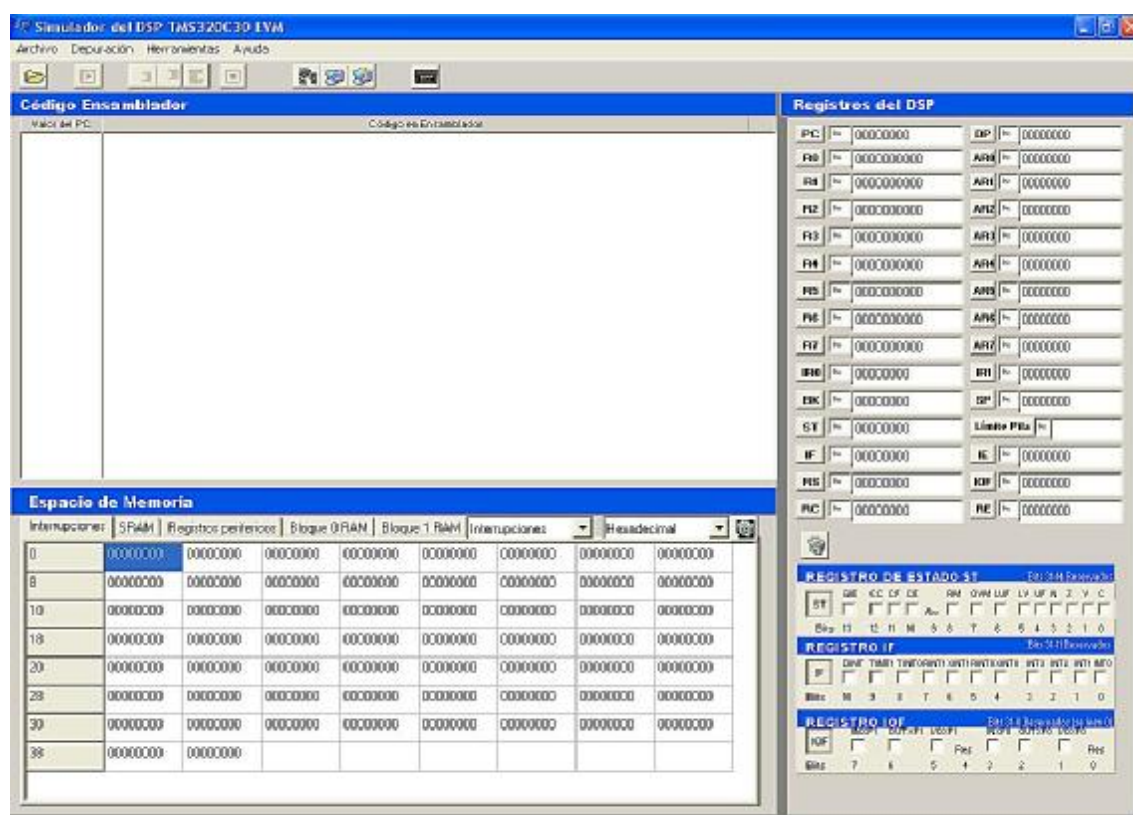


# CAPÍTULO 5: MANUAL DE USUARIO

En este capítulo se explica el modo de funcionamiento del programa, y los pasos a seguir para la correcta utilización del mismo. Para ello se ha ilustrado la explicación con capturas de pantalla del simulador.

## 5.1. INTRODUCCIÓN

Cuando se ejecuta el programa simulador, se visualiza la ventana de trabajo:



**Figura 5.1. Ventana de trabajo del simulador**

Como ya se ha comentado en la sección 3.2., la ventana de trabajo está dividida en tres partes principales:

- **Sección de código:** En ella se puede visualizar el programa ensamblador del DSP una vez que se haya cargado en el programa. Además, también se puede ver la dirección del contador de programa asociada a cada instrucción de ensamblador. Un cursor aparece sombreando la primera instrucción que se debe ejecutar. Si se elige algún modo de ejecución distinto de la ejecución continua,

el cursor va cambiando de posición y se sitúa siempre sobre la siguiente instrucción a ejecutar.

- **Sección de memoria:** En ella aparecen mapeadas las distintas zonas de memoria (Interrupciones, SRAM, registros periféricos, RAM0 y RAM1). Para cada posición de memoria existe la posibilidad de cambiar su valor, y además existe la posibilidad de cambiar el formato numérico de todo el espacio de memoria.
- **Sección de registros:** (ver capítulo 5.3) En esta sección se muestran los valores actuales de los registros. Cada registro tiene la posibilidad de cambiar su valor manualmente, al igual que su formato. En la parte inferior de esta sección aparecen las banderas de los registros ST, IF e IOF. Se pueden cambiar sus valores manualmente en cualquier momento. Por tanto, este cambio es reflejado en los registros.

## 5.2. EL MENÚ PRINCIPAL

El menú principal permite al usuario cargar el programa, manejar los distintos modos de ejecución del mismo, visualizar el menú de herramientas, y por último salir del programa si se desea.

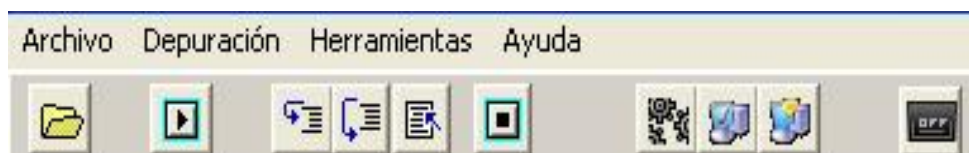





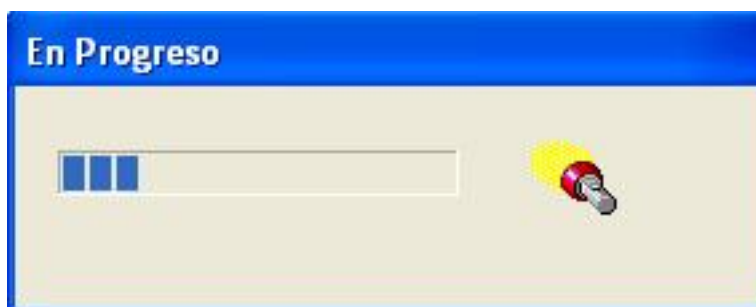
Figura 5.2. El menú principal

Cada submenú tiene una función distinta dentro del menú principal:






- **Menú Archivo:** Permite cargar el programa en formato ensamblador y salir del mismo, según la opción que se elija. En la barra del menú también existen dos botones asociados a estas funciones:
  -  Botón cargar fichero: carga un fichero en formato ensamblador.
  -  Botón salir del programa: la aplicación finaliza.

- Menú Depuración: Permite manejar los distintos modos de ejecución del programa. En la barra de menú también hay cuatro botones asociados a estas funciones, además de los botones del teclado F4, F7, F8 y F9.

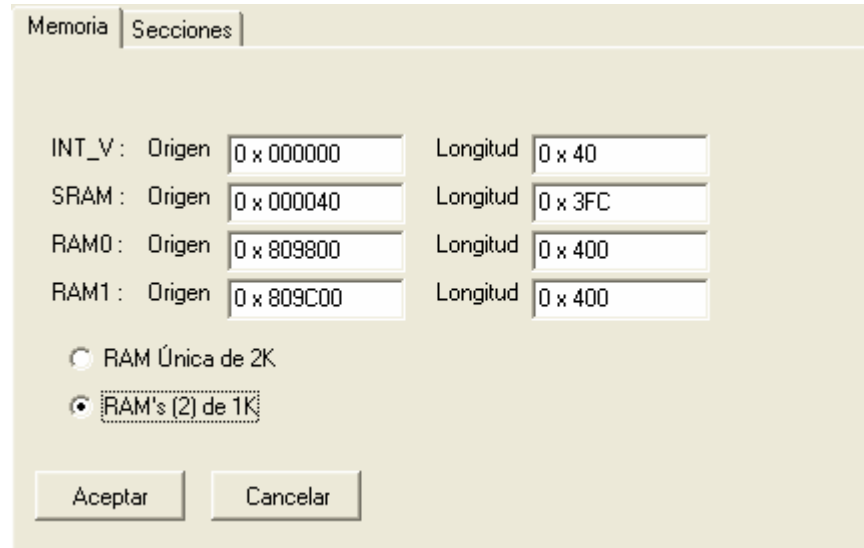
-  Ejecución continua (F9): realiza la simulación de manera continua hasta el final del intervalo temporal elegido. Mientras transcurre la ejecución, se muestra una ventana de progreso.



**Figura 5.3. Ventana de progreso**

-  Ejecución de bloque (F8): ejecuta el programa ensamblador paso a paso excepto los bloques de código que los ejecuta de manera continua.
  -  Ejecución paso a paso (F7): ejecuta el programa ensamblador paso a paso incluso los bloques.
  -  Ejecución hasta el cursor (F4): ejecuta el programa de manera continua hasta donde se sitúe el cursor antes de elegir este modo de ejecución.
  -  Parar la ejecución: detiene la ejecución del programa. El cursor vuelve a la primera instrucción del programa y se cambian los registros y la memoria a los valores iniciales.
- Menú Herramientas: En este menú se pueden ver tres utilidades de simulador. Cada una de ellas tiene también un botón asociado en la barra de herramientas.
    -  Memoria y secciones: esta utilidad muestra las posiciones de cada módulo de memoria y su longitud, así como la ubicación de las secciones de memoria en estos módulos.
      - § Menú de memoria: En esta parte del menú se pueden visualizar los módulos de memoria, la ubicación de su origen y su longitud. También se puede elegir los módulos de RAM0 y RAM1 como

dos módulos de longitud 0x400 (1K palabras) o un solo módulo de longitud 0x800 (2K palabras). Esta elección no afectará al funcionamiento del simulador.



Section	Origen	Longitud
INT_V	0 x 000000	0 x 40
SRAM	0 x 000040	0 x 3FC
RAM0	0 x 809800	0 x 400
RAM1	0 x 809C00	0 x 400

☐ RAM Única de 2K  
☒ RAM's (2) de 1K

Aceptar Cancelar

**Figura 5.4. Ventana de memoria**

§ Menú de secciones: En esta segunda opción del menú se pueden ver las secciones de memoria descritas en el programa y su ubicación en los módulos de memoria. Esta ventana se visualiza en dos momentos distintos de la ejecución del simulador:

- Cuando se carga un programa en formato ensamblador, este menú aparece y se deben ubicar las secciones de memoria en algún módulo de memoria (SRAM, RAM0, o RAM1).

Si la longitud de la pila especificada es mayor que el tamaño físico disponible, el programa nos avisa con un mensaje de error. En tal caso se debe volver a introducir la longitud correcta.

- Una vez pulsado 'Aceptar' y cargado el programa, se realiza la ubicación automática de secciones en memoria. Si se consulta otra vez este menú, es posible ver la posición de cada sección en los módulos y su longitud. Como ya se ha comentado en la sección 3.6, esta ubicación



Memoria Secciones

VectInt INT\_V Comienzo 0x 00000000 Longitud 0x 40

.text > SRAM Comienzo 0x 40 Longitud 0x 0000000F

.stack > RAM1 Comienzo 0x 809C00

Longitud de la Pila: 0x 000003FF

.sect > RAM0


Secciones: buf3

Comienzo 0x 0080981E Longitud 0x 5

ComData> Comienzo 0x 00809823 Longitud 0x B


Aceptar Cancelar

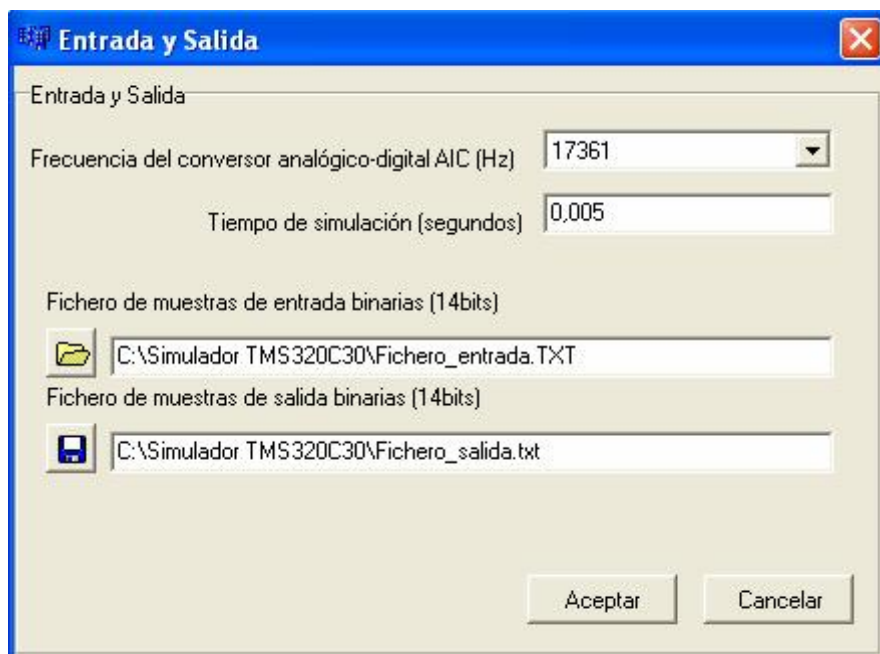
**Figura 5.6. Ventana de secciones después de cargar el programa**

-  **Mapa de memoria de los registros del puerto serie:** En esta ventana se muestran los registros del puerto serie. Para más información consultar las secciones 2.2.9, 2.2.10, y 3.7.

Dirección de Memoria	Valores
808040h	Control global de puerto serie 0.
808042h	FSX/DX/CLKX control del puerto serie 0.
808043h	FSR/DR/CLKR control del puerto serie 0.
808044h	Control del temporizador R/X del puerto serie 0.
808045h	Contador del temporizador R/X del puerto serie 0.
808046h	Periodo del temporizador R/X del puerto serie 0.
808048h	Dato a transmitir del puerto serie 0.
80804Ch	Dato a recibir del puerto serie 0.
808050h	Control global del puerto serie 1.
808052h	FSX/DX/CLKX control del puerto serie 1.
808053h	FSR/DR/CLKR control del puerto serie 1.
808054h	Control del temporizador R/X del puerto serie 1.
808055h	Contador del temporizador R/X del puerto serie 1.
808056h	Periodo del temporizador R/X del puerto serie 1.
808058h	Dato a transmitir del puerto serie 1.
80805Ch	Dato a recibir del puerto serie 1.

**Figura 5.7. Ventana del mapa de memoria de los registros del puerto serie.**

-  **Entrada y salida:** Esta misma pantalla se muestra cuando se carga un programa en ensamblador. En ella se debe especificar:
  - § La frecuencia de muestreo del conversor analógico-digital del AIC (Hz).
  - § Tiempo de la simulación (segundos): Es muy recomendable antes de empezar una ejecución escoger un tiempo de simulación bajo ya que no se puede parar la ejecución continua del simulador. El tiempo de simulación por defecto es de 0,005 segundos.
  - § Fichero de entrada de muestras binarias: se debe elegir un fichero de texto que contenga números enteros con signo de 14 bits, separados entre si por un retorno de carro. Se ha elegido este formato para facilitar la construcción del fichero de texto que contiene las muestras.
  - § Fichero de salida de muestras binarias: si no se elige ningún destino en el cual ubicar el fichero de salida, se crea uno en la ruta: 'C:\Simulador TMS320C30\Fichero\_salida.txt'. El programa genera a la salida números enteros con signo de 14 bits separados entre si mediante espacios. Se ha dado este formato a las muestras de salida para facilitar su uso en el programa matemático 'Matlab'.



**Figura 5.8. Ventana de entrada y salida**



### 5.3. LA SECCIÓN DE REGISTROS

En ella, se muestran los registros del DSP TMS320C30. Además, también se incluye un espacio que muestra el final de la pila del sistema. En la parte inferior se muestran las banderas de los registros ST, IF e IOF. Todo esto se puede ver en la figura 5.8.

**Registros del DSP**

PC	0x	00000040	DP	0x	00000000
R0	0x	0000000000	AR0	0x	00000000
R1	0x	0000000000	AR1	0x	00000000
R2	0x	0000000000	AR2	0x	00000000
R3	0x	0000000000	AR3	0x	00000000
R4	0x	0000000000	AR4	0x	00000000
R5	0x	0000000000	AR5	0x	00000000
R6	0x	0000000000	AR6	0x	00000000
R7	0x	0000000000	AR7	0x	00000000
IR0	0x	00000000	IR1	0x	00000000
BK	0x	00000000	SP	0x	00809C00
ST	0x	00000000	Límite Pila	0x	00809FFF
IF	0x	00000000	IE	0x	00000000
RS	0x	00000000	IOF	0x	00000000
RC	0x	00000000	RE	0x	00000000

**REGISTRO DE ESTADO ST** Bits 31-14 Reservados

ST	GIE	CC	CF	CE	RM	OVM	LUF	LV	UF	N	Z	V	C
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bits	13	12	11	10	9	8	7	6	5	4	3	2	1

**REGISTRO IF** Bits 31-11 Reservados

IF	DINT	TINT1	TINT0	RINT1	XINT1	RINT0	XINT0	INT3	INT2	INT1	INT0
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bits	10	9	8	7	6	5	4	3	2	1	0

**REGISTRO IOF** Bits 31-8 Reservados (se leen 0)

IOF	INXF1	OUTXF1	IOXF1	Res	INXF0	OUTXF0	IOXF0	Res
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bits	7	6	5	4	3	2	1	0

Figura 5.9. Ventana de registros



Cada ventana de registro está expresada inicialmente en valor hexadecimal. Acompañando a cada ventana de registro existe una pequeña ventana que indica el formato del valor expresado en el registro. Los símbolos mostrados en esta ventana significan lo siguiente:


- “0x”: El valor del registro está expresado en valor hexadecimal.
- “BIN”: El valor del registro está expresado en formato binario.
- “ECS”: El valor del registro está expresado en formato entero con signo.
- “FLT”: El valor del registro está expresado en formato punto flotante.
- “NC”: El valor del registro está expresado en formato entero con signo pero con notación científica.

Para cambiar el formato de visualización de cada registro, basta con pulsar el botón derecho del ratón con el cursor encima de la ventana de registro deseada. El menú que aparece es el mostrado en la figura 5.10.

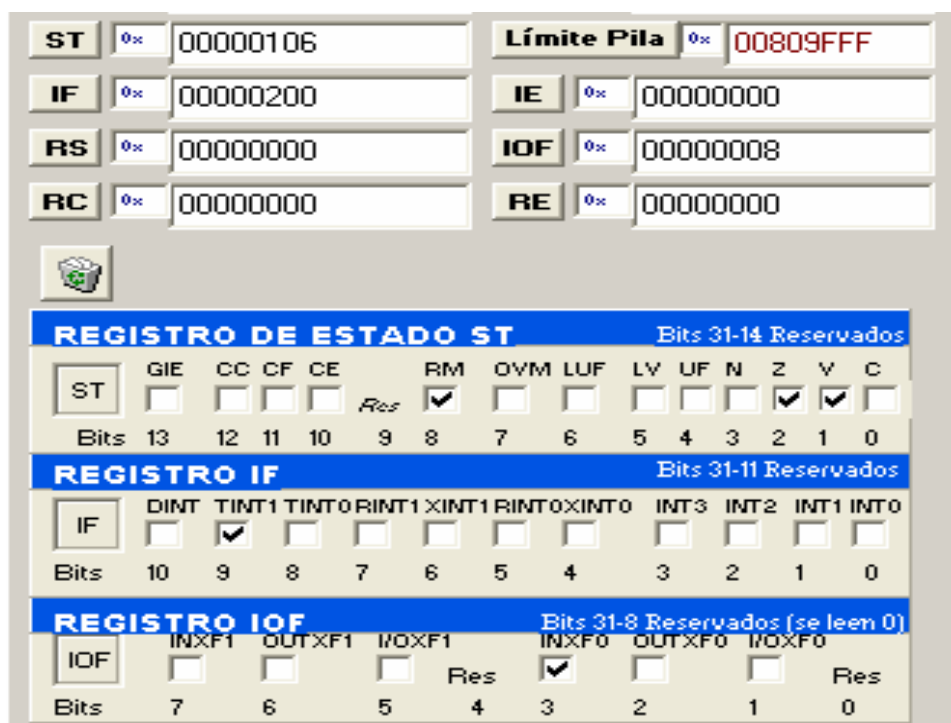


**Figura 5.10. Menú de cambio de formato del valor del registro.**

En el menú aparece marcado y sombreado el formato actual del registro, pudiendo elegir cualquier otro. Existe una restricción: si el formato del registro es de punto flotante no se puede cambiar a valor entero con signo y viceversa.

En cualquier momento de la ejecución del programa se puede limpiar el contenido de los registros mediante el botón inferior situado a la izquierda de la ventana: .

Cualquier cambio en los registros ST, IF e IOF es reflejado en las banderas de cada uno situadas en la parte inferior de la ventana. Un ejemplo de esto se muestra en la figura 5.11.



La interfaz de usuario muestra los registros ST, IF e IOF y sus banderas correspondientes. Los registros ST, IF e IOF tienen valores hexadecimales de 00000106, 00000200 y 00000000 respectivamente. El límite de la pila (Límite Pila) está configurado en 00809FFF. Las banderas de los registros ST, IF e IOF se muestran en la parte inferior de la ventana.

REGISTRO DE ESTADO ST													
ST	GIE	CC	CF	CE	RM	OVM	LUF	LV	UF	N	Z	V	C
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Bits	13	12	11	10	9	8	7	6	5	4	3	2	1

REGISTRO IF											
IF	DINT	TINT1	TINT0	RINT1	XINT1	RINT0	XINT0	INT3	INT2	INT1	INT0
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bits	10	9	8	7	6	5	4	3	2	1	0

REGISTRO IOF							
IOF	INXF1	OUTXF1	I/OXF1	Res	INXF0	OUTXF0	I/OXF0
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bits	7	6	5	4	3	2	1

Figura 5.11. Banderas de los registros ST, IF e IOF

## 5.4. LA SECCIÓN DE MEMORIA

En esta sección se muestran las partes del mapa de memoria del TMS320C30 EVM que se han considerado necesarias para el desarrollo del simulador. Estas partes son:

- Vectores de interrupción: desde la posición 0x0 a la 0x39 de memoria.
- SRAM: desde la posición 0x40 a la 0x3FF.
- Registros periféricos: desde la posición 0x808000 a la 0x80806F.
- RAM0: desde la posición 0x809800 a la 0x809BFF.
- RAM1: desde la posición 0x809C00 a la 0x809FFF.

Cada parte está dividida en palabras de 32bits, que forman una cuadrícula de memoria para cada sección, pudiendo visualizarse cada una por separado.

Espacio de Memoria								
Interrupciones	SRAM	Registros perifericos	Bloque 0 RAM	Bloque 1 RAM	Interrupciones	Hexadecimal		
0	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
8	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
10	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
18	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
20	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
28	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
30	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
38	00000000	00000000						

Figura 5.12. Sección de memoria

En la parte superior derecha se puede elegir la sección de memoria a la que se quiera acceder. Además, se puede acceder a una posición concreta de memoria escribiéndola en la ventana y pulsando 'Enter'. La posición de memoria buscada queda marcada con color azul.

Espacio de Memoria								
Interrupciones	SRAM	Registros perifericos	Bloque 0 RAM	Bloque 1 RAM	80981B	Hexadecimal		
809800	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
809808	00000001	00000004	00000000	00000001	BF800000	3FF5BE4D	C047295F	
809810	4025CE07	00808040	00000000	00000000	00000001	00000001	00000000	
809818	00000004	00000001	BF800000	3FF5BE4D	C047295F	4025CE07	00000000	
809820	00000000	00000000	00000000	00000000	00000000	00000000	00000000	
809828	00000000	00000000	00000000	00000000	00000000	00000000	00000000	
809830	00000000	00000000	00000000	00000000	00000000	00000000	00000000	
809838	00000000	00000000	00000000	00000000	00000000	00000000	00000000	
809840	00000000	00000000	00000000	00000000	00000000	00000000	00000000	

Figura 5.13. Acceso a una posición de memoria.

A continuación, se encuentra otro control desplegable mediante el cual se puede cambiar el formato de los valores de la memoria. En este caso, cuando se cambia el control de selección de formato, se cambia el formato de todas las posiciones de memoria.

En el extremo superior derecho se encuentra un botón de limpieza de la memoria. Este botón puede utilizarse en cualquier momento de la ejecución, siempre que no se utilice el modo continuo.

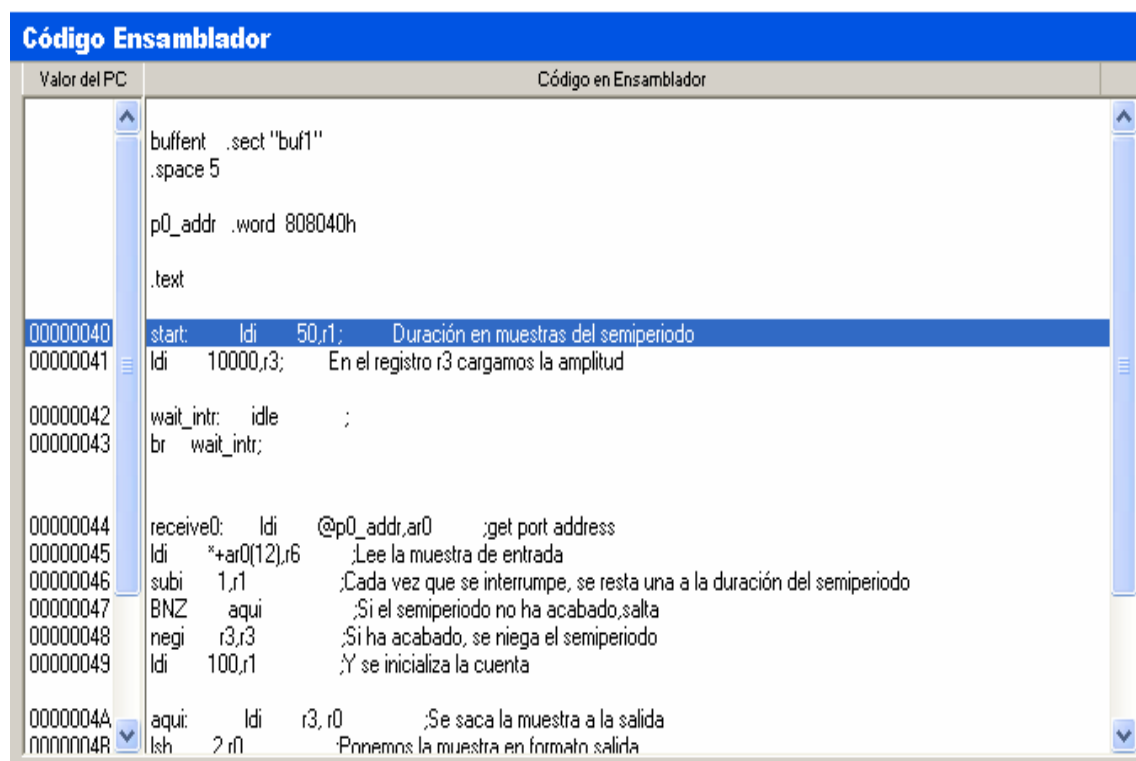


**Figura 5.14.**

Si se desea cambiar el valor de una posición de memoria, se debe hacer un doble clic del ratón sobre la posición de memoria que se quiera cambiar. Como consecuencia de esto aparece una ventana donde podremos introducir el nuevo valor.

## 5.5. LA VENTANA DE EJECUCIÓN

En la ventana de ejecución se muestra el programa ensamblador del DSP que se ha cargado previamente. A la izquierda de las instrucciones aparece el valor del contador de programa asociado a cada instrucción. Inicialmente, el cursor de ejecución aparece sobre la primera instrucción del programa ensamblador.



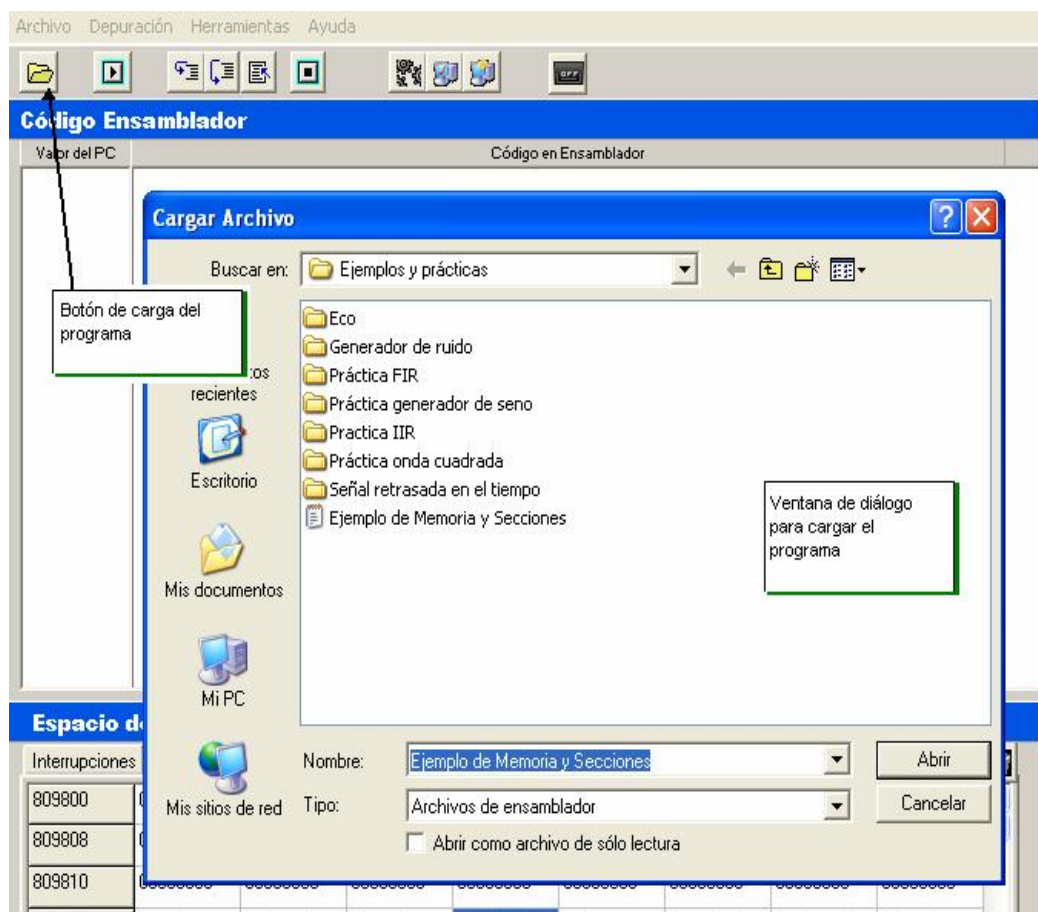
**Figura 5.15. Ventana de ejecución**

En todos los modos de ejecución excepto el continuo, el cursor cambia de posición a medida que se ejecutan instrucciones.

## 5.6. CARGA Y EJECUCIÓN DE UN PROGRAMA ENSAMBLADOR

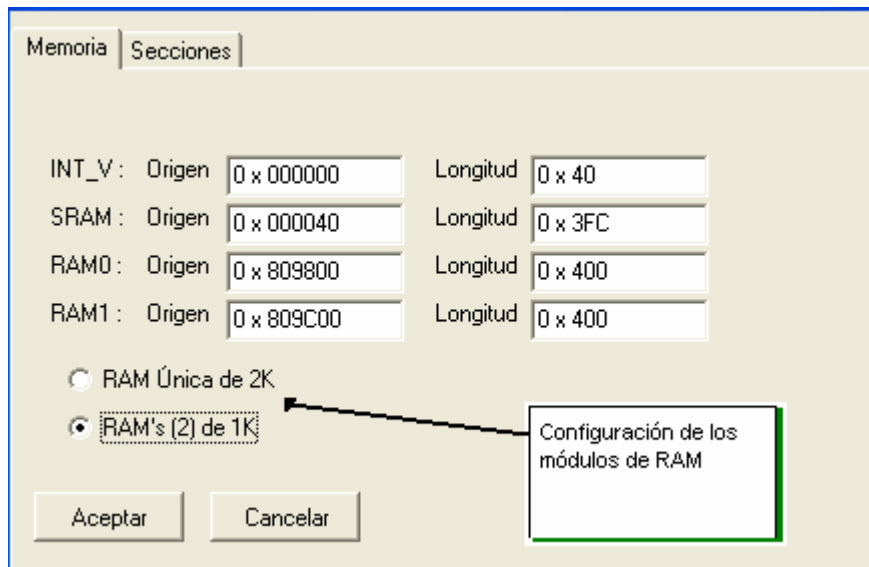
Los pasos a seguir desde que arrancamos el programa hasta que empieza la ejecución del mismo son los siguientes:

- Carga del programa ensamblador: Para cargar un programa ensamblador del DSP, debemos pulsar el botón de carga del programa (Figura 5.16) o el elegir 'Archivo/Abrir...' del menú principal.

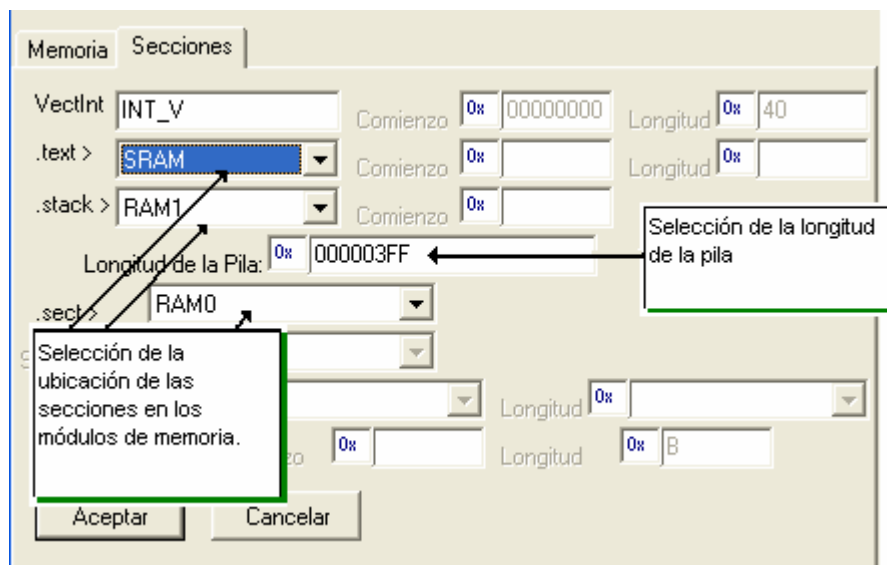


**Figura 5.16. Ventana de diálogo de carga**

- Ubicación de las secciones en los módulos de memoria: Una vez seleccionado un programa ensamblador, se pide introducir los datos correspondientes a la memoria y secciones. (Ver sección 3.6 para más detalles). En las figuras 5.16 y 5.17 se muestran las dos pantallas correspondientes a la ventana de memoria y secciones.

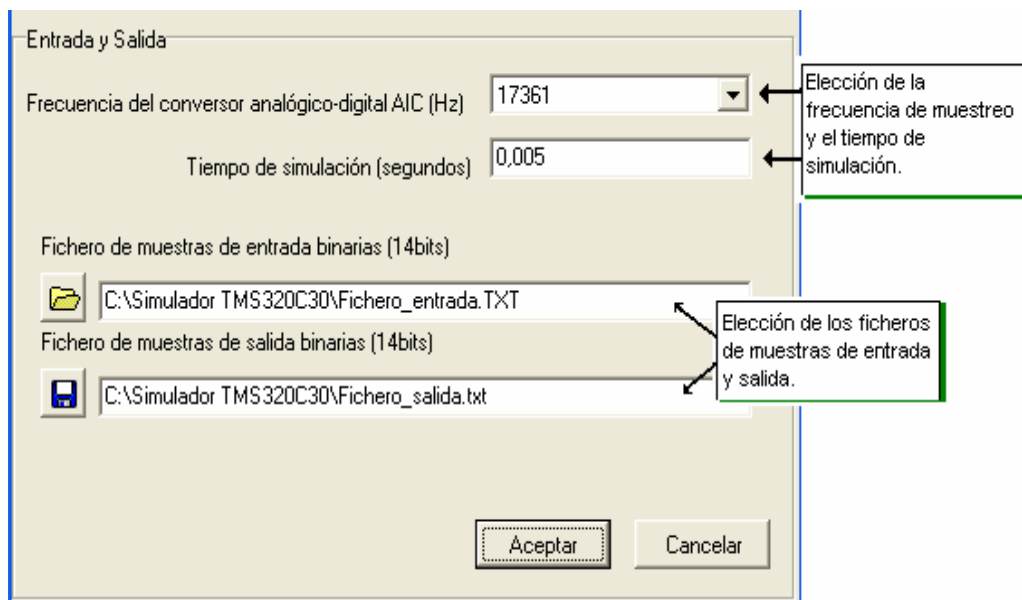


**Figura 5.17. Pantalla de memoria de la ventana 'Memoria y secciones'**



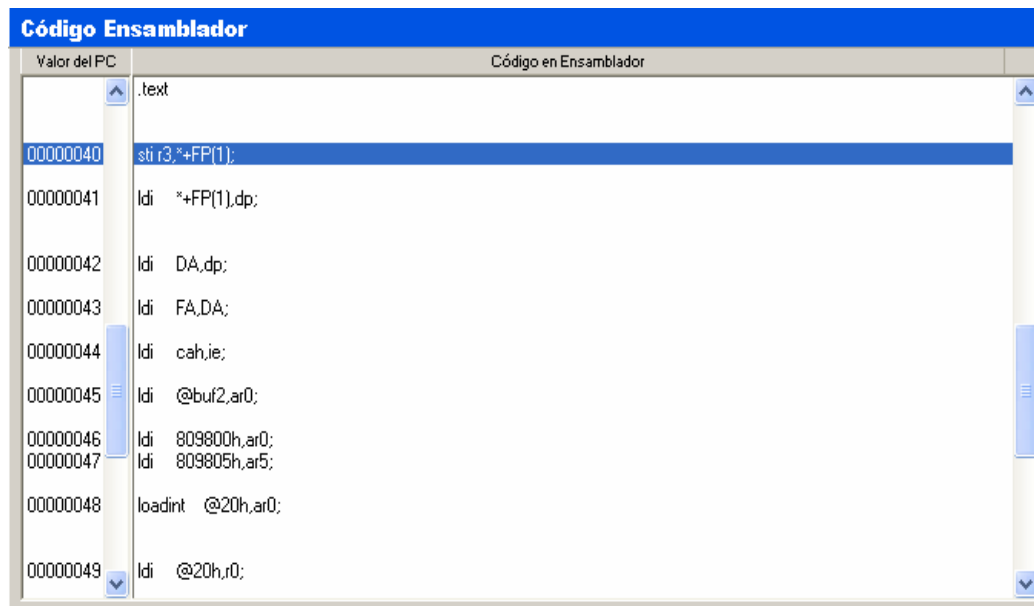
**Figura 5.18. Pantalla de secciones de la ventana 'Memoria y secciones'**

- Elección de la frecuencia de muestreo del ADC, el tiempo de ejecución y de los ficheros de muestras de entrada y salida.



**Figura 5.19. Ventana de ‘Entrada y salida’**

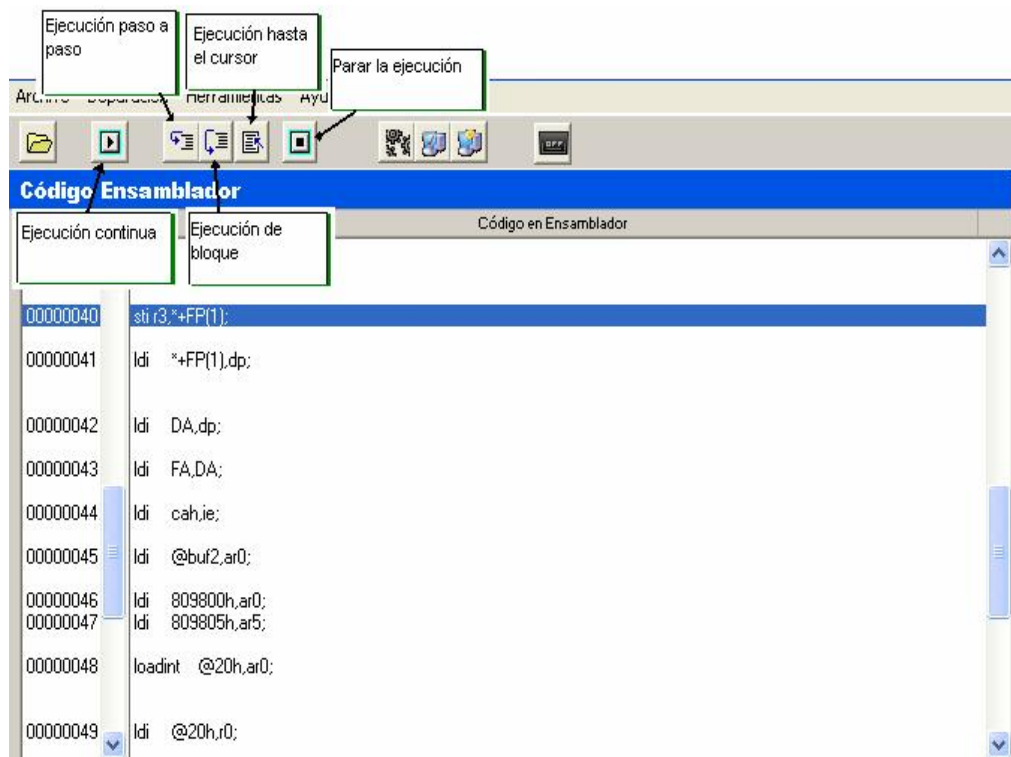
- Una vez cargado el programa, se realiza automáticamente la numeración de las instrucciones.



**Figura 5.20. Ventana de ejecución. El contador de programa está asignado para cada instrucción.**

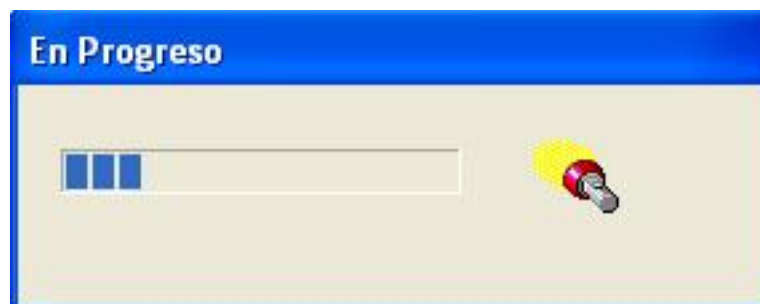
- Elección del modo de ejecución: cuando ya se ha asignado automáticamente el contador de programa a cada una de las instrucciones, se puede empezar la ejecución del programa.



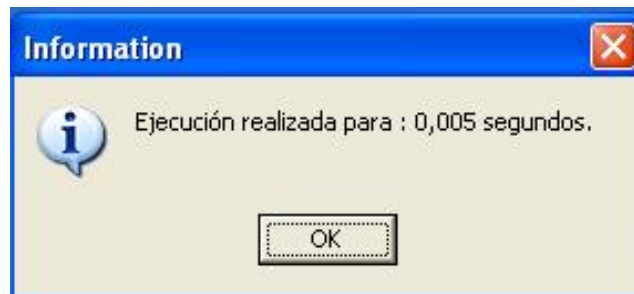


**Figura 5.21. Ventana de ejecución. El contador de programa está asignado para cada instrucción**

- En el modo de ejecución continua, se observa un mensaje advirtiendo que una vez se comience la ejecución, no podrá ser interrumpida. Mientras se realiza la ejecución, aparece un mensaje con una barra de progreso. Esto se muestra en la figura 5.22. Una vez acabada la ejecución, se muestra un mensaje en la pantalla comunicando el fin de la ejecución (figura 5.23).

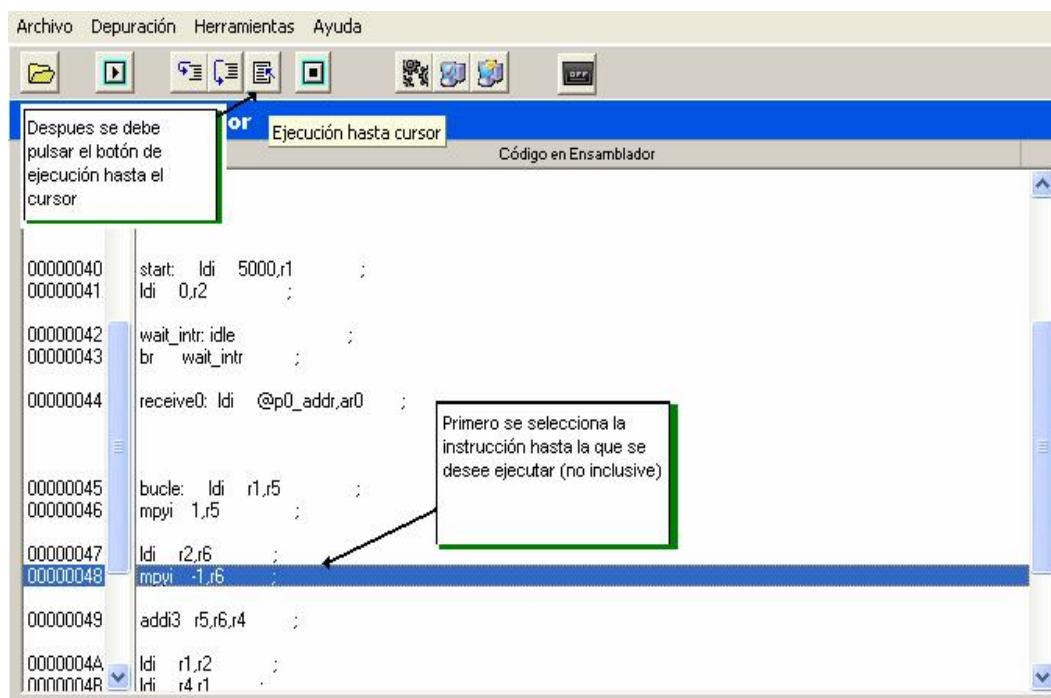


**Figura 5.22. Ventana de progreso**



**Figura 5.23. Mensaje de fin de la ejecución**

- Los modos de ejecución paso a paso y de bloque son similares. Se pueden consultar y modificar los valores de los registros y memoria en cada paso.
- Para realizar el modo de ejecución hasta el cursor basta con seleccionar con el ratón la instrucción hasta donde se desee ejecutar el programa. Seguidamente se debe pulsar el botón de ejecución hasta cursor.



**Figura 5.24. Ejecución hasta el cursor.**

- Si en algún momento se desea parar la ejecución en cursor, solo hay que pulsar el botón de parada de ejecución. Esto hace que se limpie la memoria y los registros y se comience la ejecución desde la primera línea de código. En el caso de tener variables inicializadas, se debe cargar nuevamente el programa.
- Para salir del programa, existe un botón de salida en la barra de herramientas que muestra la figura 5.25.



**Figura 5.25. Botón de salida del programa.**

## **5.7. RECOMENDACIONES**

- Deben utilizarse estrictamente las instrucciones enumeradas en la sección 3.5. y no utilizar el tipo de direccionamiento a bit invertido, que no ha sido implementado.
- Evitar el uso de directivas de ensamblador no enumeradas en la sección 3.6.
- Para realizar pequeñas pruebas con el modo de ejecución continua, es conveniente elegir un tiempo de ejecución bajo, ya que una vez comenzada la ejecución, no se puede detener.
- Cuando se haga uso del *buffer* circular, el valor del registro BK debe ser un valor distinto de cero. Si es cero, el programa mostrará un mensaje de error.
- Evitar que en la misma línea donde exista una etiqueta de bloque, no haya una instrucción en esa misma línea.