1. **Ensamble las siguientes instrucciones considerando los códigos de operación de MARIE que el programa inicia en la dirección 0.**

|  |  |
| --- | --- |
| **Código fuente** | **Dirección – Contenido (HEX)** |
| **Load CNT** | 0000 - 1003 |
| **Add ONE** | 0001 - 3004 |
| **Store CN** | 0002 - 2003 |
| **CNT, DEC 0** | 0003 - 0000 |
| **ONE, DEC 1** | 0004 - 0001 |

1. **Desensamble las siguientes 5 localidades de memoria.**

|  |  |
| --- | --- |
| **Dirección - Contenido (hex)** | **Código desensamblado** |
| **0000 – A000** | Clear |
| **0001 – 3004** | loop, Add x |
| **0002 – 6000** | Output |
| **0003 – 9001** | Jump loop |
| **0004 - 0001** | x, DEC 1 |

1. **Escriba una subrutina para realizar 4 corrimientos a la izquierda al registro acumulador.**

/Subrutina 4 corrimientos a la izquierda que equivale a multiplicar por 2, 4 veces

ShL4,   HEX 000

        Store x     /Utilizamos a "X" para guardar el registro acumulador

Loop,   Load x

        Skipcond 000    /Revisamos el bit más significativo para evitar problemas de overflow

        Jump test4      /También revisamos el segundo bit más significativo ya que al recorrerlo a la izq., también mandaría error de overflow

        Subt limit8     /Eliminamos el MSB

        Store x         /Guardamos el valor x modificado

test4,  Load x          /Cargamos el valor x para revisar el segundo MSB

        Subt limit4     /Si x >= 4000(HEX), activamos el MSB, ya que al hacer el desplazamiento se activaría y mandaría error

        Skipcond 000    /Si x-4000h es negativo

        Jump continue

        Jump normal     /El número no causa error y puede ser desplazado con normalidad

continue,   Store x

        Add x

        Add limit8      /Activamos el MSB manualmente

        Jump end

normal, Load x          /Desplazamos con normalidad

        Add x

end,    Store x         /Guardamos el valor modificado

        Load n          /Decrementamos el contador

        Subt ONE

        Store n

        Skipcond 800    /Si y es mayor a 0 saltamos la instrucción

        Jump Return2AC  /  Terminar ciclo

        Jump Loop       /  Repetir el ciclo

Return2AC,  Load x      /Regresamos el valor modificado al acumulador

        JumpI ShL4      /Terminar subrutina

limit4, HEX 4000

limit8, HEX 8000

ONE, dec 1

x, dec 0

n, dec 4

1. **Escriba una subrutina para realizar n corrimientos al registro AC donde N puede ser un valor de 0 a 15**

/Subrutina N corrimientos a la izquierda que equivale a multiplicar por 2, N veces

ShLN,   HEX 000

        Store x     /Utilizamos a "X" para guardar el registro acumulador

Loop,   Load x

        Skipcond 000    /Revisamos el bit más significativo para evitar problemas de overflow

        Jump test4      /También revisamos el segundo bit más significativo ya que al recorrerlo a la izq., también mandaría error de overflow

        Subt limit8     /Eliminamos el MSB

        Store x         /Guardamos el valor x modificado

test4,  Load x          /Cargamos el valor x para revisar el segundo MSB

        Subt limit4     /Si x >= 4000(HEX), activamos el MSB, ya que al hacer el desplazamiento se activaría y mandaría error

        Skipcond 000    /Si x-4000h es negativo

        Jump continue

        Jump normal     /El número no causa error y puede ser desplazado con normalidad

continue,   Store x

        Add x

        Add limit8      /Activamos el MSB manualmente

        Jump end

normal, Load x          /Desplazamos con normalidad

        Add x

end,    Store x         /Guardamos el valor modificado

        Load n          /Decrementamos el contador

        Subt ONE

        Store n

        Skipcond 800    /Si y es mayor a 0 saltamos la instrucción

        Jump Return2AC  /  Terminar ciclo

        Jump Loop       /  Repetir el ciclo

Return2AC,  Load x      /Regresamos el valor modificado al acumulador

        JumpI ShLN      /Terminar subrutina

limit4, HEX 4000

limit8, HEX 8000

ONE, dec 1

x, dec 0

n, dec 0

## **Anexos:**

Carpeta a Drive con códigos fuente Marie.JS

<https://drive.google.com/drive/folders/1ZYvExCBNVY_5Q6lwtbpoTu7Na5A6XAmz?usp=sharing>