual agrupa los registros anteriores.

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura Escritura	LBKDIE	RXEDGIE	0	SBR[12:8]				
Reset	0	0	0	0	0	0	0	0

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura Escritura	SBR[7:0]							
Reset	0	0	0	0	0	0	0	0

LBKDIE

Bit para habilitar un evento de interrupción debido a la recepción de un caracter de pausa (*break*).
0: No habilita interrupción por evento de pausa
1: Habilita interrupción por evento de pausa

RXEDGIE

Bit para habilitar que se genere un evento de interrupción debido a un flanco presente en el pin de recepción.
0: No habilita interrupción por flanco en pin de Rx
1: Habilita interrupción por flanco en pin de Rx

SBR[12:0]

Bits para la programación del divisor de la tasa de baudios.
Cuando el valor de estos bits es cero, el generador de la tasa de baudios se detiene con el fin de ahorrar energía. La ecuación para el cálculo de la tasa de baudios es:

$$Tasa\ de\ Baudios = \frac{Reloj\ de\ BUS}{(16 * SBR)}$$

Por ejemplo, si se necesita una tasa de baudios de 9600 y se está trabajando con un reloj de BUS interno a 8MHZ, el SBR será de:

$$SBR = \frac{8\ MHz}{(16 * 9600Hz)}$$

$$SBR = 52.08$$

El valor anterior se puede aproximar a 52, sin ningún deterioro del sincronismo de la comunicación.

- ***Registro de control 1 del SCI (SCIxC1)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura Escritura	LOOPS	SCISWAI	RSRC	M	WAKE	ILT	PE	PT
Reset	0	0	0	0	0	0	0	0

LOOPS
Bit para seleccionar entre operación normal y modo de realimentación (LOOP). Cuando este bit vale "1", la salida del transmisor es conectada internamente a la entrada del receptor.

0: Operación normal del SCI. Los pins RxD y TxD son usados de manera independiente
1: Habilita modo LOOP o conexión por un único alambre (*Single Wire Mode*)**.** El pin RxD se desconecta del SCI

SCISWAI
Bit para detener el SCI en modo WAIT.
0: El SCI continúa trabajando estando la máquina en modo WAIT, de tal forma que un evento de interrupción del SCI podría despertar la máquina.
1: El módulo SCI está detenido en modo WAIT.

RSRC
La operación con este bit tiene sentido si el bit LOOP está en "1". En este modo el transmisor está conectado internamente al bloque de entrada del receptor, entonces el bit RSRC decide cuando ésta unión se conecta al pin de salida (TxD) del SCI.

0: Modo normal de LOOP, la entrada RxD está desconectada del sistema y la salida del transmisor es conectada internamente a la entrada del bloque receptor.
1: Pone el SCI en modo de un único alambre (*Single Wire Mode*), entonces el pin de TxD es conectado a la salida del transmisor y a la entrada del receptor, internamente.

M
Bit para seleccionar dato de 8 o 9 bits.
0: Modo estándar: 1 bit de start + 8 bits de dato + 1 bit de stop.
1: Modo a 9 bits: 1 bit de start + 8 bits de dato + bit 9 + 1 bit de stop.

WAKE
Bit para seleccionar el método de despertar (*Wakeup)* del SCI
0: Modo de despertar por *Idle Line*
1: Modo de despertar por *Address Mark*

ILT
Bit para seleccionar el tipo de línea en modo IDLE. Cuando este bit es puesto a "1", se asegura de que el bit de stop y el MSB del dato no cuenten dentro de los 10 y 11 bits necesarios como nivel IDLE para la lógica de detección

0: El contador de bits, del caracter para la condición de IDLE, comienza después del bit de start
1: El contador de bits, del caracter para la condición de IDLE, comienza después del bit de stop

PE
Bit para habilitar la generación y el chequeo de paridad en la trama de comunicación.
0: No se trabaja con paridad
1: Habilita el trabajo con paridad

PT
Bit para seleccionar el tipo de paridad a generarse o muestrar.
0: Paridad par (Even)
1: Paridad impar (Odd)

- ***Registro de control 2 del SCI (SCIxC2)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura Escritura	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
Reset	0	0	0	0	0	0	0	0

TIE
Bit para habilitar un posible evento de interrupción por transmisión, para cuando el *buffer* de transmisión ha bajado el dato al *shift register*.
0: Deshabilita interrupción por evento de TDRE = 1
1: Habilita interrupción por evento de TDRE = 1

TCIE
Bit para habilitar un posible evento de interrupción por transmisión, para cuando el último bit ha salido del *shift register*.
0: Deshabilita interrupción por evento de TC = 1
1: Habilita interrupción por evento de TC = 1

RIE
Bit para habilitar un posible evento de interrupción por recepción.
0: Deshabilita interrupción por evento de RDRF = 1
1: Habilita interrupción por evento de RDRF = 1

ILIE
Bit para habilitar un posible evento de interrupción por línea en modo IDLE.
0: Deshabilita interrupción por evento de línea en estado de IDLE
1: Habilita interrupción por evento de línea en estado de IDLE

TE
Bit para habilitar la operación del transmisor
0: Deshabilita la operación del transmisor
1: Habilita la operación del transmisor

RE
Bit para habilitar la operación del receptor
0: Deshabilita la operación del receptor
1: Habilita la operación del receptor

RWU
Bit para colocar el receptor del SCI en modo de standby (atento), esperando a que se dé un evento de hardware o se presente un modo de despertar (*Wakeup*), sea por *idle line* (WAKE = 0) o por *address mark* (WAKE = 1)
0: SCI en operación normal
1: SCI en modo de *standby,* esperando un estado de WAKE

SBK
Bit para permitir enviar un caracter *break*. Un caracter *break* consiste en el apilamiento de "0" en toda la trama de comunicación. La condición no desaparece hasta tanto no se haga SBK = 0
0: SCI en operación normal
1: SCI enviando caracteres *break*

- ***Registro de estado 1 del SCI (SCIxS1)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
Escritura								
Reset	0	0	0	0	0	0	0	0

TDRE
Bandera que indica un dato del *buffer* de transmisión ha pasado al *shift register*. Este evento revela que se puede depositar otro dato en el *buffer* de transmisión. Para borrar la bandera TDRE es necesario leer el registro SCIxS1 y luego escribir un nuevo dato en el registro de datos del SCI (SCIxD).
0: El *buffer* de transmisión está lleno
1: El *buffer* de transmisión está vacío

TC
Bandera que indica, el último bit de una trama ha salido por el *shift register*. Este evento indica que se puede depositar otro dato en el *buffer* de transmisión. Para borrar la bandera TC es necesario leer el registro SCIxS1 y luego ejecutar una de las siguientes acciones:

Escribir un nuevo dato en el registro de datos del SCI (SCIxD).
Escribir un preámbulo pasando el bit TE de "1" a "0".
Enviar un caracter de *break* escribiendo un "1" en el bit SBK.
0: Transmisor enviando un dato
1: Transmisor en modo IDLE

RDRF
Bandera que indica, el *buffer* de recepción está lleno. El evento revela cuándo un dato recibido en el *shift register* ha pasado al registro de datos de SCI (SCIxD). Para poner el bit a cero, el estado del bit RDRF es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.
0: El registro de datos del SCI está vacío
1: El registro de datos del SCI está lleno

IDLE
Bandera se pone a "1" cuando el receptor ha recibido un caracter de sólo "1's". Para poner le bit a cero la bandera de IDLE es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.

0: No se ha detectado un caracter IDLE
1: Se ha detectado un caracter IDLE

OR
Bandera de sobre carrera del receptor (*Receiver Overrun*). Esta bandera indica cuándo se va a escribir un nuevo dato entrante sobre el registro de datos de SCI (SCIxD) y aún no se ha leído el dato anterior, en cuyo caso el dato nuevo se pierde. Para para poner el bit a cero el bit OR es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.
0: No se ha detectado una sobre carrera en el receptor
1: Se ha detectado una sobre carrera en el receptor y se ha perdido el último dato

NF
Todo bit de start es muestreado siete veces para los demás bits. Si alguna de esas muestras es errónea, la bandera NF se pone a "1" al igual que el bit RDRF. Para poner el bit a cero el bit NF es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.
0: No se ha detectado ruido en el canal
1: Se ha detectado ruido en el canal

FE
Bandera para detección de error en una trama aunque no es la única condición para descartar una trama corrupta. Una trama se invalida cuando se detecta un "0" en el tiempo del bit de *stop*. Para poner el bit a cero el bit FE es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.
0: No se ha detectado una trama errónea
1: Se ha detectado una trama errónea

PF
Bandera que indica cuándo el bit de paridad no coincide con el valor de la paridad pactada. Para poner el bit a cero el bit PF es necesario leer el registro SCIxS1 y luego leer el registro de datos SCIxD.
0: No se ha detectado paridad inválida
1: Se ha detectado paridad inválida

- ***Registro de estado 2 del SCI (SCIxS2)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura	LBKDIF	RXEDGIF	0	RXINV	RWUID	BRK_{13}	LBKDE	RAF
Escritura								
Reset	0	0	0	0	0	0	0	0

LBKDIF
Bandera que indica cuándo se ha detectado un caracter de *break* en la línea. Esta bandera se puede regresar a su valor original escribiendo un “1” en ella.
0: No se ha detectado un caracter de *break* en la línea
1: Se ha detectado un caracter de *break* en la línea

RXEDGIF
Bit para indicar cuándo ha ocurrido un evento de interrupción por un flanco recibido en el pin de RxD. Puede ser de bajada (RXINV = 0) o de subida (RXINV = 1). Esta bandera se puede regresar a su valor original escribiendo un “1” en ella.
0: No se ha detectado un flanco en el pin RxD
1: Se ha detectado un flanco en el pin RxD

RXINV
Bit para invertir la polaridad de la señal eléctrica aplicada al pin RxD.
0: No se ha invertido la polaridad en el pin RxD
1: Se ha invertido la polaridad en el pin RxD

RXUID
Bit para indicar la detección de un estado de despertar por modo *Idle Wakeup*. Indica cuando un caracter IDLE ha puesto la bandera en “1”.
0: Durante el estado de atento del receptor (RWU = “1”), el bit IDLE no ha sido puesto a “1” en la detección de un caracter IDLE
1: Durante el estado de atento del receptor (RWU = “1”), el bit IDLE ha sido puesto a “1” en la detección de un caracter IDLE

BRK_{13}
Bit para seleccionar la longitud de un caracter de *break*.
0: El caracter de *break* será de 10 bits (11 si M =1)
1: El caracter de *break* será de 13 bits (14 si M =1)

LBKDE
Bit para habilitar un caracter *break* en la línea. Mientras este bit sea “1”, el sistema previene que el bit FE (*Famming Error*) y RDRF sean puestos a “1”.
0: Un caracter de *break* de 10 bits es detectado (11 si M =1)
1: Un caracter de *break* de 13 bits es detectado (14 si M =1)

RAF
Bandera para indicar que el receptor ha detectado un bit de *start* válido y es borrada automáticamente cuando el receptor detecta la línea en modo IDLE. Esta bandera puede ser usada para verificar cuándo un caracter está comenzando a ser recibido, antes de que la CPU entre en un estado de STOP.
0: El receptor en modo IDLE está esperando un bit de *start*
1: El receptor del SCI está activo y el pin RxD no está en IDLE

- ***Registro de control 3 del SCI (SCIxC3)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura	R_8	T_8	TXDIR	TXINV	ORIE	NEIE	FEIE	PEIE
Escritura								
Reset	0	0	0	0	0	0	0	0

R8
Noveno bit del receptor, para M = 1 y conforma el MSB del dato recibido. Cuando se está trabajando en modo de 9 bits, se recomienda leer R8 antes de leer el SCIxD.

T8
Noveno bit a transmitir, para M = 1 y conforma el MSB del dato a transmitir. Cuando se está trabajando en modo de 9 bits, escribir primero el dato a transmitir en el registro SCIxD y luego definir el bit T8.

TXDIR
Bit para definir la dirección del pin de TxD, en modo de un solo hilo *Simple Wire half duplex* (LOOPS = RSRC = 1).
0: El pin TxD es una entrada en el modo de un solo hilo
1: El pin TxD es una salida en el modo de un solo hilo

TXINV
Bit para invertir el estado eléctrico de los bits en el pin TxD.
0: Los bits transmitidos no son invertidos
1: Los bits transmitidos son invertidos

ORIE
Bit para habilitar un evento de interrupción por *Receiver Overrun*.
0: Deshabilita interrupción por *Receiver Overrun*
1: Habilita interrupción por *Receiver Overrun*

NEIE
Bit para habilitar un evento de interrupción por *Noise Error*.
0: Deshabilita interrupción por *Noise Error*
1: Habilita interrupción por *Noise Error*

FEIE

Bit para habilitar un evento de interrupción por *Framming Error*.
0: Deshabilita interrupción por *Framming Error*
1: Habilita interrupción por *Framming Error*

PEIE

Bit para habilitar un evento de interrupción por *Parity Error*.
0: Deshabilita interrupción por *Parity Error*
1: Habilita interrupción por *Parity Error*

- ***Registro de datos del SCI (SCIxD)***

	Bit 7	6	5	4	3	2	1	Bit 0
Lectura	R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0
Escritura	T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0
Reset	0	0	0	0	0	0	0	0

Rn: Dato Recepción
Tn: Dato Transmisión

Desarrollo

El circuito de la figura 3.42 detalla la aplicación a implementar, como ejercicio del manejo de la comunicación serial con una PC.

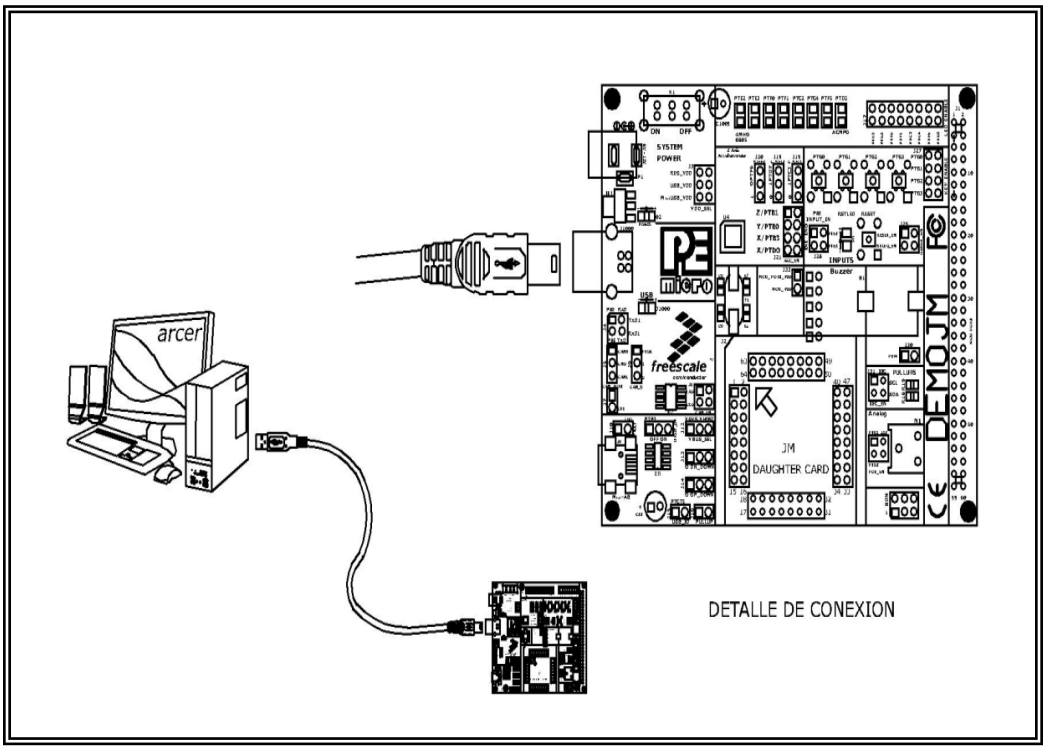
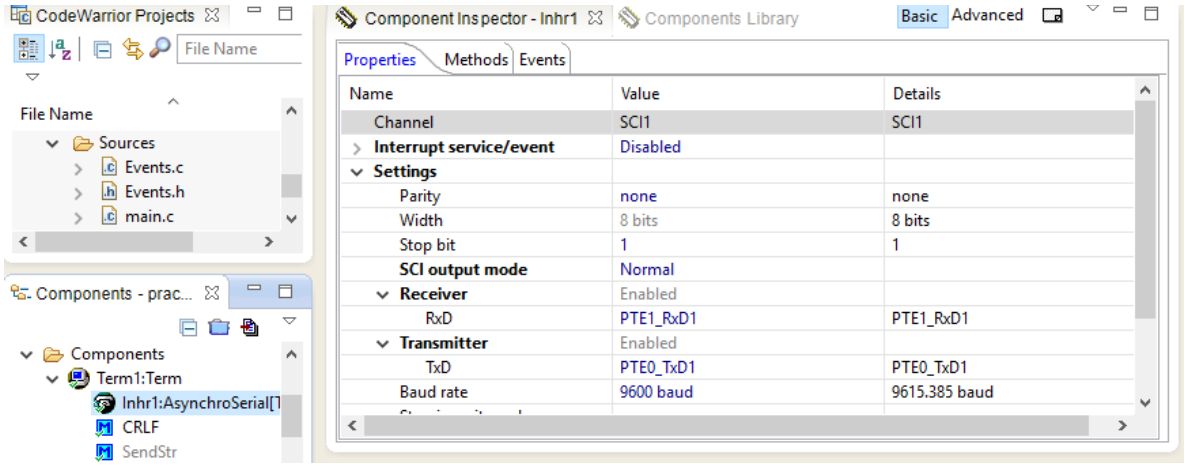


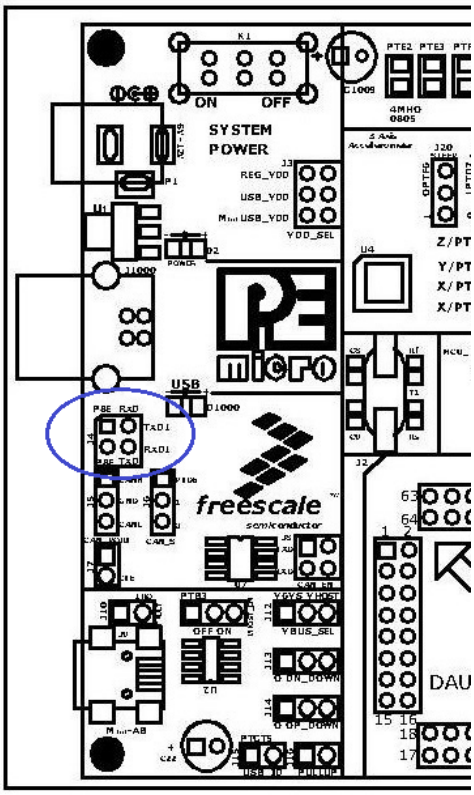
Figura 3.42 Circuito para transmitir vía serial a la computadora.

1. Crear un nuevo proyecto en CodeWarrior (Eclipse) seleccionando el dispositivo MC9S08JM60 para ser conectado con P&E USB Multilink y seleccionar **Lenguaje C**, y seleccionar **Processor Expert**.

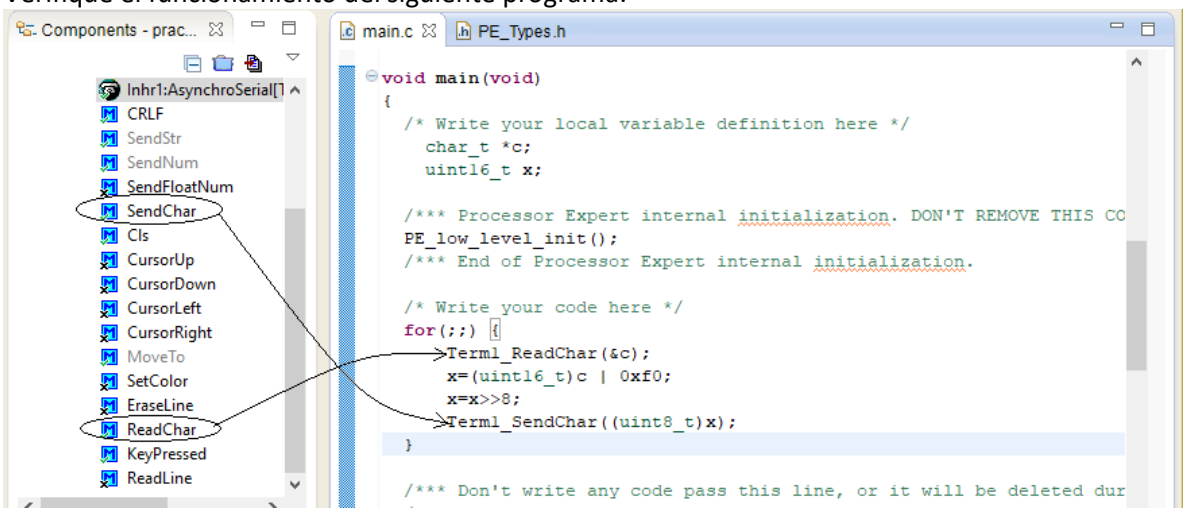
Agregar el componente ***Term***, que se encuentra en ***CPU External Devices***, ***Display***. Y configurar el **SCI1 a 9600** baud.



NOTA: Es importante quitar los Jumper TXD1 y RxD1



2. Verifique el funcionamiento del siguiente programa:



3. Realizar la comunicación serial entre el microcontrolador y la PC, atra vez de la herramienta *P&E Embedded Multilink Toolkit* (Solo para equipos con windows de 32 bits) o descargue **Tera Term**: http://ttssh2.osdn.jp/

4. Modificar el programa anterior, agregando **PTC4** para:
   a) Que se encienda y se apague 2 veces, cuando reciba un numero.
   b) Que se encienda y se apague 4 veces, cuando reciba una letra.