



Tarea 1

Indicaciones:

- Debe ser resuelta de manera individual.
 - Debe ser entregada a través del sistema Mediación Virtual, en un PDF denominado Tarea1_<carne>.pdf.
 - La fecha máxima de entrega es el día 20 de Enero del 2021 a las 11:59pm, como se describe en el cronograma del curso.
1. Para el MCU MC9S12DG256 construya tres tablas: una tabla para todos los posibles valores para cada uno de los bits de INITREG, otra con los valores posibles para cada uno de los bits de INITRM y la última con los valores posibles de los bits INITEE para este MCU. Incluya dos columnas en cada tabla donde muestre la dirección de inicio y la dirección de fin del espacio direccionable de cada una de las memorias, según los valores asignados a los bits de los respectivos registros de configuración.
 2. Considere el siguiente segmento de programa:

```
Ldd #$FE3D
Ldx #$1030
Ldab #$10
Std b,X
Bset b,X,$55
Bclr a,X,$37
```

 - a. Luego de que este programa es ejecutado, cuáles son los números que quedan en las posiciones de memoria modificadas por BSet y BClr. Considere que el contenido original de la posición para BClr es \$E5. Cuáles son las direcciones efectivas de esas posiciones de memoria.
 - b. Determine cuántos ciclos de máquina requiere la ejecución de este programa.
 - c. Si el reloj del sistema (sysclk) es de 24 MHz, determine cuánto tiempo toma ejecutar este programa.
 3. Escriba el código de programa en ensamblador para un 9S12 por separado para cada uno de los siguientes requerimientos:
 - a. Cuente el número de ceros contenidos en el Word de las posiciones de memoria cuya dirección efectiva es Y+\$20, donde Y es el registro índice del modelo de programación. Salve el resultado en la dirección de memoria Y+\$EE10.
 - b. Escriba un programa que niegue (toggle) los bits pares y borre los bits impares de la localización de memoria \$2087.



Universidad de Costa Rica
Escuela de Ingeniería Eléctrica
Microprocesadores
IE-0623

EIE

Escuela de
Ingeniería Eléctrica

-
4. Se tiene un conjunto de $N < 200$ valores de 1 byte con signo almacenados en la memoria RAM, a partir de la posición DATOS. Escriba un programa en lenguaje ensamblador para el S12 que copie los valores que sean mayores de -50 a las localizaciones de memoria a partir de la posición MAYORES. Utilice direccionamiento indexado de post incremento tanto para el análisis de los datos como para su movimiento.
 5. Repita el ejercicio #4 utilizando direccionamiento indexado por acumulador.
Observación: Note que en este caso el valor de los punteros permanece inalterado luego de la ejecución del programa.
 6. En una aplicación se tienen N tablas que contienen datos de N nodos de comunicaciones. Se sabe que se tienen hasta 2500 de estos nodos. La información almacenada en cada una de las tablas es:
 - I. Número de ID.
 - II. Número de Ubicación.
 - III. Baud Rate.
 - IV. Throughput.

Todos estos son registros son de tamaño Word, los datos están en binario y están almacenados en ese orden.

Las direcciones de estas N Tablas están almacenadas en un arreglo de $2N$ bytes ubicado en la dirección DIRECCIONES en orden ascendente: Primero está la dirección de la Tabla del Nodo #1, luego la dirección de la Tabla del Nodo #2 y así sucesivamente.

Por medio de un puerto de comunicaciones se recibe el número de nodo que debe activarse, este valor es almacenado en la variable NODO. Para activar ese nodo se debe copiar su información a un buffer de salida ubicado en la dirección BUFFER, pero en orden inverso: primero el throughput y de último el número de ID.

Se debe hacer un programa que active el nodo respectivo. Para la lectura de datos utilice direccionamiento indirecto indexado y para la escritura se debe utilizar direccionamiento indexado de post decremento.

Escriba el código del programa en lenguaje ensamblador del S12 que realice esta tarea.