

3º I.T.I. de Sistemas. PERIFÉRICOS. Práctica 3

Comunicaciones serie asíncronas mediante interrupciones

En esta práctica se pide escribir un programa que descifre el flujo de datos que llegue al puerto serie del microcontrolador y lo devuelva descifrado. La rutina de descifrado está disponible como un fichero objeto en “*ejemplos/cripto/decrypt.o*” y se puede enlazar con el código de la práctica. Este fichero objeto contiene la función:

```
void decrypt(unsigned char *origen, unsigned char *destino);
```

Esta función opera en un bloque de datos de 16 caracteres, cuya dirección se pasa como “origen”, y deja los resultados en un bloque de 16 bytes cuya dirección se pasa como “destino”. La descifrado se realiza por lo tanto en bloques de 16 caracteres lo que implica recibir 16 caracteres desde la UART, llamar a la función “decrypt” y transmitir los 16 caracteres resultantes por la UART. El tiempo de ejecución estimado para la función “decrypt” es de entre 5 y 10 ms. El formato de los datos es 9600 bits/s, 8 bits de datos, sin paridad y 1 bit de STOP.

En el directorio “*ejemplos/cripto/*” se tienen varios ejemplos de ficheros encriptados para probar el programa objeto de la práctica. Estos ficheros se pueden transmitir mediante el programa terminal al microcontrolador pulsando “<ctrl>-f” y especificando el nombre del fichero.

NOTAS:

- El uso de rutinas de interrupción para la recepción y transmisión de datos por el puerto serie es una necesidad. De lo contrario no podrían decodificarse de forma fiable ficheros de más de 16 bytes pues podríamos tener errores de overrun si se reciben datos mientras se está decodificando o transmitiendo.
- Las rutinas de interrupción de la UART deben ser del tipo “signal”. Esto es: que no se ejecute la instrucción “STI” durante la llamada a la función.
- El algoritmo de encriptación es muy simple. No utilizarlo para aplicaciones serias ya que puede ser resuelto sin mucho esfuerzo por gente experta.