
Instituto Tecnológico y de Estudios Superiores de Monterrey



**Tecnológico
de Monterrey**

Tarea 3 : Programación en ensamblador

Fecha: 9 de octubre del 2022

**Modelación de sistemas mínimos y arquitecturas
computacionales (Gpo 200)**

Alumnos:

A01423940 Daniel De Luna Díaz - Daniel De Luna Díaz

1. Redacta brevemente la respuesta a los conceptos de Revisión de conceptos básicos

1. ¿Cuál es la función de un CPU?

Es el componente más importante dentro del PC ya que es el encargado de dirigir todas las tareas que lleva a cabo el equipo y de ejecutar el código de los diferentes programas.

2. ¿A qué propósito sirve que exista un “datapath”?

Un “datapath” es una colección de unidades funcionales, como unidades aritméticas lógicas o multiplicadores que realizan operaciones de procesamiento de datos, registros y buses. Junto con la unidad de control, compone la unidad central de procesamiento (CPU).

3. ¿Qué tarea realiza la unidad de control?

La unidad de control es un componente de la unidad central de procesamiento de una computadora que dirige el funcionamiento del procesador. Una CU generalmente usa un decodificador binario para convertir instrucciones codificadas en señales de tiempo y control que dirigen la operación de las otras unidades.

4. ¿Dónde están localizados los registros y cuáles son los tipos que existen?

Un registro es una memoria de alta velocidad y poca capacidad, integrada en el microprocesador, que permite guardar transitoriamente y acceder a valores muy usados, generalmente en operaciones matemáticas.

5. ¿Cómo sabe el ALU cuáles son las funciones que debe realizar?

La señal de control es la encargada de seleccionar la operación a realizar por la ALU

6. ¿Por qué un bus puede ser un “cuello de botella”?

El bus puede saturarse si hay varios dispositivos comunicándose y transfiriendo datos al mismo tiempo, especialmente si el bus en sí tiene una tasa de transferencia poco baja

7. Explica la diferencia entre bus de datos, bus de direcciones y bus de control.

Bus de direcciones: Transporta las direcciones de memoria desde el procesador a otros componentes, como el almacenamiento primario y los dispositivos de entrada/salida.

Bus de datos: Transporta los datos entre el procesador y otros componentes

Bus de control: Transporta señales de control desde el procesador a otros componentes.

8. ¿Cuál es el ciclo del bus?

La operación básica del bus se denomina ciclo de bus. Un ciclo permite realizar una transferencia elemental de un dato entre dos dispositivos. En esta transferencia, la información se lleva de un elemento que se denomina fuente a otro que se denomina destino.

9. Nombra tres tipos diferentes de buses y dónde se encuentran en una computadora.

Bus de CPU - Integrado en la placa base.

Bus serial - Integrado generalmente en la placa base pero se accesa mediante conectores (USB)

Bus de Periféricos - Integrado generalmente en la placa base pero se accesa mediante conectores (SATA)

10. ¿Cuál es la diferencia entre un bus síncrono y asíncrono?

La transmisión de datos asíncrona es en serie, envía bits de datos uno tras otro a través de un solo canal. La transmisión de datos síncrona es en paralelo envía múltiples bits de datos al mismo tiempo a través de múltiples canales

11. Explica la diferencia entre ciclo de reloj y frecuencia de reloj.

Un ciclo de reloj es un período único de una señal de reloj oscilante. La velocidad, el índice y la frecuencia del reloj se usan para describir lo mismo: el número de ciclos de reloj por segundo, medidos en hercios (Hz)

12. ¿Cuál es la función de una interfaz de E/O?

La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina,

13. Explica la diferencia entre E/S con mapeo de memoria y E/S basada en instrucción.

La E/S mapeada en memoria usa el mismo bus de direcciones para memoria y dispositivos de E/S, y las instrucciones de la CPU usadas para acceder a la memoria son también usadas para acceder a los dispositivos.

La E/S independiente usa un tipo especial de instrucciones de la CPU para implementar E/S.

Principalmente en microprocesadores Intel encontramos las instrucciones IN y OUT que pueden leer y escribir un único byte en un dispositivo de E/S. Estos tienen un espacio de direcciones separadas de la memoria, llevado a cabo o bien por un "pin E/S" extra en la CPU o bien por un bus entero dedicado a E/S.

14. ¿Cuál es la diferencia entre byte y word?

Un byte son 8 bits mientras que un word son 2 bytes (16 bits)

15. Explica la diferencia entre direccionamiento-byte y direccionamiento-word.

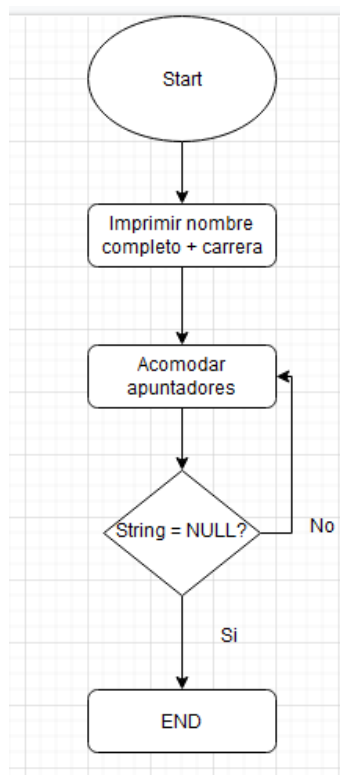
Transfiere un byte o palabra de datos inmediato hacia el operando destino. Este modo es usado para inicializar registros o localidades de memoria y para operar sobre ellos con valores constantes de datos.

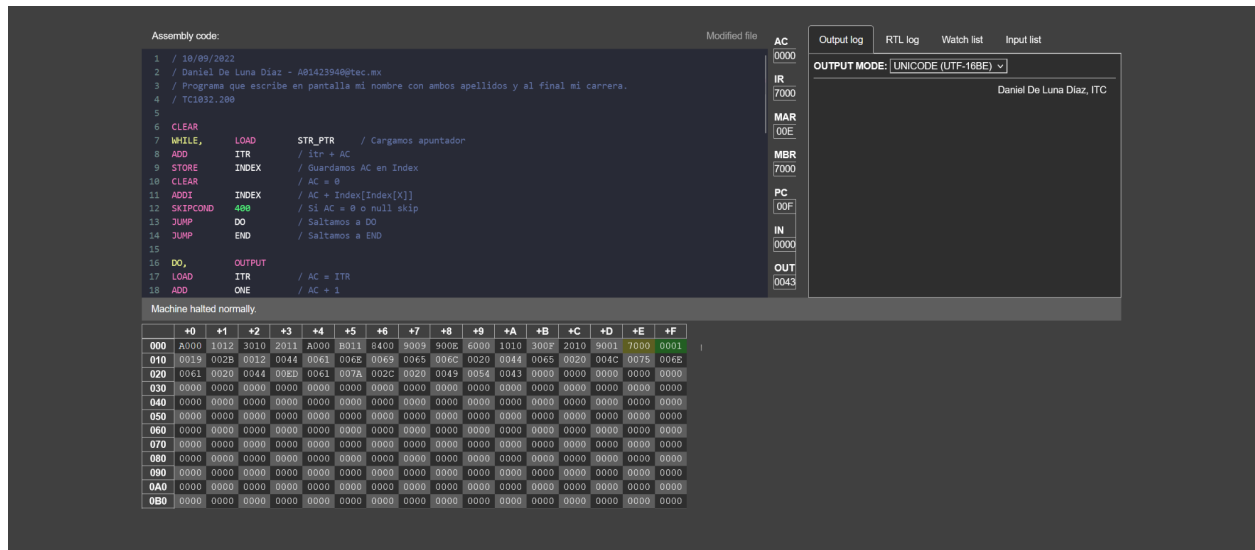
2. Ensambla y prueba los ejemplos que vienen al descargar el simulador.

1. Realiza un programa que escriba en pantalla tu nombre con ambos apellidos y al final tu carrera. Utiliza mayúsculas y minúsculas.

Bosquejo:

- **Tomar una letra e imprimirla a la pantalla**
- **Imprimir nombre completo + nombre de carrera**
- **Iterar hasta llegar al final de la frase o NULL**

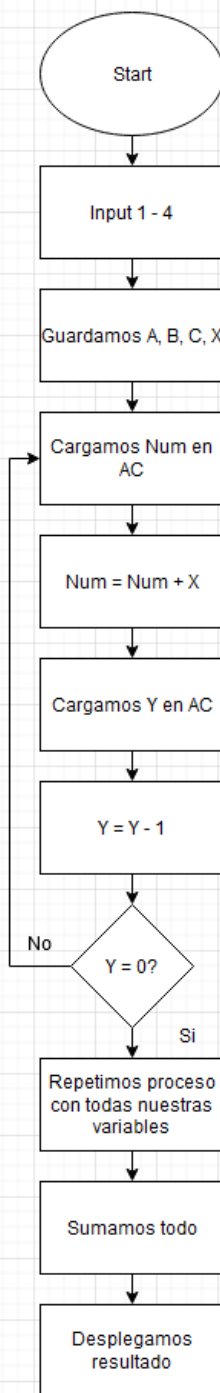




- Realiza un programa que calcule: $Y = X - 2 * A - 3 * B - 4 * C$. X, A, B y C deben ser proporcionadas por el usuario.

Bosquejo:

- Tomar 4 inputs
- Guardarlos en las variables A, B, C, X
- Hacer la ecuación brindada
- Guardar resultado
- Desplegar resultado



Assembly code:
Autosaved file

```

1 / 10/09/2022
2 / Daniel De Luna Diaz - A8142394@tec.mx
3 / Programa que calcula: Y=X-2*A-3*B-4*C
4 / TC1892.208
5
6 CLEAR
7
8 / Negamos nuestras constantes
9 Load TWO
10 Subt TWO
11 Subt TWO
12 Store TWO
13
14 Load THREE
15 Subt THREE
16 Subt THREE
17 Store THREE
18

```

AC
FFF0

IR
7000

MAR
048

MBR
7000

PC
049

IN
0004

OUT
FFF0

Output log
RTL log
Watch list
Input list

OUTPUT MODE:
DEC

1
2
3
4
-16

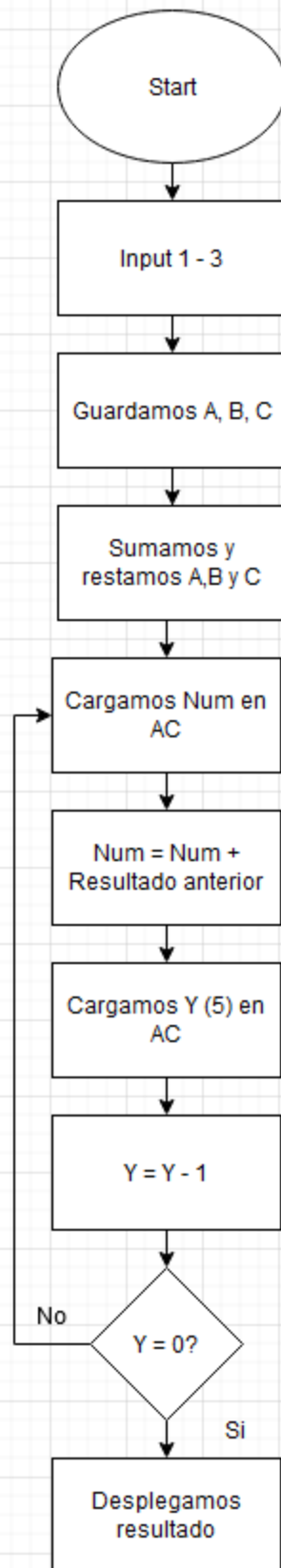
Machine halted normally.

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0A0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0B0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0C0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0D0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0E0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0F0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
100	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
110	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
120	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
130	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

3. Realiza un programa que calcule: $Y=5*(A-B+C)$. A, B y C denes ser proporcionadas por el usuario.

Bosquejo:

- Tomar 3 inputs
- Guardarlos en las variables A, B, C,
- Hacer la ecuación brindada
- Guardar resultado
- Desplegar resultado



Assembly code: Autosaved file

```

1 / 18/09/2022
2 / Daniel De Luna Díaz - AB1423948@tec.mx
3 / Programa que calcula:  $Y=5*(A-B+C)$ .
4 / TC1032.200
5
6 CLEAR
7
8 INPUT
9 STORE A
10 INPUT
11 STORE B
12 INPUT
13 STORE C
14
15 LOAD A
16 OUTPUT
17 LOAD B
18 OUTPUT

```

Machine halted normally.

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0A0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0B0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0C0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0D0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0E0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0F0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
100	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
110	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
120	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
130	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

AC 000A
IR 7000
MAR 01D
MBR 7000
PC 01E
IN 0003
OUT 000A

Output log RTL log Watch list Input list

OUTPUT MODE: DEC

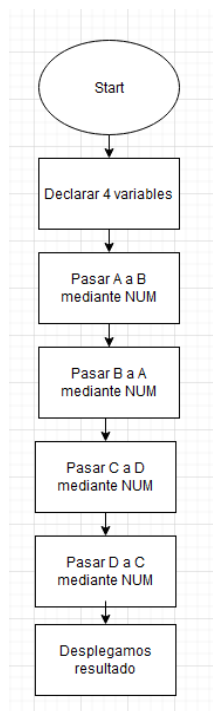
1
2
3
10

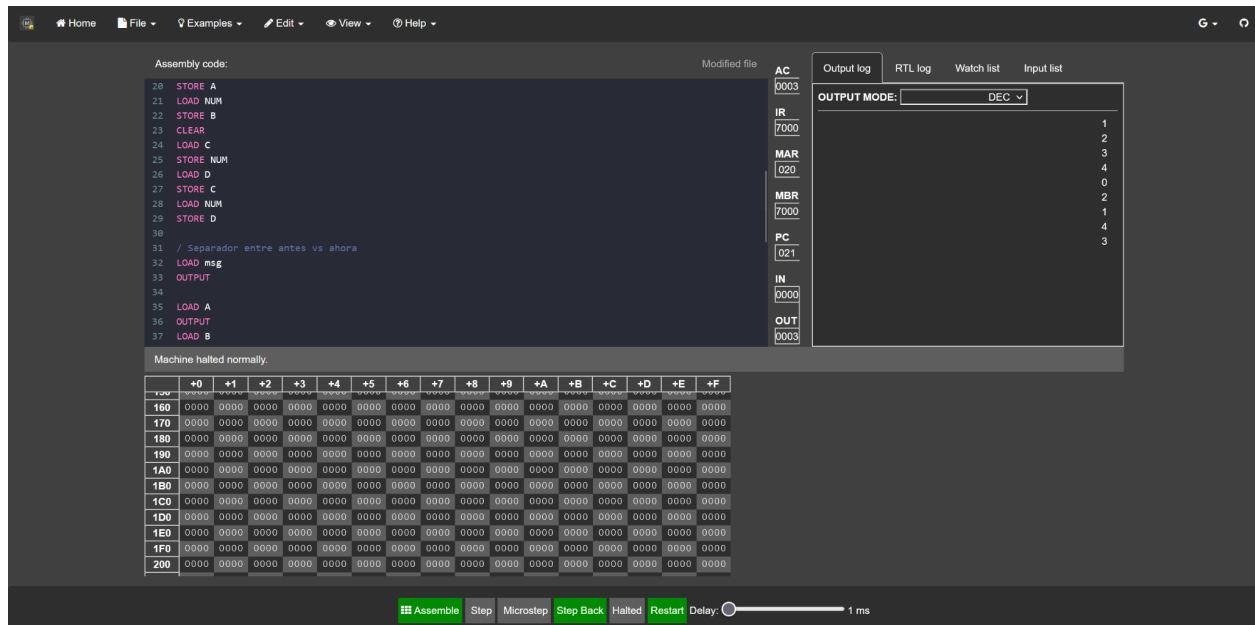
Assemble Step Microstep Step Back Halted Restart Delay 1 ms

4. Realiza un programa que intercambie los datos almacenados en 4 localidades de memoria.

Bosquejo:

- Declarar 4 variables para reservar memoria
- Pasar el valor de una variable a otra mediante una variable intermediaria
- Desplegar resultado (Antes y después)

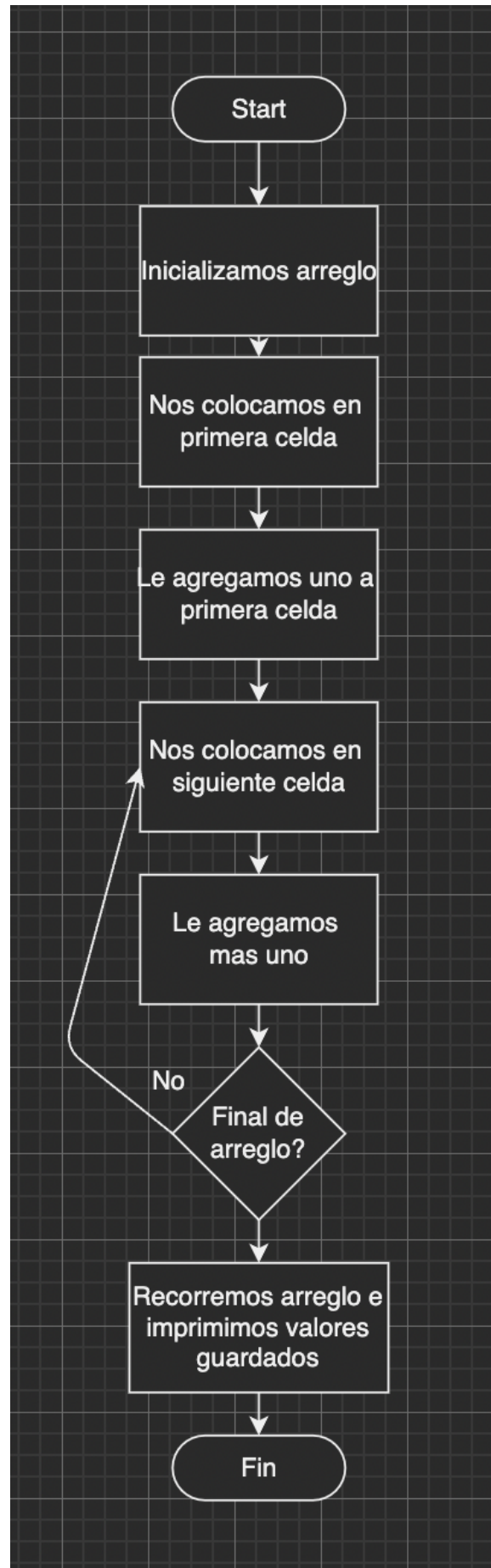




- Realiza un programa que tome la primera localidad de memoria de una secuencia de 6 números, llenando las restantes cinco con el valor de la anterior más uno.

Bosquejo:

- Inicializar arreglo
- Recorrer arreglo sumandole el numero anterior + 1 a cada celda
- Guardar
- Desplegar arreglo completo



Assembly code:
Autosaved file

Output log
RTL log
Watch list
Input list

AC
0001

IR
7000

MAR
022

MBR
7000

PC
023

IN
0000

OUT
0001

OUTPUT MODE:
DEC

6
5
4
3
2
1

```

1 / 09/10/2022
2 / Daniel De Luna Diaz - A0142394@tec.mx
3 / Programa que toma la primera localidad de memoria de una secuencia de 6 números,
4 / llenando las restantes cinco con el valor de la anterior más uno.
5 / TCI032.200
6
7 CLEAR
8
9 / Guardamos arreglo en apuntador
10 LOADI ARR
11 ADD ONE
12 STORE PTR
13
14 STORENUMS, CLEAR
15 / Cargamos apuntador
16 LOAD PTR
17 ADD ONE
18 STORE PTR

```

Machine halted normally.

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	A000	D027	3023	2025	A000	1025	3023	2025	A000	1027	3024	E025	A000	1024	3023	2024
010	A000	1026	4023	2026	8400	9004	D025	6000	4023	6000	4023	6000	4023	6000	4023	6000
020	4023	6000	7000	0001	0006	D02E	0000	0001	0000	0001	0002	0003	0004	0005	0006	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
090	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0A0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0B0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

Green: Current PC, Yellow: Current MAR

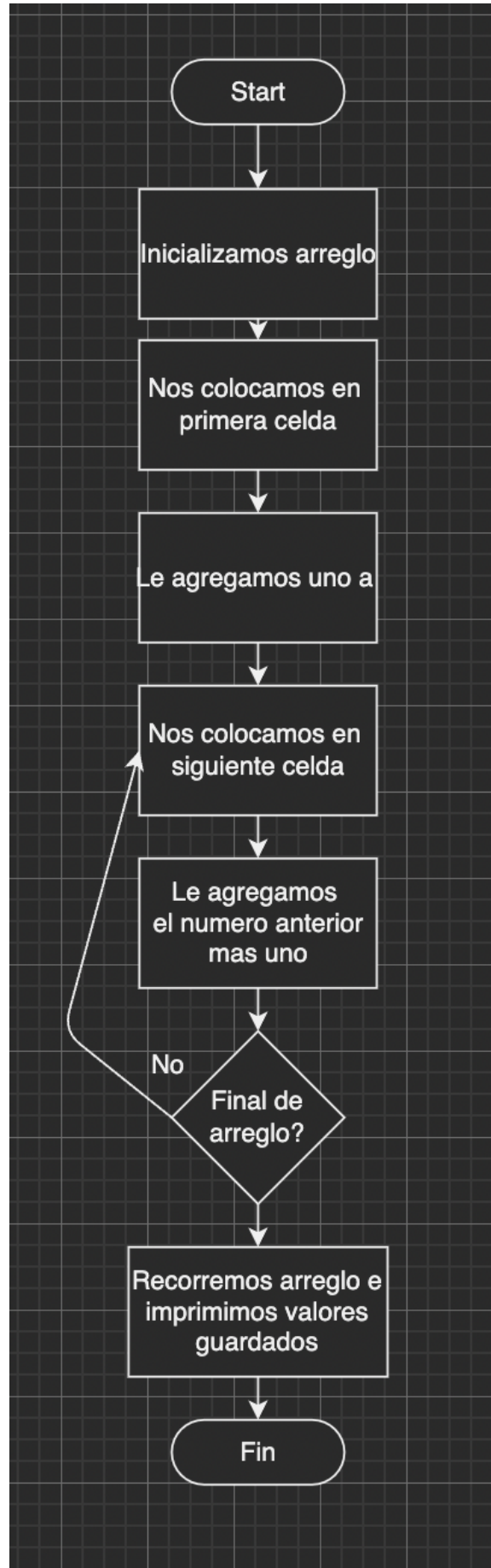
III Assemble
Step
Microstep
Step Back
Halted
Restart

Delay: 1 ms

- Realiza un programa que tome una serie de cinco localidades de memoria con unos valores iniciales y a cada una suma un valor, la primera 1, la segunda 2, etc.

Bosquejo:

- Inicializar arreglo
- Recorrer arreglo sumándole el numero indicado dependiendo de su ubicación en el arreglo
- Guardar
- Desplegar arreglo completo



Assembly code:

Modified file

```
1 / 09/10/2022
2 / Daniel De Luna Diaz - AB1423948@tec.mx
3 / programa que tome una serie de cinco localidades de memoria con unos valores iniciales
4 / y a cada una sume un valor, la primera 1, la segunda 2, etc.
5 / TC1032.200
6
7 CLEAR
8
9 / Guardamos arreglo en apuntador
10 LOADI ARR
11 STOREI PTR
12
13 LOOP1, LOADI PTR
14 ADD COUNTER
15 STOREI PTR
16
17 CLEAR
18
```

Machine halted normally.

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000	A000	B029	2027	B027	3026	E027	A000	1027	3025	2027	A000	1026	3025	2026	A000	1028
010	4025	2028	8400	9003	1027	4025	2027	1026	4025	2026	B027	6000	1027	4025	2027	1026
020	4025	2026	8400	901A	7000	0001	0000	B028	0000	0002	0002	0003	0004	0005	0000	0000
030	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
040	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
050	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
060	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
070	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
090	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0A0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
0B0	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

AC

0000

IR

7000

MAR

024

MBR

7000

PC

025

IN

0000

OUT

0002

Output log

RTL log

Watch list

Input list

OUTPUT MODE: DEC

5

4

3

2

2