

Tarea #4: Microprocesadores II - 2024
Roger Daniel Piovet García C15990

Memoria de cálculo

El periodo de la interrupción RTI, TRTI, viene dado por:

$$T_{RTI} = \frac{(N+1) \cdot 2^{M+9}}{Osc_Clk} \quad (1)$$

Donde $N = RTR[3:0]$, $M = RTR[6:4]$ en el registro RTICTL, y $Osc_Clk = 8MHz$ para la Dragon 12+2. Se define $T_{RTI} = 1ms$. Si $M=1$, según (1), se tiene:

$$\Rightarrow N = \frac{T_{RTI} \cdot Osc_Clk}{2^{10}} - 1$$

$$\Rightarrow N \approx 6.81 \approx 7$$

Para $N=7$, de (1) se obtiene el valor aproximado

$$\hat{T}_{RTI} = \frac{8 \cdot 2^{10}}{8MHz} = 1.024ms$$

Lo cual corresponde a un porcentaje de error:

$$\%_{RTI} = \frac{|T_{RTI} - \hat{T}_{RTI}|}{T_{RTI}} \cdot 100\% = 2,4\%$$

Lo cual es aceptable para esta aplicación, ya que no precisa mediciones temporales exactas.

Por tanto, RTICTL = 0: RTR[6:0] = 17

(1)

Rutina principal (MAIN)

En esta rutina, se realiza la configuración del hardware, la inicialización de las estructuras de datos utilizadas por las subrutinas del programa, y despachaa las tareas que implementan las funciones del programa.

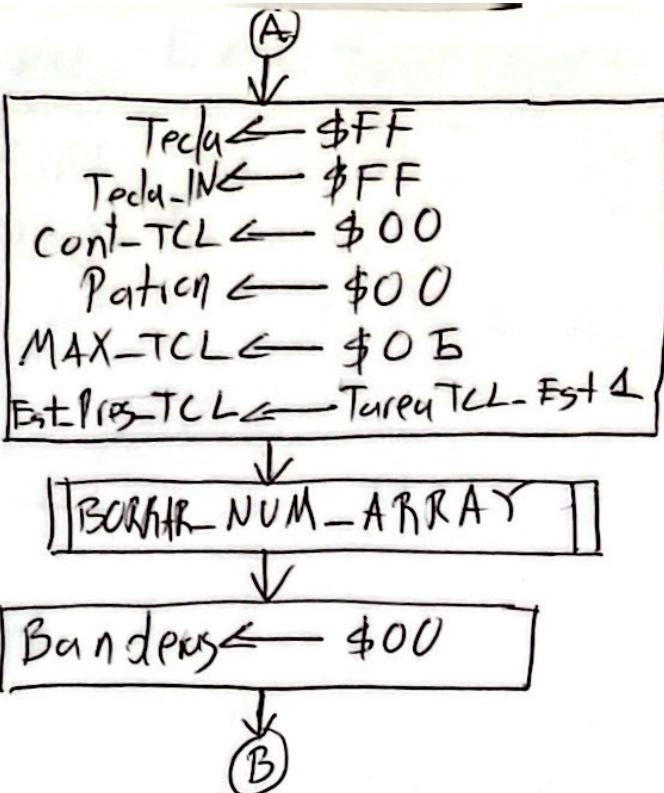
Para efectos del diagrama de flujo mostrado, se carga MAX-TCL con 5. Este valor puede ser modificado por el usuario que utilice el código que implementa los diagramas de flujo de esta tarea dentro del rango permisible para esta variable, donde:

$$\text{MAX_TCL} \in \{1, 2, \dots, 5\}$$

MAIN

```

DDR.B.0 ← 1
DDR.B.7 ← 1
DDR.J.1 ← 1
PTJ.1 ← 0
DDR.P ← $0F
PTP ← $0F
RTICL ← $17
RTIE ← 1
PUPAE ← 1
    
```



(A)

(B)

(B)

Est-Prog-LeerPB ← LeerPB-Est+1

```

Timer1ms ← +Timer1ms
Timer10ms ← +Timer10ms
Timer100ms ← +Timer100ms
Timer1s ← +Timer1s
    
```

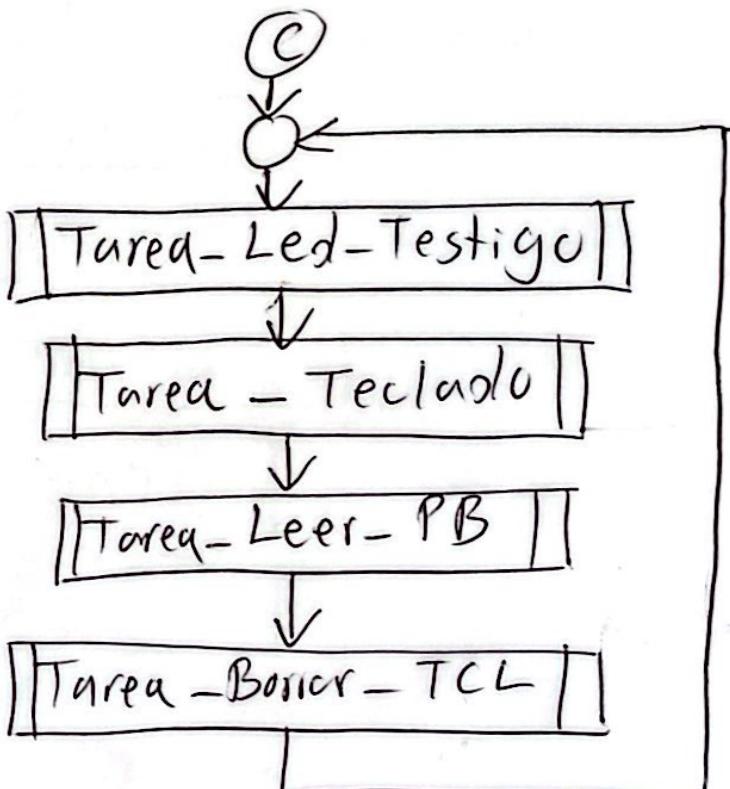
```

TIMER-LED-Testigo ← +TimerLDTst
Timer-SHP ← $00
Timer-LP ← $00
Timer-Reb-PB ← $00
Timer-RebTCL ← $00
    
```

```

SP ← $3BFF
I ← 0
    
```

(C)



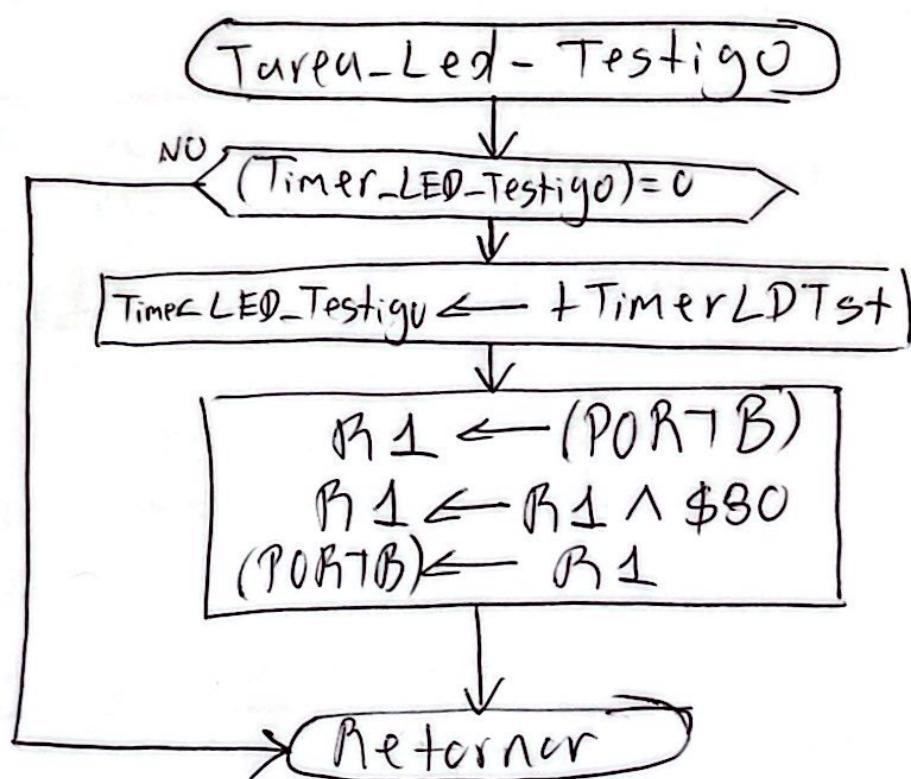
(3)

Subrutina Tarea-Led-Testigo

En esta subrutina, se asegura el correcto funcionamiento de la máquina de tiempos al parpadear el LED conectado al PB7 en la Dragon 12+2

Estructuras de datos

*Timer-LED-Testigo: Timer que define el periodo de parpadeo del LED conectado a PB7. Se define con un valor de carga de 1, representando 1 segundo, ya que el timer se coloca en la tabla Tabla-Timers Base 15.



(4)

Subrutina Tarea - Teclado

Esta subrutina implementa una máquina de estados para la lectura del teclado matricial conectado a PORTA en la Dragon 12+2.

Estados

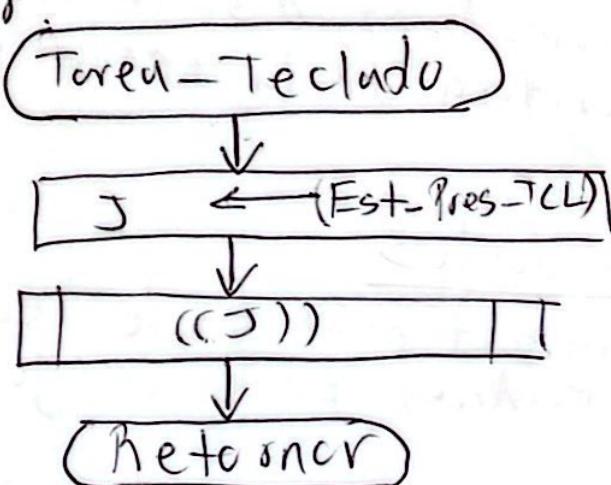
- * Tarea TCL-Est1: Se lee una tecla presionada a través de la subrutina Leer_Teclado, y se carga el timer de supresión de rebotes.
- * Tarea TCL-Est2: Se espera a que finalice el timer de rebotes, y se verifica que la lectura haya sido válida; es decir, si la tecla sigue presionada.
- * Tarea TCL-Est3: Si la tecla es retenida por el usuario, se espera a que deje de serlo para que sea almacenada en Num_Array.
- * Tarea TCL-Est4: Se analizan aquellos casos de esquina en donde se está escribiendo la primera o última tecla en Num_Array, o si la tecla es Borrar o Enter, para la escritura de la tecla presionada en Num_Array.

Estructuras de datos

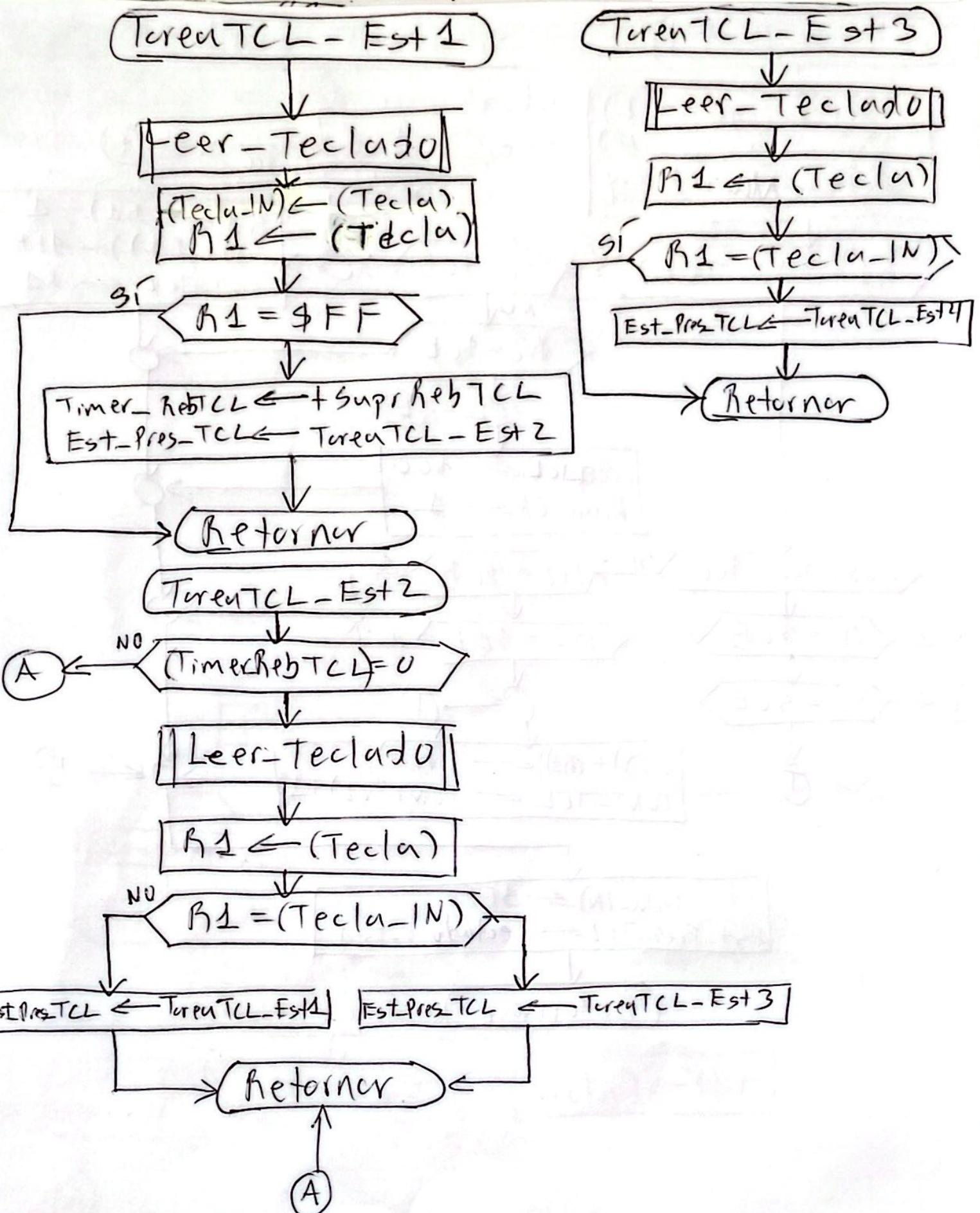
- * MAX_TCL: Cantidad máxima de teclas que se presionan y se almacenan en Num_Array. En el rango $\{1, \dots, 5\}$.
Variable tipo byte.

- * Tecla: Variable tipo byte que almacena el valor de una tecla presionada o \$FF si no se presionó.
- * Banderas: Variable bandera tipo byte. X:X:X:X:ArrayOK:LungP:
Se utiliza ArrayOK para indicar la formación de una secuencia de teclas válidas.

- * Tecla-N: Variable tipo byte que almacena temporalmente el valor de una tecla presionada.
- * Cont-TCL: Variable tipo byte que almacena el offset para indexar la tabla Teclas según la tecla presionada.
- * Pation: Variable tipo byte que almacena una máscara para leer las teclas presionadas en el PORTA
- * Est-Pres-TCL: Variable tipo word que almacena el próximo estado a cargarse en la máquina de estados Tarea-Teclado
- * Num-Array: Arreglo que almacena una secuencia de teclas presionadas o ~~FFF~~^{en todos sus posiciones} si no se han presionado ninguna tecla.
- * Teclas: Tabla que almacena el valor de las teclas que se pueden presionar en el teclado matricial 0, 1, 2, ..., 9, B, E, donde B y E son las teclas de Borrar y Enter.



(6)



*(Se define el tiempo de supresión de rebotes de 5ms)

(Teclado - Est+4)

$M1 \leftarrow (\text{CONT-TCL})$
 $R2 \leftarrow (\text{Tecla-IN})$
 $J \leftarrow \text{Num-Array}$

$R1 = (\text{MAX-TCL})$

B
↓
 $(\text{CONT-TCL}) = \$00$ → Sí → F

$R1 \leftarrow (R1) - 1$
 $((J) + (R1)) \leftarrow \FF
 $\text{CONT-TCL} \leftarrow (R1)$

$R2 = \$0B$ → O

NO

$R2 = \$0E$

E

$(\text{CONT-TCL}) \leftarrow \00
 $\text{Array-OK} \leftarrow 1$

$(\text{CONT-TCL}) = \$00$

Sí → R2 = \$0B

Sí → R2 = \$0E

Sí → B

Sí → E

Sí → T

$((J) + (R2)) \leftarrow (R2)$
 $\text{CONT-TCL} \leftarrow (\text{CONT-TCL}) + 1$

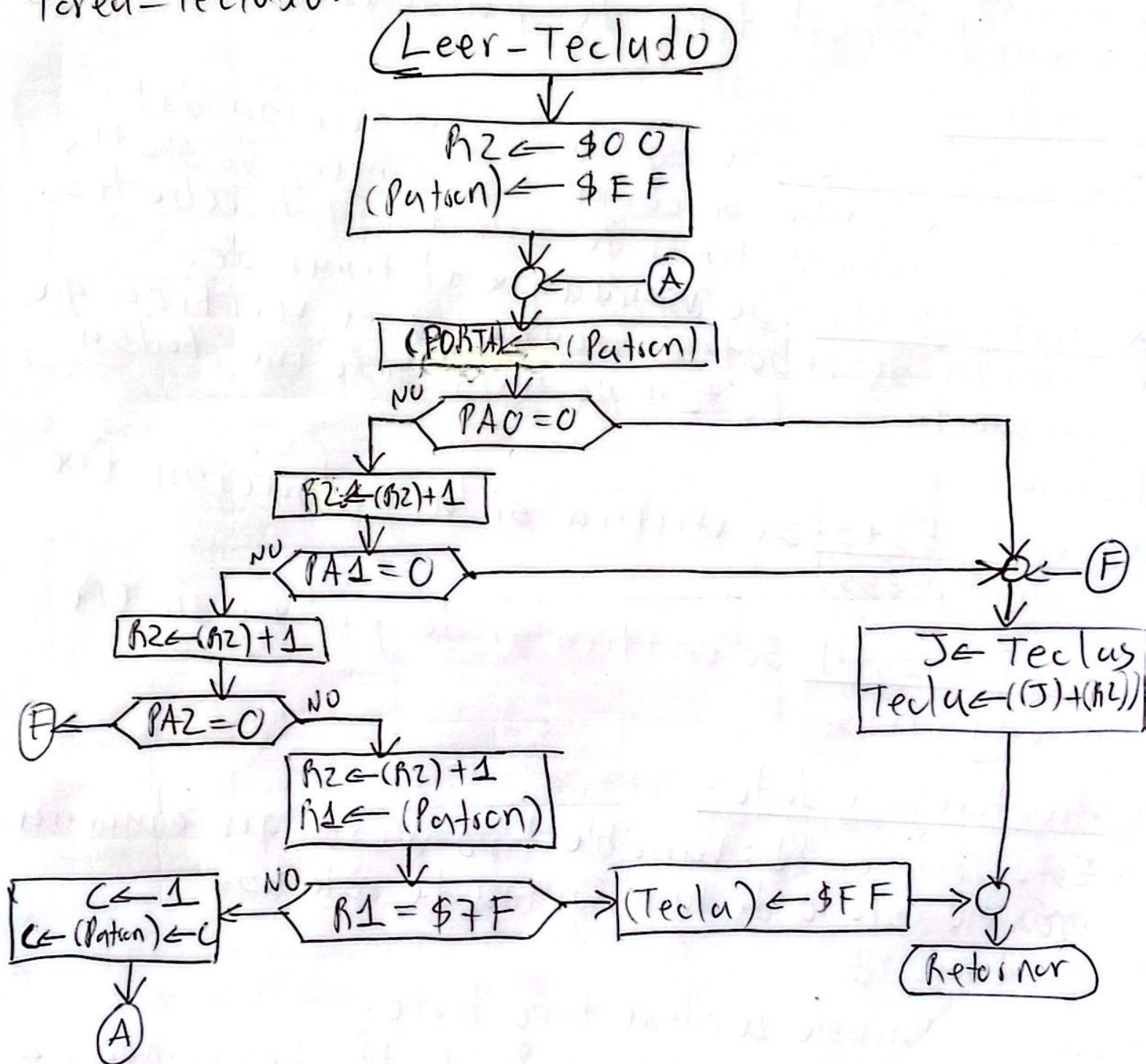
$(\text{Tecla-IN}) \leftarrow \FF
 $\text{Est-Pres-TCL} \leftarrow \text{TareaTCL-Est+1}$

Retornar

8

Subrutina Leer-Teclado

Esta subrutina lee una tecla presionada en el teclado matricial de la Dragon 12+2. La tecla leída es almacenada en la variable tecla. Utiliza las mismas estructuras de datos que Tarea-Teclado.



Subrutina Tarea-Leer-PB

Esta subrutina implementa una máquina de estados para la lectura del botón pulsador P10 en la Dsayaun 12+2. Además, se detecta si la pulsación fue corta o larga, levantando las banderas ShortP y LongP en Banderas en caso que suceden este tipo de pulsaciones.

Estados

* LeerPB-Est1: Se espera la pulsación del botón. Si se pulsa, se cargan los timers de ShortP y LongP, así como el timer de supresión de rebotes.

* LeerPB-Est2: Se espera que el timer de supresión de rebotes acabe, y se verifica que la pulsación del botón no haya sido una falsa lectura.

* LeerPB-Est3: Se verifica si la pulsación fue un short press.

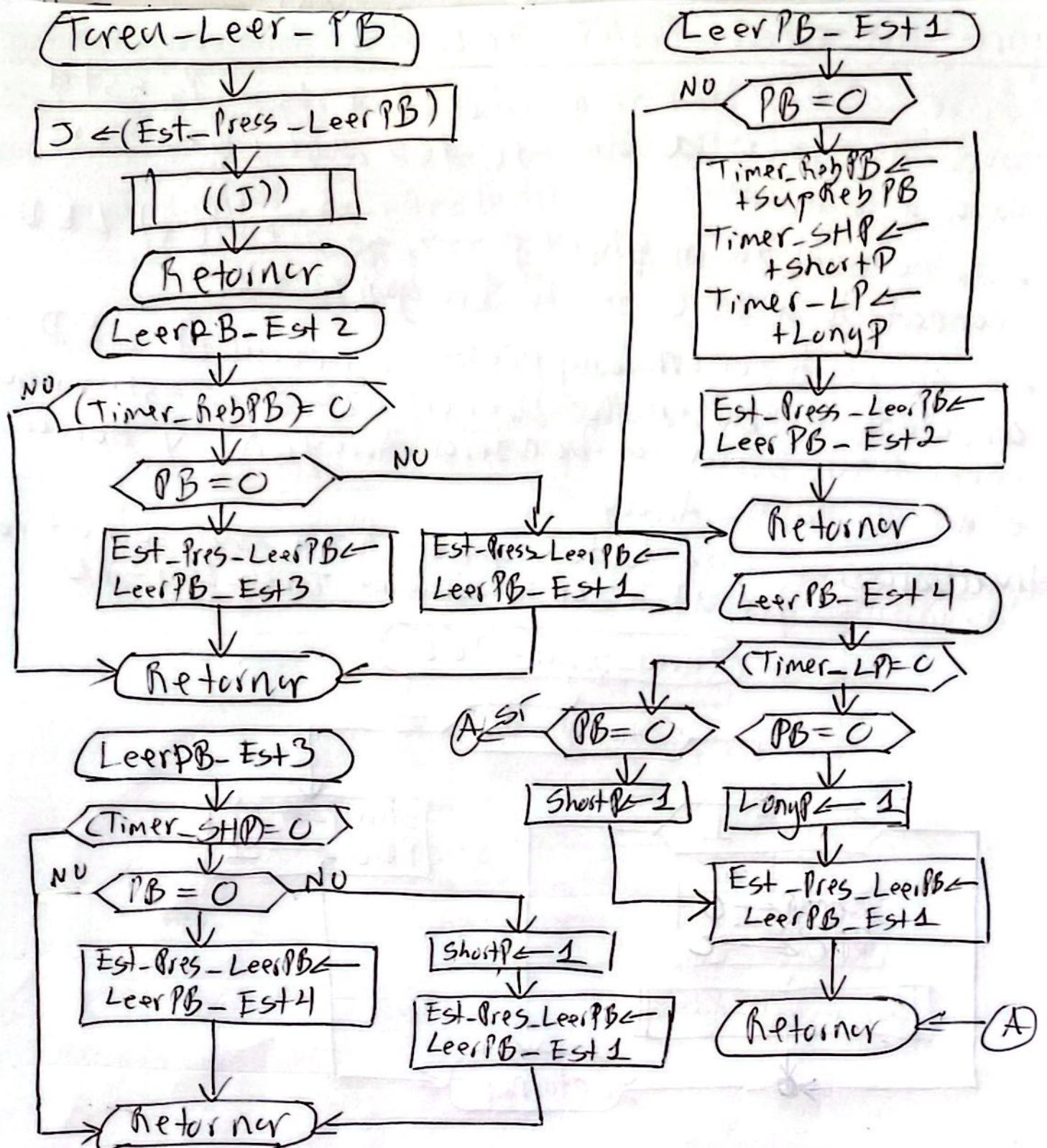
* LeerPB-Est4: Se verifica si la pulsación fue un long press.

Estructuras de datos

* Est_PressLeerPB: Variable tipo word que almacena el próximo estado de la máquina de estados Tarea-Leer-PB.

* Banderas: Variable bandera tipo byte.

X:X:X:X:Array-OK:LongP:ShortP. Los primeros 2 bits se utilizan para clasificar la pulsación. (10)



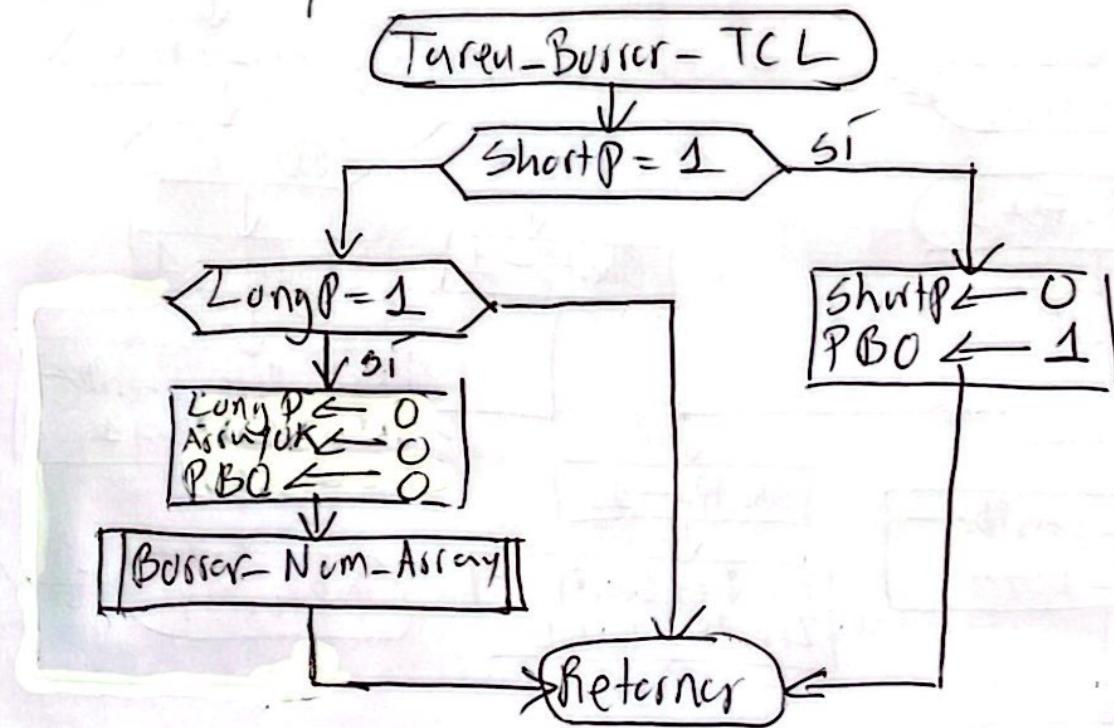
* (se define el tiempo de supresión de rebotes de 50ms)
 (el tiempo de short press de 100ms, y de long press de 2s)

Subrutina Tarea - Borrar - TCL

En base a las banderas ShortP y LongP, esta subrutina ejecuta una de dos acciones por llamada a la misma:

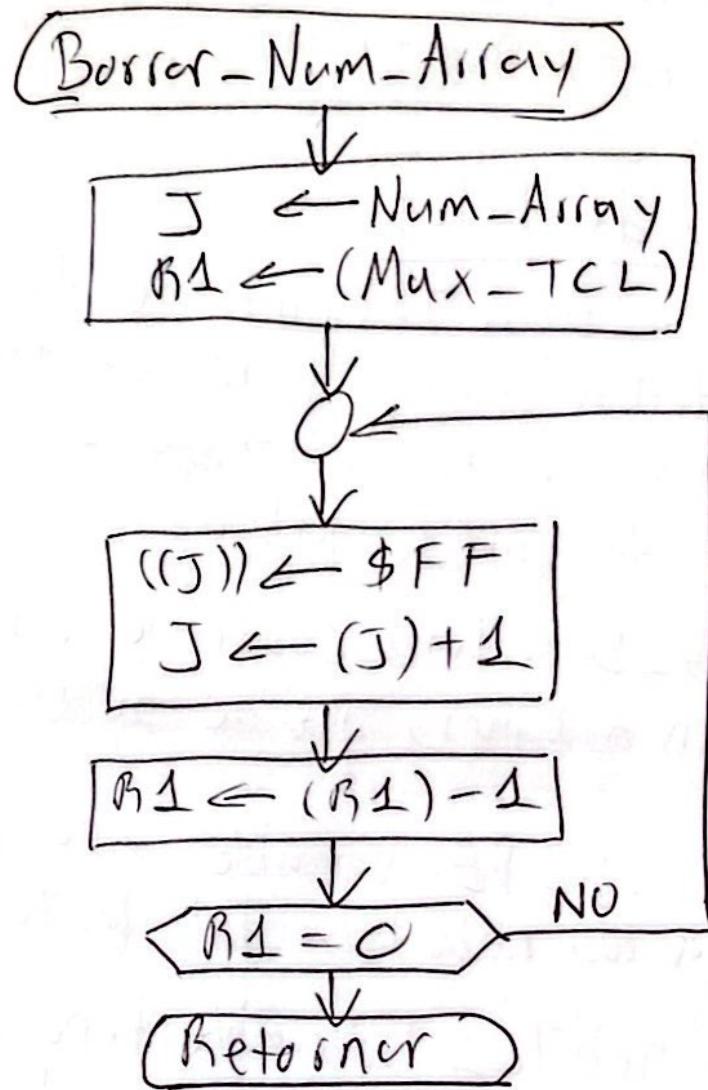
1. Si se realiza un short press, se prende el LED conectado a PBO en la Dragon 12+2
2. Si se realiza un long press, se apaga el LED conectado a PBO en la Dragon 12+2 si estuviera encendido, borra la bandera ArrayOK, y borra el arreglo Num-Array.

Las estructuras de datos que utiliza esta subrutina son las mismas que utiliza la subrutina Tarea-Leer-PB



Subrutina Borrar-Num-Array

Esta subrutina borra Num-Array. Las estructuras de datos que utiliza son las mismas que utiliza la subrutina Tarea-Telclado.



Subrutina Maquina-Tiempos

Esta subrutina atiende a interrupciones RTI con un periodo de 1ms (ISR). A partir de un arreglo de tablas, se definen bases de tiempos y timers que pueden ser utilizados en aplicaciones específicas.

Estructuras de datos

* Tabla-Timers-BaseT: Contiene ~~timers~~ variables tipo word que corresponden a timers para generar bases de tiempo de 1ms, 10ms, 100ms, y 1s. Tiene \$FFFF como indicador de fin de tabla.

* Tabla-Timers-Base1ms: Contiene variables tipo byte que corresponden a timers que se decrementan cada 1ms.

 → Timer-Reb-PB: Variable tipo byte que maneja los rebotes del botón pulsador

 → Timer-RebTCL: Variable tipo byte timer que maneja los rebotes del teclado matricial

* Tabla-Timers-Base10ms: Contiene variables tipo byte que corresponden a timers que se decrementan cada 10ms

 → Timer-SHP: Variable tipo byte timer utilizado para identificar un short press.

(24)

* Tabla-Timers-Base100ms: Contiene variables tipo byte que corresponden a timers que se decrementan cada 100ms.

↳ Timer1-100ms: Variable tipo byte timer

* Tabla-Timers-Base1s: Contiene variables tipo byte que corresponden a timers que se decrementan cada 1s

- ↳ Timer_LP: Variable tipo byte timer utilizada para identificar un long press
- ↳ Timer_LED-Testigo: Variable tipo byte timer utilizada para pulsar un LED en la subrutina Tarea-LED-Testigo.

Subrutina Decre-Timers-BaseT

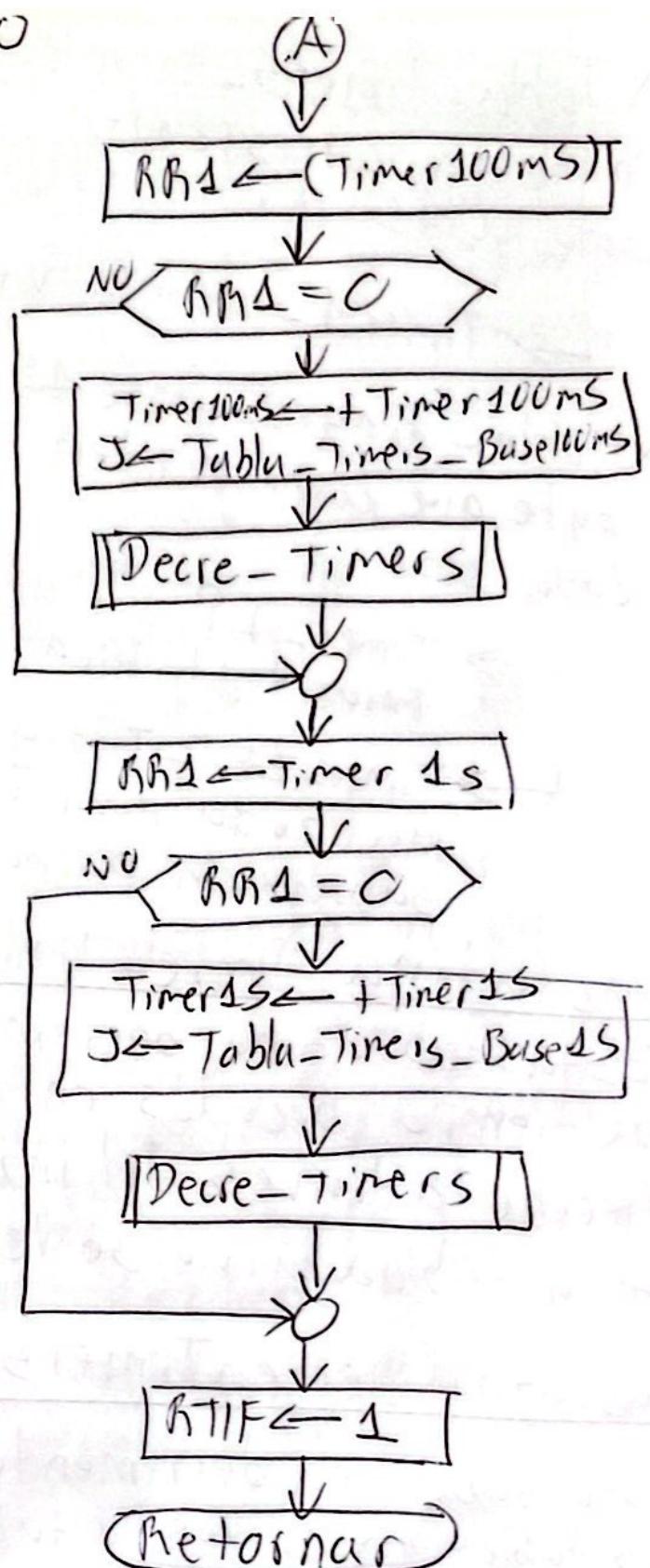
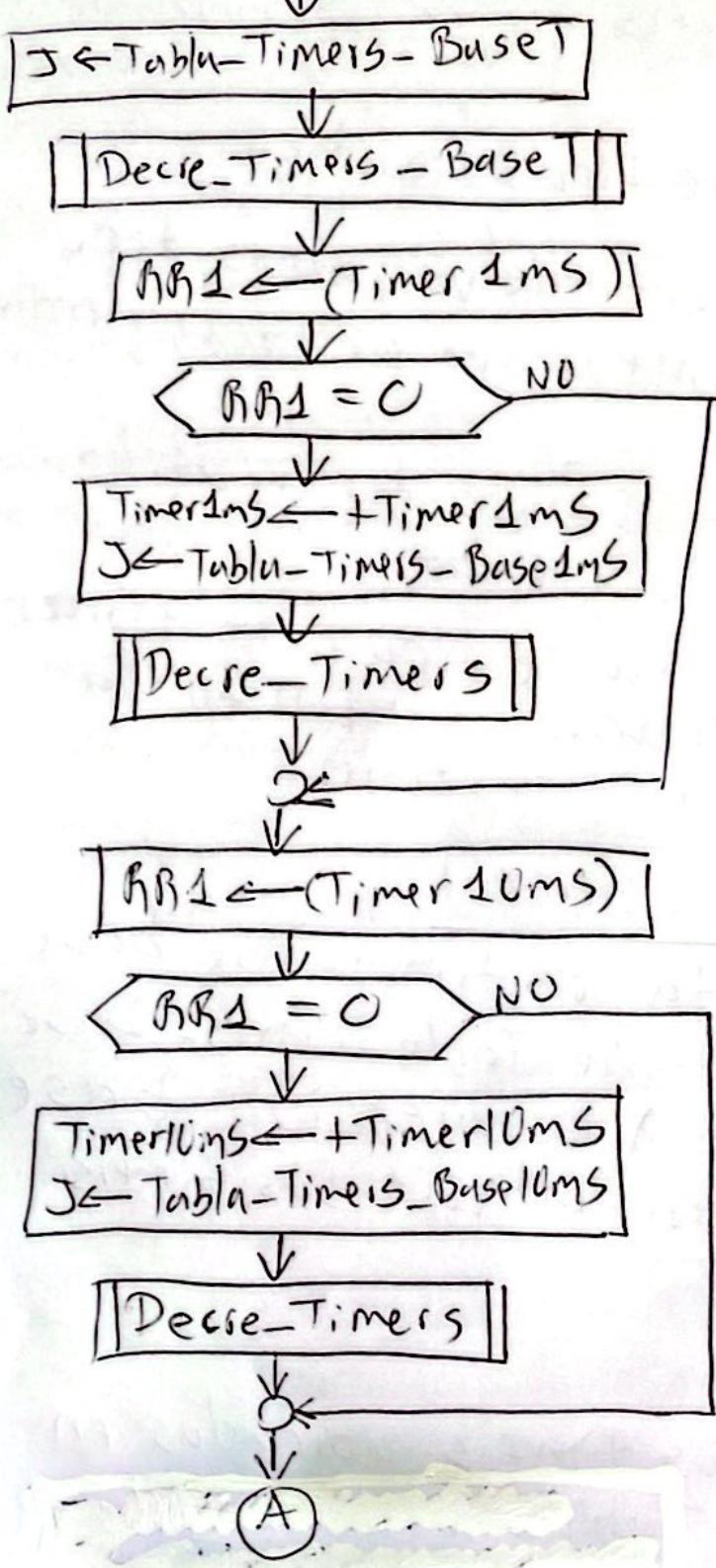
Esta subrutina decremente los timers de base de tiempo ubicados en la tabla Tabla-Timers-BaseT. Recibe a través del índice X la dirección base de la tabla a la cual se le decrementarán sus timers.

Subrutina Decre-Timers

Esta subrutina decremente los timers ubicados en las tablas que no son la tabla Tabla-Timers-BaseT.
detimers

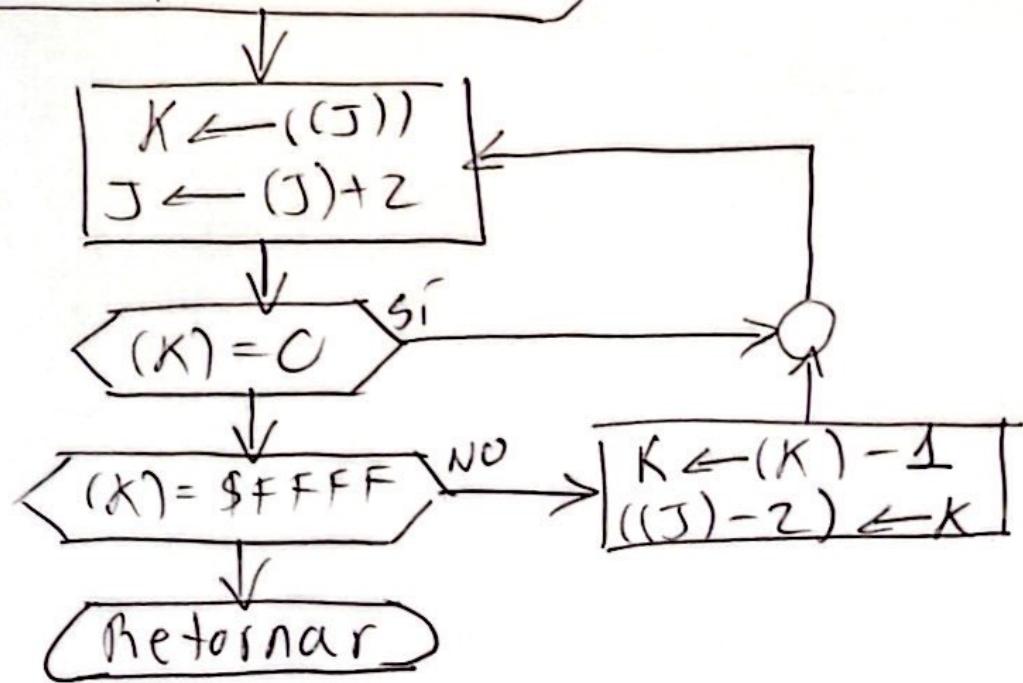
Recibe a través del índice X la dirección base de la tabla a la cual se le decrementarán sus timers.

(Maquina - Tiempo) → BE70



16

Decre-Timers-Baset



Decre-Timers

