



## **TAREA #2**

Para cada uno de los siguientes problemas debe utilizar el simulador para hacer todas las pruebas que le permitan validar que los programas cumplen con las funcionalidades requeridas. La calificación de la tarea incluirá, entre otras cosas, dicha validación y la persona estudiante debe verificar que su solución las cumple. En todos los casos se evaluará el cumplimiento de las funcionalidades y la eficiencia de código de los programas.

### **Problema #1.**

Diseñe y codifique en ensamblador del S12 un programa llamado CONVERSIONES que incluye dos rutinas (no subrutinas) llamadas BIN-BCD y BCD-BIN.

- a. Rutina BIN-BCD: Esta rutina realiza la conversión de un número Binario de 12 bits a BCD utilizando el algoritmo XS3. El número binario está en el acumulador D. La rutina coloca el resultado en la variable NUM\_BCD ubicada en la memoria a partir de la posición \$1010.
- b. Rutina BCD-BIN: Esta rutina realiza la conversión de un número BCD a Binario, utilizando el método de multiplicación de décadas y suma. El número en BCD es menor o igual a 9999 y está ubicado en el acumulador D. La subrutina guarda el resultado en las posiciones de memoria NUM\_BIN ubicadas a partir de la dirección \$1020.

El programa principal debe crear los valores a convertir como constantes. El valor binario estará en la constante BIN ubicada en las posiciones \$1000-\$1001, en tanto el valor BCD a convertir estará en la variable BCD ubicada en las posiciones \$1002-\$1003.

En el programa CONVERSIONES primero se copia el valor del BIN en el acumulador D y se implementa el código de la rutina BIN\_BCD, luego el programa copia el valor de BCD en el acumulador D y se implementa el código de la rutina BCD-BIN. Realice las pruebas pertinentes a su programa para garantizar que el programa satisface los requerimientos establecidos. Ubique el programa a partir de la posición \$2000.

### **Problema #2.**

Se tienen dos tablas, la primera de ellas se encuentra en la posición DATOS y contiene números con signo en el intervalo  $[-127, +127]$ . La segunda tabla contiene máscaras sin signo menores que 254 y se encuentra en la dirección MASCARAS. Ambos arreglos son de tamaño variable menor de 1,000. El último valor en DATOS será siempre \$80 y el último valor en MASCARAS será \$FE, siendo estos valores los indicadores de fin de



tabla. Se debe realizar la XOR de los números con las máscaras, el último número con la primera máscara, el penúltimo número con la segunda máscara y así sucesivamente hasta procesar todos los datos del arreglo que se termine primero. Se deben utilizar los índices J y K para barrer las tablas y además el índice K también se utiliza para generar un arreglo de resultados que se encuentra a partir de la posición NEGAT. En NEGAT se colocarán los resultados de las XORs que sean números negativos.

- Diseñe este programa incluyendo las estructuras de datos utilizadas.
- Codifique el programa en lenguaje ensamblador de S12 considerando que la dirección DATOS es \$1050, la de MASCARAS es \$1150, la de NEGAT es \$1300. El programa debe ubicarse a partir de la posición \$2000. Declare cualquier otra estructura de datos que utilice. Ensamble el programa y genere el archivo .S19. Pruebe el programa para una tabla de su elección. El código debe crear las tablas DATOS y MASCARAS en tiempo de ensamblado.

### Problema #3:

Considere que se tiene un arreglo de números de 1 byte, con signo, de tamaño variable L, llamado DATOS, donde L es un número menor que 255. Se deben encontrar todos los números en el arreglo que son divisibles por 4 y copiarlos en otro arreglo llamado DIV4. Adicionalmente el programa debe calcular la cantidad de números divisibles por 4 encontrados y almacenarlo en la variable CANT4. En el diseño el registro índice X apunta a Datos y el índice Y apunta a DIV4, estos índices NO se deben modificar durante la ejecución del programa. Nota: Observe que para determinar aquellos números negativos divisibles por 4 hay que conocer primero su magnitud.

- Haga el diseño del programa incluyendo las respectivas estructura de datos a utilizar.
- Codifique su diseño en lenguaje ensamblador del S12. El programa debe iniciar en \$2000. Las direcciones para DATOS y DIV4 son \$1100, \$1200, respectivamente. Además las variables L y CANT4 se deben ubicar en las posiciones \$1000 y \$1001 respectivamente. Ensamble el programa y compruebe que satisface los requerimientos funcionales

Remita el **código fuente únicamente** (Archivo .asm) de cada uno de sus programas así como el documento pdf con los diseños, explicaciones, notas de cálculo, etc. La remisión se podrá realizar entre las 9:00 a.m. y las 9:30 a.m. de la fecha indicada para entregar su tarea. El formato del archivo debe ser SUNOMBRE\_Pk\_T2.asm, donde k es el número del problema de la tarea (1,2,3), puede remitir todos estos archivos de manera comprimida en un solo archivo .zip y el documento pdf debe nombrarse SUNOMBRE\_T2.pdf.