

SISTEMA DE GENERACIÓN DE RUIDO BASADO EN MICROCONTROLADOR

ALBERTO MOLINA¹, ARIEL GOMEZ¹ y ANTONIO FRAILE²

¹Departamento de Tecnología Electrónica. Facultad de Informática. Universidad de Sevilla. 41012-SEVILLA. España

²Departamento de Informática. SAINCO. Sevilla

En este artículo se presenta un sistema que posibilita la programación de protocolos punto a punto y multipunto en comunicaciones RS-232 entre ordenadores. Asimismo, la capacidad de generación de ruido en los diferentes canales de comunicación, hace de este sistema, un elemento útil para la comprobación de la fiabilidad de los protocolos implementados.

1. Introducción y objetivos

Los niveles más básicos en el conocimiento de las comunicaciones electrónicas, incluyen el estudio de dispositivos electrónicos y técnicas software para la detección y corrección de errores en las comunicaciones de datos. La elaboración de experiencias reales que apoyen los conceptos teóricos relacionados con CRC, BCC, etc. bajo sistemas (ordenadores personales) en comunicaciones RS-232 punto a punto, requieren de sistemas externos que generen ruido de forma sistemática o aleatoria entre los PC's.

2. Descripción general

El sistema generador de ruido (desde ahora "diablillo"), puede trabajar con un máximo de 4 PC's simultáneamente (aunque puede ampliarse a 5) (ver figura 1). Existen tres configuraciones básicas: modo **multipunto**, modo **punto a punto** y modo **bucle local**.

En el primero de ellos, el diablillo está configurado como concentrador, permitiendo una conexión en bus de los cuatro PC's a través de la RS-232. Esto permite que todos los PC's sean capaces de recibir lo enviado por cualquiera de ellos. El diablillo permite la detección de colisiones (2 o más PC's transmitiendo simultáneamente) y señalización de colisión a cada uno de los PC's (activación de /DSR y envío de BREAK durante 10ms)

El modo multipunto abre la posibilidad de realización de prácticas algo más avanzadas: protocolos de acceso al medio, token-bus, etc. Usando como sustrato, siempre, la RS-232.

En modo **punto a punto**, se establecen conexiones dedicadas fijas entre los cuatro ordenadores de dos en dos y en modo bucle **local** cada ordenador recibe lo que el mismo envía.

En cualquiera de los modos anteriores, existe la posibilidad de activar o desactivar la generación de ruido. El ruido generado consiste en ráfagas de bits de longitud variable (de 1 a 8 bits), que se interfieren con los valores lógicos de los bits que se transmiten en cualquiera de las configuraciones anteriores. Asimismo, la frecuencia de aparición de las ráfagas puede variarse desde 10ms a 8seg aproximadamente, con el objeto de disponer de diferentes tasas de ruido en el canal.

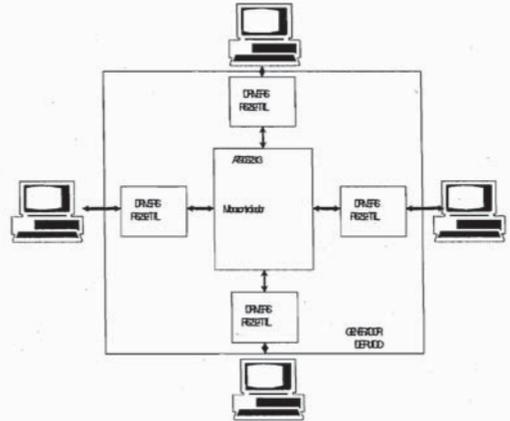


Figura 1: Estructura general de conexión

En la figura 2 se esquematiza el procedimiento de generación de ruido. El algoritmo implantado requiere de dos variables de memoria (*Cont* y *Gen*) en el programa del microcontrolador. La primera de ellas genera el patrón de bits de la ráfaga.

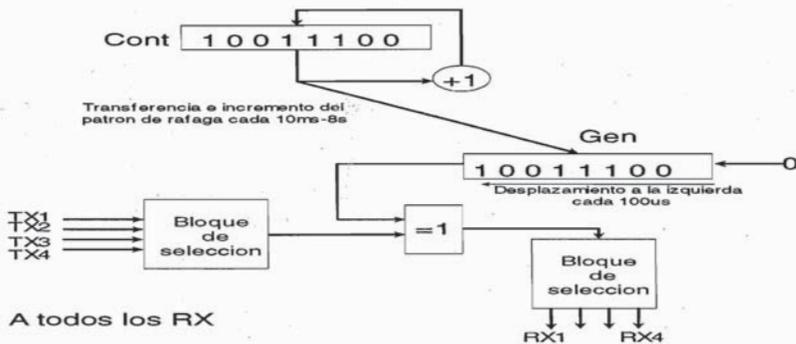


Figura 2: Procedimiento de generación de ráfagas de ruido

Su contenido se incrementa con una frecuencia igual a la de generación de tasa de ruido. Cada vez que se incrementa *Cont*, su contenido se transfiere a *Gen*, que produce la generación real del ruido. El contenido de *Gen* es desplazado, ahora, cada 108µs, (T_{bit} para velocidades de 9600bps). Ahora se calcula la XOR entre el bit7 de *Gen* con el bit de la entrada Tx_i elegida

por el bloque de selección. El resultado de esta operación se sitúa en la salida Rx, correspondiente. Este proceso se repite de forma continúa.

Por lo explicada hasta aquí, el diablillo dispone de diversos parámetros de configuración (modo de conexión, generación de ruido y tasa de error). Estos parámetros se almacenan en una EEPROM interna que elimina la necesidad de configurarlo cada vez que este entre en funcionamiento. La modificación de los mismos se establece a través de uno de los PC conectados al "diablillo" que denominaremos "PC amo". El PC amo dispone de un cableado especial, ya que además de las líneas TX, RX, y GND, común a todos los interfaces, dispone de la línea /DTR. La activación de esta línea provoca que el diablillo entre en estado de configuración, en el que se establece un diálogo entre el PC amo y el "diablillo", y al final del cual, este último, recibe los nuevos parámetros de configuración.

3. Descripción técnica

La arquitectura hardware del "diablillo" se compone de los siguientes elementos:

El microcontrolador AT90S2313 (figura 3). En la tabla 1 se resume las conexiones del microcontrolador con los diferentes interfaces.

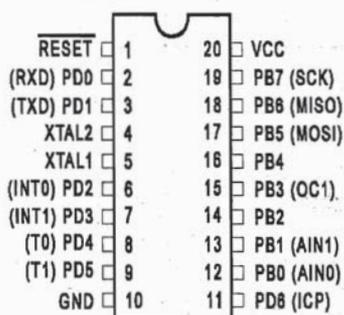


Figura 3: Microcontrolador AT90S2313

PIN	Configurado como	Conectado a (*)
PD0	Entrada	TX1 (PC amo)
PD1	Salida	RX1 (PC amo)
PD2	Entrada	/DTR (PC amo)
PB7	Salida	RX4
PB6	Salida	RX3
PB5	Salida	RX2
PB3	Entrada	TX4
PB2	Entrada	TX3
PB1	Entrada	TX2
PD3	Salida	/DSR (todos los PC's)

(*)A través de los correspondientes drivers

Tabla 1 : Conexionado del microcontrolador con los PC's

La operación del microcontrolador es simple. Este se encuentra muestreando las salidas de los cuatro ordenadores. En función de los valores lógicos de estas, y el modo de configuración, se activan las entradas de estos. Si el ruido está activo, para cada intervalo de tiempo programado, se genera una ráfaga de bits que se interfieren, mediante operación EXOR con los valores lógicos de las salidas de los PC's.

4. Realizaciones experimentales

Podemos dividir en dos grupos, los tipos de prácticas que pueden implementarse con el "diablillo": prácticas de nivel físico y prácticas de nivel de enlace aunque puede ampliarse sustancialmente dicha clasificación.

Nivel físico

P1: programación de la UART. ("Diablillo configurado en modo bucle") Los alumnos pueden realizar rutinas de nivel físico para la programación de la UART y comprobar su funcionamiento con un **ÚNICO ordenador**.

P2: detección de errores en la UART. ("Diablillo" configurado en modo bucle y simple con generación de ruido o sin generación). Los alumnos prueban las comunicaciones entre dos ordenadores a diferentes configuraciones, con o sin ruido y depuran sus propios programas de detección de errores en modo bucle local con ruido)

Nivel de enlace

P1: transmisión de paquetes con CRC. ("Diablillo" configurado en modo simple con generación de ruido).

P2: protocolos de acceso al medio: contienda, sondeo, token bus... ("Diablillo" configurado en modo múltiple)

5. Conclusiones

El "diablillo" constituye un sistema de muy bajo coste que permite la realización de una gran gama de experiencias prácticas.

Referencias

- [1] ATMEL corporation. *Microcontrolador AT90S2313. Hardware and Software.*
- [2] J. Campbell. *Comunicaciones serie.* Anaya