

Simulador 'SIMPLEZ+i4'



Gustavo Moreno – Ricardo Carvajal
Técnicas Digitales II

Universidad Tecnológica Nacional
Facultad Regional Mendoza

Abril 2000

Introducción

El *Simulador 'SIMPLEZ+i4'* es un entorno de simulación sencillo y práctico que le permitirá ejercitar el desarrollo y ejecución de programas para la computadora teórica 'SIMPLEZ+i4', y además, le permitirá estar en contacto con el funcionamiento de su hardware facilitando su comprensión.

El objetivo de este manual es explicar paso a paso el uso correcto del Simulador. Para ello se aconseja un conocimiento previo de la arquitectura de SIMPLEZ+i4 y de su modelo funcional (Ver Apéndice). Si deseara profundizar sobre el tema puede hallarlo *en Conceptos Básicos de Arquitectura de Computadoras* de Gregorio Fernández.

Respecto a los convenios simbólicos del lenguaje ensamblador de la SIMPLEZ+i4 establecidos en el libro antes mencionado se han producido cambios mínimos que se detallan en el Apéndice.

El manual consiste en la explicación de los menús de la ventana principal del programa Simulador SIMPLEZ+i4 y como trabajar con cada uno de ellos.

Iniciando su trabajo con el Simulador 'SIMPLEZ+i4'

Para correr la aplicación ejecute el archivo SIMPLZi4.EXE.

Al iniciar su tarea con el *Simulador SIMPLEZ+i4* aparecerá la ventana principal, titulada SIMPLEZ+i4, y que constituye el administrador del entorno de trabajo.

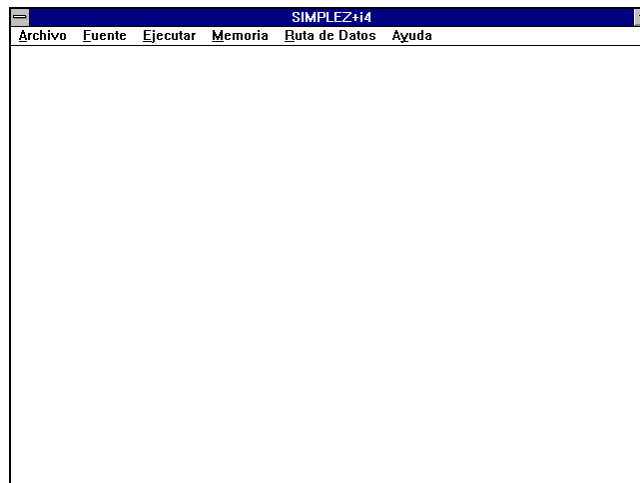


Fig. 1 - Ventana Principal

El menú principal está formado por 6 menús que son:

- Archivo
- Fuente
- Ejecutar
- Memoria
- Ruta de Datos
- Ayuda

Utilice el menú principal para acceder a las distintas tareas que puede realizar con el Simulador.

Al comenzar su tarea notará que en el menú principal sólo aparecen habilitados los menús Archivo y Ayuda. Esto es parte del modo de trabajo del Simulador, puesto que, para que usted pueda acceder a los restantes menús del Simulador debe previamente crear o abrir un archivo que contenga el programa que luego se cargará en la memoria principal de la SIMPLEZ+i4. Mientras no haya un programa cargado en la memoria, los menús Fuente, Ejecutar, Memoria y Ruta de Datos no serán habilitados.

Descripción de los Menús

1 - Menú Archivo



Fig. 2 - Menú Archivo

Utilice el menú Archivo para crear, abrir, guardar y cerrar un archivo de programa.

Un archivo de programa tiene un nombre, que está dado por una cadena de ocho caracteres; y una extensión, que por definición es <nombre> .SZ4 si el archivo contiene un programa fuente (programa en lenguaje ensamblador), y si el archivo contiene un programa objeto (programa en lenguaje de máquina) la extensión será <nombre>.OB4.

1.1 - Comando **A**rchivo | **N**uevo

Use este comando cuando desee escribir un programa nuevo en lenguaje ensamblador.

Este comando abre el Editor de Código Fuente, el cual le facilita un medio para la escritura del programa , aunque también podría utilizar cualquier otro editor de texto.

En la ventana del editor encontrará los siguientes submenús:

Editar

Le permite copiar, cortar, pegar y borrar el texto seleccionado. El tipo de carácter está definido por defecto y no podrá ser editado.

Imprimir

Le permite imprimir el texto que aparece en el editor.

Ensamblar

Es el comando más importante. Este comando ejecuta el programa ensamblador, el cual, primero examina el programa fuente y, si es correcto, lo traduce a lenguaje de máquina y, si no, indica sus errores mediante un mensaje.

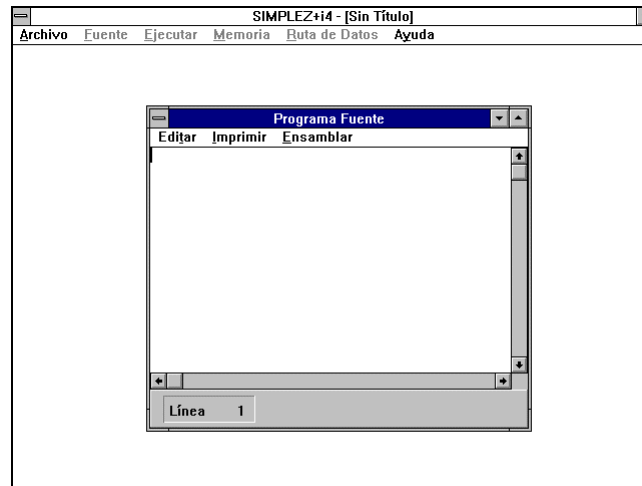


Fig. 3 - Ventana Editor de Código Fuente

Una vez que usted haya escrito su programa fuente, éste debe ser ensamblado. El comando **Ensamblar** de la ventana del Editor ejecuta el programa ensamblador, el cual, traduce el programa de lenguaje ensamblador a lenguaje de máquina, y si no existen errores lo carga en la memoria principal, informando que el programa ha sido ensamblado con éxito.

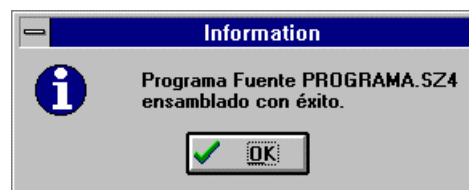


Fig. 4 - Mensaje de programa fuente ensamblado con éxito

Si el programa ensamblador halla un error emite un mensaje de error, indicando el posible error y el número de línea en el que se ha producido.

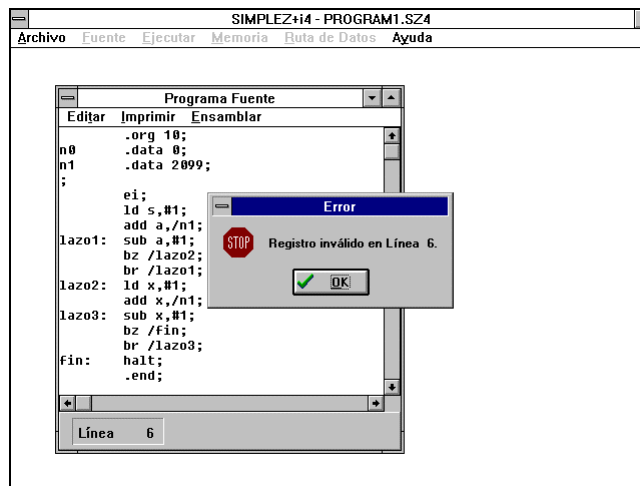


Fig. 5 - Mensaje de error en el proceso de ensamblado

Utilice el cuadro guía que aparece en la parte inferior izquierda de la ventana del editor para ubicar la línea que contiene el error. El proceso de depuración de errores será repetitivo, es decir que, se debe ensamblar el programa hasta que quede libre de errores.

Luego de que el programa ha sido ensamblado con éxito, se habilita el acceso a las distintas tareas del Simulador.

1.2 - Comando Archivo | Abrir

Use este comando para abrir un archivo de programa fuente o un archivo de programa objeto.

Estas dos opciones aparecen en un cuadro de diálogo cuando se ejecuta el comando.



Fig. 6 - Cuadro de diálogo Abrir

Si elige Abrir | Archivo Fuente a continuación aparece



Fig. 7 - Abrir - Archivo Fuente

Seleccione el nombre del archivo que se desea abrir y escoja **Aceptar** del cuadro de diálogo.

En este caso, se abrirá la ventana del Editor de Código Fuente con el contenido del programa fuente como se muestra en la figura siguiente

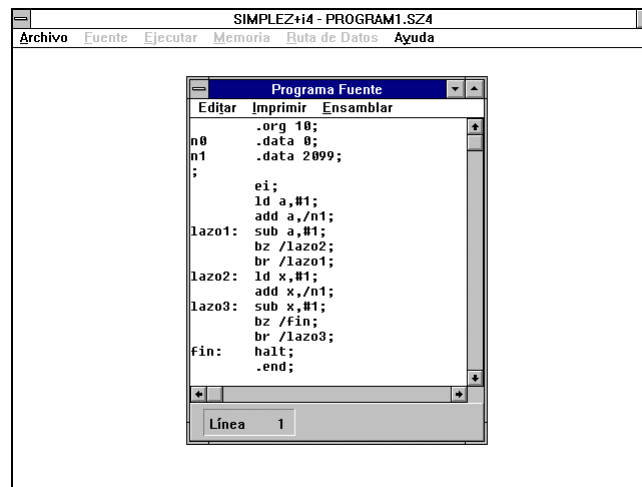


Fig. 8 - Programa Fuente cargado en el Editor

El nombre del archivo abierto aparece en la barra de título de la ventana principal junto al título de la misma.

El programa fuente que usted ha cargado quizás ya había sido ensamblado con éxito antes de ser guardado, aún así, usted debe volver a ensamblarlo para poder continuar con la modo de trabajo explicado anteriormente en el tema comando Archivo | Nuevo.

Si usted elige Abrir | Archivo Objeto aparece un cuadro de diálogo similar al de Abrir - Archivo Fuente, solo que cambia la extensión del archivo.



Fig. 9 - Abrir - Archivo Objeto

Seleccione el nombre del archivo que desea abrir y elija Aceptar. Recuerde que el archivo objeto consiste en código máquina, por lo tanto, al abrirlo su contenido se carga automáticamente en la Memoria Principal de la SIMPLEZ+i4, dicha acción es informada mediante el siguiente mensaje

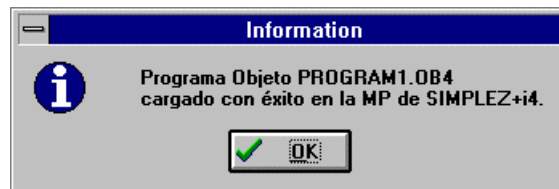


Fig. 10 - Mensaje de programa objeto ensamblado con éxito

1.3 - Comando Archivo | Guardar

Use este comando para guardar en una unidad de almacenamiento permanente, como es el disco rígido, el archivo de programa fuente u objeto con el que está trabajando.

Si usted no ha abierto ningún archivo este comando permanece inhabilitado.

1.4 - Comando Archivo | Guardar como

Use este comando para guardar con un nombre y/o ruta distinta el archivo de programa fuente u objeto con el que está trabajando.

Si usted no ha abierto ningún archivo este comando permanece inhabilitado.

1.5 - Comando Archivo | Cerrar

Este comando cierra el archivo con el que está trabajando, es decir si usted quiere crear un nuevo archivo o abrir uno distinto debe cerrar el actual. Tenga en cuenta que los datos tanto en la

memoria como en los registros no serán borrados al cerrarse todas las ventanas de trabajo.

Si usted no ha abierto ningún archivo este comando permanece inhabilitado.

1.6 - Comando Archivo | Salir

Este comando le permite finalizar su trabajo con el Simulador 'SIMPLEZ+i4'.

NOTA: este comando no brinda ninguna opción de salida, por lo que si el archivo fue modificado y se desea guardar dicha modificación, se deberá ejecutar el comando Archivo | Guardar o Guardar como... previamente al ejecutar Salir.

2 - Menú Fuente

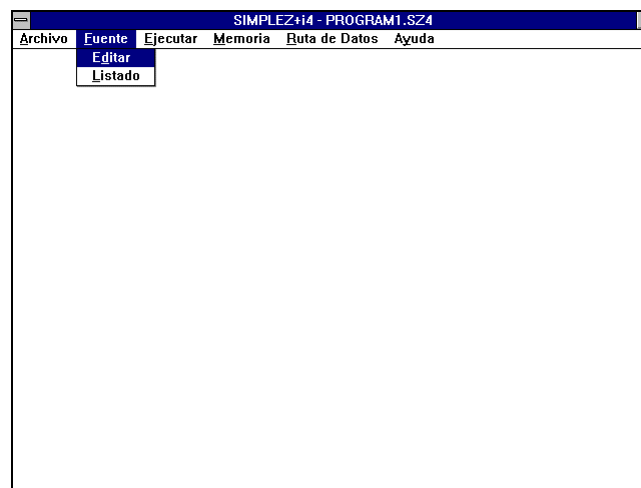


Fig. 11 - Menú Fuente

2.1 - Comando Fuente | Edición

Use este comando para poder reeditar el programa fuente actual. Recuerde que si produce cambios en un programa fuente debe ensamblarlo, para que esos cambios sean tomados.

2.2 - Comando Fuente | Listado

Este comando le permite obtener un listado de Ensamblador que consiste en una lista enumerada con los siguientes campos: número de línea de programa fuente, dirección de memoria en decimal, contenido de la memoria en octal, y programa fuente

sin errores. Una vez generado este listado se muestra en una ventana como la siguiente

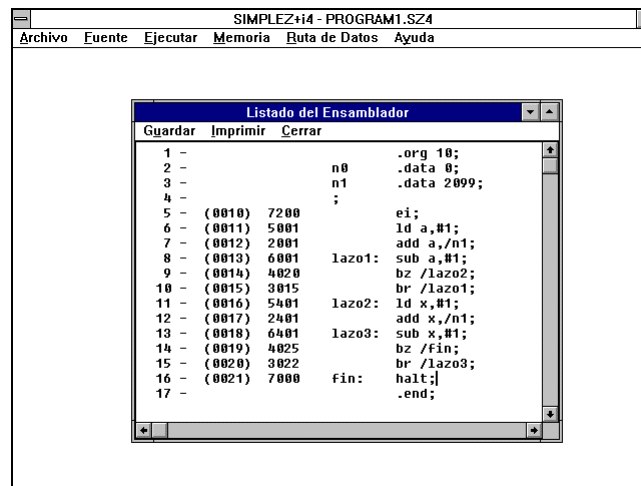


Fig. 12 - Ventana Listado de Ensamblador

El listado puede ser guardado como un archivo de texto con la extensión <nombre>.LST mediante el comando Guardar del ventana del listado.

El listado también se puede imprimir, lo que se logra al ejecutar el comando Imprimir de la ventana del listado.

3 - Menú Ejecutar

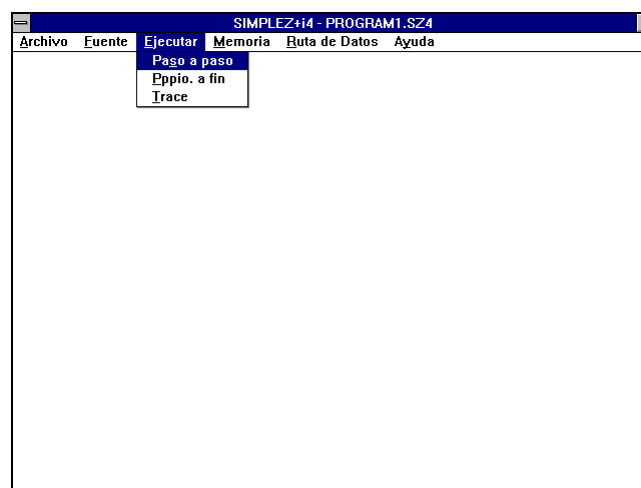


Fig. 13 - Menú Ejecutar

El menú **Ejecutar** le posibilita la ejecución de un programa cargado en memoria en distintas modalidades:

- Paso a paso
- De principio a fin

- Trace

3.1 - Comando Ejecutar | Paso a paso

En esta modalidad el programa se ejecuta instrucción por instrucción.



Fig. 14 - Ventana Registros (Ejecución Paso a Paso)

Para ejecutar cada instrucción del programa indicada por el Registro de Instrucciones(RI) presione el botón comando **prox. paso**. Si la instrucción en RI es un 'HALT', una vez ejecutada, aparecerá el mensaje '*proceso detenido*', indicándole que la ejecución del programa llegó a su fin.

Durante la ejecución se puede ver el contenido de todos los Registros.

El contenido del Registro de Instrucciones (RI) está representado en binario mostrando el código de máquina que ingresa a la Unidad de Control. Para facilitar su interpretación, sobre el cuadro del contenido se especifica el mnemónico correspondiente a dicho código.

Los contenidos de los registros: Contador de Programa (CP), Acumulador (A) y Registro Puntero (X), pueden verse en decimal o en octal (está opción aparece en el menú de la ventana Registros como: **decimal** y **octal**). Además, estos contenidos pueden ser modificados. Para ello, teclee el nuevo valor en el cuadro del contenido del registro que desee modificar. En el momento de modificar el valor del contenido notará que la etiqueta con el nombre del registro cuyo contenido está

modificando, cambia su color, y que aparece el mensaje '*Para ingresar presione mod. reg.*', esto le está indicando que el valor actual será ingresado realmente sólo si presiona, luego de modificar el valor, el comando **mod. reg** del menú de la ventana Registros. Si el valor ingresado es incorrecto aparecerá un mensaje de error. También existe la posibilidad de poner a cero el contenido de todos estos registros mediante el comando **reset** del menú de la ventana Registros.

Los contenidos de los registros: Registro de Estado de Salida de la ALU (Z) y Registro de Habilitación de Interrupciones (EI), consisten en un bit de información y por lo tanto lo que usted ve es un 0, o un 1, según corresponda.

Los contenidos de los registros: Registro de Entrada de Datos (TECL) y Registro de Salida de Datos (PANT), se representan en código ASCII (por ej. si el contenido es la letra A, el valor que verá será el número 65).

3.2 - Comando **Ejecutar | Ppio. a fin**

El programa se ejecuta en forma completa a partir de la dirección de comienzo especificada, mostrando el contenido final de todos los registros.

Antes de realizar la ejecución se requiere la dirección de comienzo de ejecución.

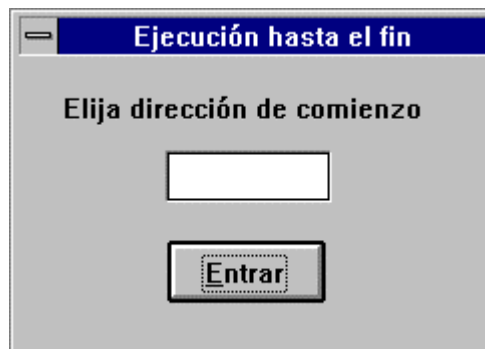


Fig. 15 - Elección de la dirección de comienzo

Esto es así, por que si está trabajando a partir de un programa objeto, el programa ejecutor tomará como dirección de comienzo por defecto, la dirección 0, y quizás el programa que desea ejecutar está cargado a partir de una determinada dirección que no es la dirección 0. El programador debe conocer dicha dirección, de lo contrario la ejecución del programa puede ser errónea.

Si se está trabajando a partir de un programa fuente, la dirección de inicio será también por defecto la dirección 0, si no, será la dirección especificada por la directiva **.ORG**.

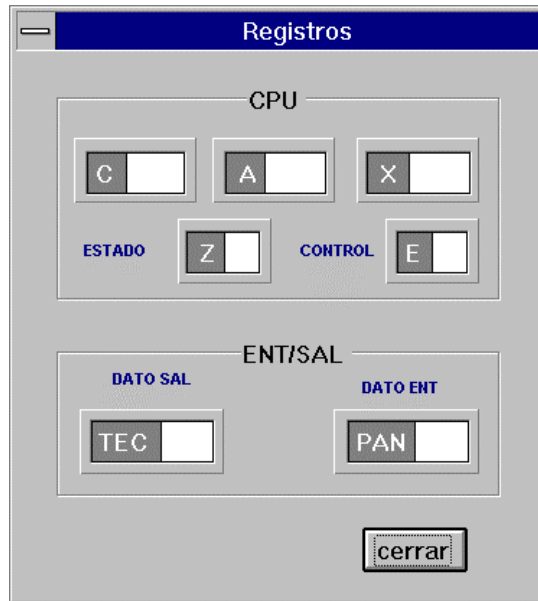


Fig. 16 - Ventana Registros (Ejecución Ppio. a Fin)

Durante la ejecución del programa al pulsar una tecla del teclado de la PC se provoca una interrupción por teclado. Si $EI = 1$ (interrupción habilitada) la ejecución salta automáticamente a la dirección 4000, guardando previamente el contenido del CP en la dirección 63.

Por lo tanto, el programador debe cargar a partir de la dirección 4000 su rutina de atención de interrupción, de lo contrario la ejecución del programa será errónea.

Al producirse la interrupción por teclado el código ASCII de la tecla pulsada se almacena en el Registro de Entrada de Datos. El programador puede entonces tomar dicho dato para procesarlo, o bien, puede enviarlo directamente al Registro de Salida de Datos.

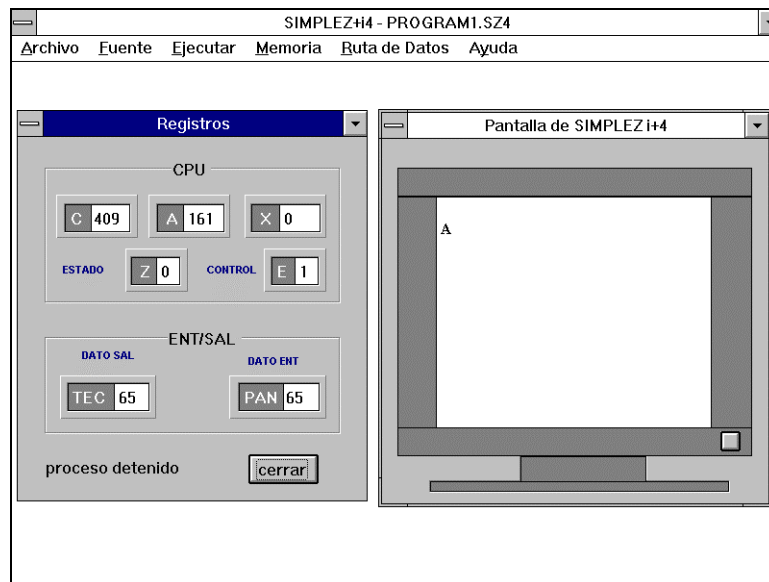


Fig. 17 - Ventana posterior a realizar una interrupción

Si esto ocurre, dicho dato aparece en la pantalla virtual de SIMPLÉZ+i4, cuya ventana se activa en el instante en que el dato es almacenado en el registro correspondiente.

Las teclas que responderán al ser pulsadas son: barra espaciadora, retroceso(backspace), retorno de carro(return), todas las letras y números (mayúsculas y minúsculas).

Se aconseja al programador incluir, entre las líneas de su programa, un retardo, de manera que el tiempo de ejecución del programa sea el suficiente para alcanzar a provocar la interrupción por teclado.

3.3 - Comando Ejecutar | Trace

En esta modalidad el programa se ejecuta instrucción por instrucción una cantidad determinada de pasos (valor máximo 255) a partir de una dirección específica, y luego se muestra el trazado de las instrucciones ejecutadas, donde en cada línea se indica el contenido de todos los registros junto al mnemónico de la instrucción.

Veamos un ejemplo

- Ingrese al menú Ejecutar.
- Elija Ejecutar | Trace.
- En el cuadro de dialogo que aparece especifique los siguientes parámetros

dirección de comienzo = 0

cantidad de pasos = 20

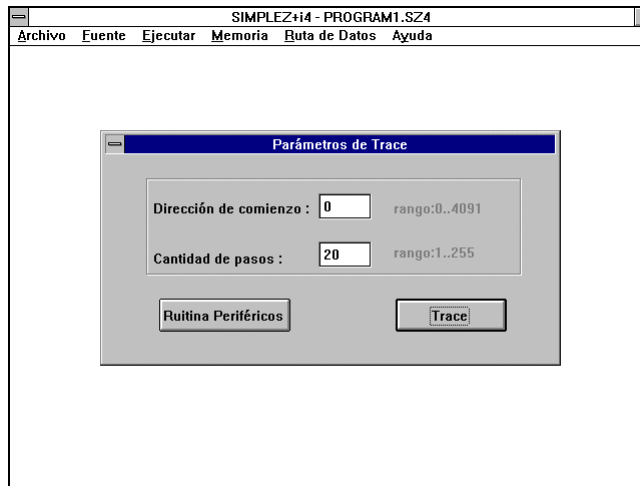


Fig. 18 - Elección de los parámetros Trace

Presione el botón comando Trace. El programa ejecutor toma el valor de los parámetros y una vez que ejecuta la cantidad de pasos indicada muestra el trazado en la ventana Trace como sigue

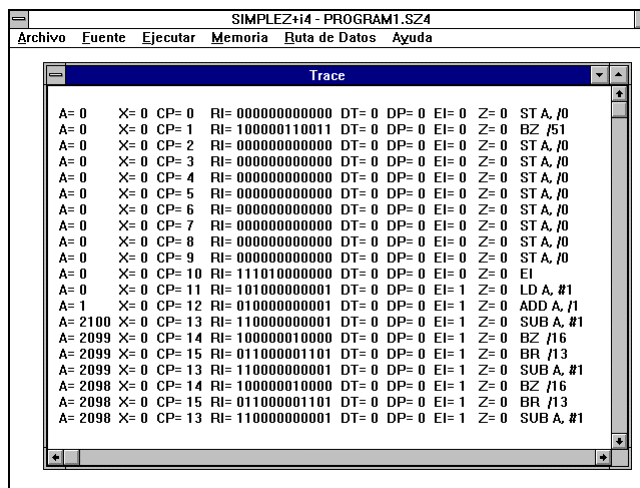


Fig. 19 - Ventana Trace luego de la ejecución

Si dentro del bloque de instrucciones a ejecutar se encuentra la instrucción 'HALT' el programa finalizará el trazado en esa línea aunque el número de pasos especificado sea mayor al ejecutado. Por ejemplo, si se especifica iniciar la ejecución desde una cierta dirección una cantidad de pasos igual a 20, si en el paso 10 aparece un 'HALT', en la ventana Trace veremos diez líneas.

Utilice el botón comando Rutina Periféricos para ver qué sucede cuando se provoca una interrupción por teclado, es decir, una vez que usted presiona Rutina Periféricos pulse inmediatamente una tecla, como resultado de esta acción en la ventana Trace se

visualizará el salto a la dirección 4000 que la CPU genera al atender la interrupción., dicha ventana tendrá el siguiente aspecto.

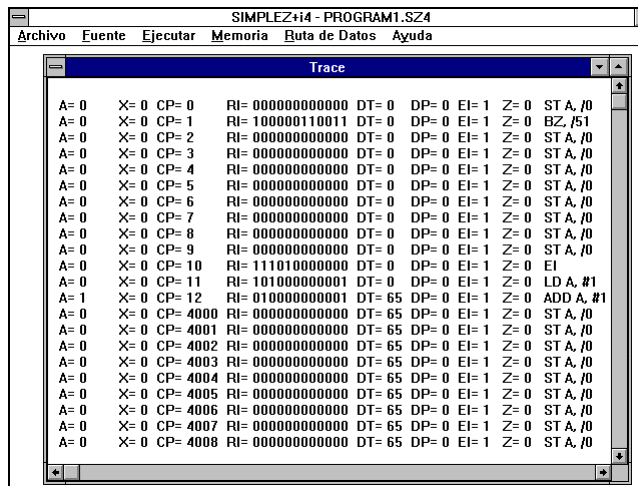


Fig. 20 - Ventana Trace una vez ejecutado el comando Rutina de Servicio

En la ventana anterior se puede ver cómo al pulsar una tecla, en este caso la letra mayúscula A (ASCII 65), se carga en el registro de entrada de datos (DT), la CPU termina de ejecutar la instrucción actual (ADD A, #1) y salta a la dirección 4000 a partir de la cual el programador debe cargar la rutina de atención de interrupción, en este caso no existe tal rutina y es por eso que a partir de la dirección 4000 el contenido de todos los registros es cero.

4 - Menú Memoria

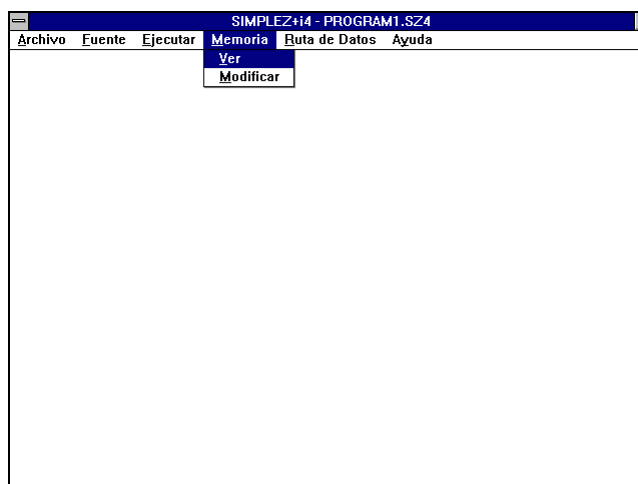


Fig. 21 - Menú Memoria

4.1 - Comando Memoria | Ver

Este comando le permite ver el contenido de la Memoria Principal de la SIMPLEZ+i4. El contenido aparece en representación decimal y octal, y además se especifica el mnemónico de dicho contenido. Tenga presente que en el contenido en crudo de la memoria no hay manera de identificar entre instrucción y operando, por lo tanto, para no caer en conclusiones erróneas, el programador debe interpretar este mnemónico sólo como el resultado de desensamblar el código de máquina correspondiente a esa posición de memoria.

Si usted modifica el contenido de la memoria debe actualizar lo que usted está viendo mediante el comando Actualizar de la ventana Memoria - Ver.

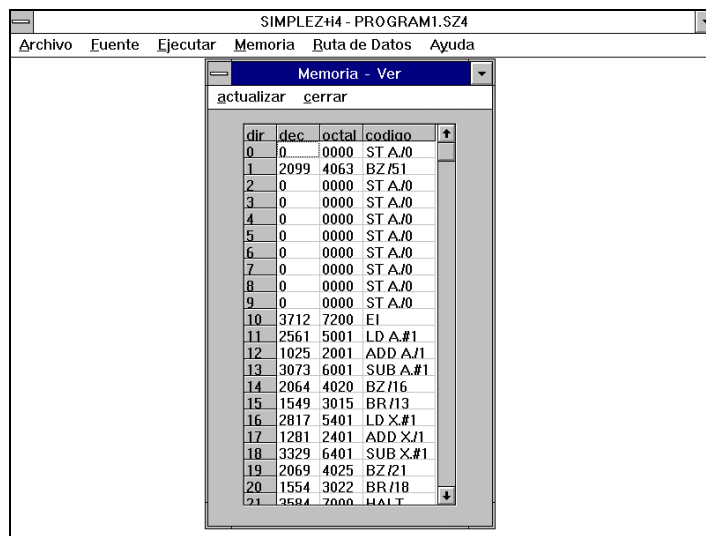


Fig. 22 - Ventana Memoria (Ver)

4.2 - Comando Memoria | Modificar

Este comando le permite acceder a una posición de memoria e ingresar un nuevo valor. La posición de memoria cuyo valor se desea modificar se debe especificar en decimal pero el nuevo valor puede ser especificado en decimal o en octal

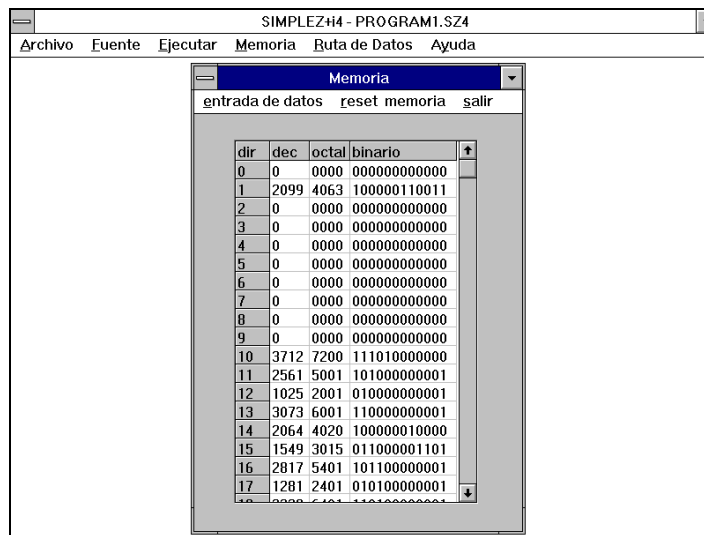


Fig. 23 - Ventana Memoria (Modificar)

Para poder modificar el contenido de una posición de memoria se debe ingresar al submenú **entrada de datos** de la ventana Memoria. Aparecen entonces las opciones antes mencionadas, octal o decimal. Si los valores que usted ingresa son incorrectos aparece un mensaje de error que le indica dicha situación.

El comando reset de la ventana Memoria pone a cero el contenido de todas las posiciones de memoria. Esta opción puede ser usada para 'limpiar' la memoria de datos que no son útiles.

5 - Menú Ruta de Datos

Ruta de Datos es un comando que permite ejecutar paso a paso las microinstrucciones de cada instrucción, visualizándolas en forma gráfica, lo que hace que, ésta, sea la parte más importante del Simulador en cuanto a la comprensión del funcionamiento de la parte estructural de la SIMPLÉZ+i4.

Para ejecutar cada microinstrucción presione el botón comando **ciclo siguiente**. La ejecución comenzará con la instrucción que se encuentra almacenada en la dirección que indica el Contador de Programa en la ventana Registros del menú Ejecutar | Paso a paso.

Cada vez que se salga de esta ventana por menú a través del comando Cerrar, y se vuelva a entrar el contador de micro instrucciones comienza en cero aún para el caso que se hubiera salido a la mitad de completada una instrucción.

Existe la posibilidad de simular una interrupción ya sea por teclado o por pantalla (solo desde la ventana ruta de dato), para ello debe estar habilitada la interrupción, es decir haber ejecutado antes una instrucción "EI". Momento en el cual cambia de color la unidad de entrada-salida de rojo a verde. Presionando cualquiera de los botones ya sea pantalla o teclado se indica al usuario mediante un cambio de color en la flecha de entrada o de salida respectivamente y además con una leyenda que se ha producido una interrupción, momento en el cual si esta habilitado el permiso termina de

ejecutar la última microinstrucción y salta a la ejecutar la rutina de atención de periférico indicándole con otra leyenda que esta procesando la interrupción.

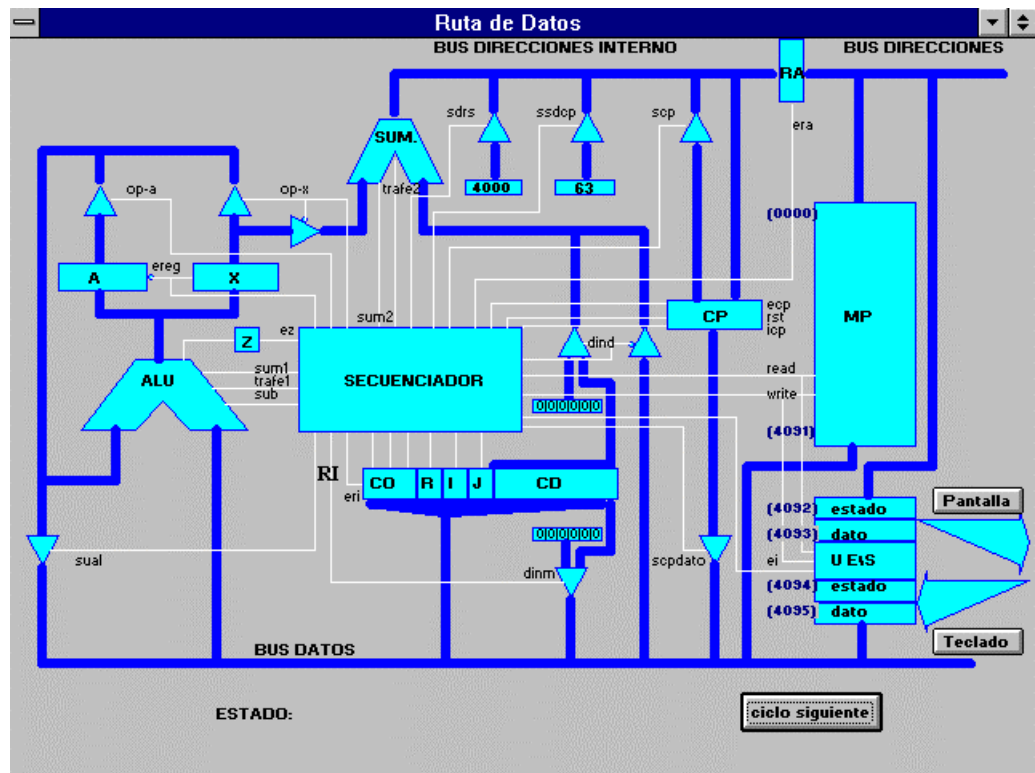
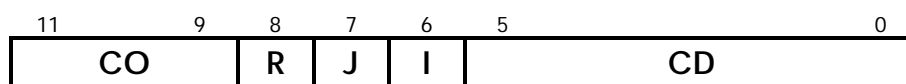


Fig. 24 - Ventana Ruta de Datos

Apéndice

A1 - Modelo Funcional de SIMPLEZ+i4



Formato de Instrucciones

CO	Código de Operación
R	Registro Fuente o Destino (R = 0 Acumulador, R = 1 Puntero X)
J I	Modos de Direccionamiento
CD	Campo de Dirección

Tabla 1 - Descripción de los campos de la palabra de Simplez+i4

Bit 7 J	Bit 6 I	Modos de Direccionamiento	Dirección Efectiva (DE) obtenida	
			Representación Simbólica	Descripción
0	0	Directo	$DE = (CD)$	El contenido de CD es directamente la dirección de la palabra que contiene el operando o la dirección de bifurcación.
0	1	Indirecto	$DE = ((CD))$	El contenido de CD apunta a una palabra de la MP que contiene la dirección de la palabra en la que está el operando o la dirección de bifurcación.
1	0	Indexado	$DE = (CD) + (X)$	El contenido de CD se suma al contenido de X para obtener la dirección de la palabra que contiene el operando o la dirección de bifurcación.
1	1	Ind. e Indexado	$DE = ((CD)) + (X)$	Se realiza una postindexación, es decir, primero se acude al puntero para buscar una dirección y luego se suma al contenido de X para obtener la dirección de la palabra que contiene el operando o la dirección de bifurcación.

Tabla 2 - Modos de Direccionamiento

CO Binario	CO Octal	CO Nemónico	Descripción
0 0 0	0	ST	Almacena (STORE) el contenido del acumulador si R=0 o, el contenido del Reg X si R=1, en la palabra cuya dirección de memoria se encuentra en DE.
0 0 1	1	LD	Carga (LOAD) en el acumulador si R=0 o, en el RegX si R=1, el contenido de la palabra de memoria cuya dirección se encuentra en DE.
0 1 0	2	ADD	Suma (ADD) al acumulador si R=0 o, al Reg X si R=1, el contenido de la palabra de memoria cuya dirección se encuentra en DE, dejando el resultado en el acumulador o en el Reg X, según corresponda.
0 1 1	3	BR	Bifurca (BRANCH) a la dirección indicada por la DE, es decir, la próxima instrucción a ejecutar es la que se encuentra en la palabra de memoria cuya dirección está indicada por DE.
1 0 0	4	BZ	(BRANCH IF ZERO) Bifurca a la dirección indicada por la DE si la última operación efectuada por la ALU es 0 (Z=!). Si no, continua secuencialmente.
1 0 1	5	LD	Carga en forma inmediata en el acumulador si R=0 o, en el RegX si R=1, el valor expresado en CD.
1 1 0	6	SUB	Resta (SUBTRACTION) al contenido del acumulador si R=0 o, al contenido del reg X si R=1, el valor expresado en CD, dejando el resultado en el registro indicado por R.
1 1 1 con R = 0 J = 0	7 0	HALT	Hace que el proceso se detenga.
1 1 1 con R = 0 J = 1	7 2	EI	Habilita las interrupciones.
1 1 1 con R = 1 J = 0	7 4	DI	Inhabilita las interrupciones.

Tabla 3 - Conjunto de Instrucciones

A2 - Convención Simbólica

Si bien los convenios simbólicos del lenguaje ensamblador de la SIMPLEZ+i4 que aparece en el libro de consulta *Conceptos Básicos de Arquitectura de Computadoras* de Gregorio Fernández han sido respetados casi en su totalidad cabe hacer una aclaración respecto de los cambios que se han realizado sobre los mismos.

- . En las instrucciones que hacen referencia a un registro (A o X), el punto que antecede al nombre del registro ha sido eliminado.
- . En las instrucciones que utilizan los modos de direccionamiento indirecto, indexado o ambos, los corchetes han sido reemplazado por paréntesis.

La siguiente tabla resume la sintaxis de los modos de direccionamiento para instrucciones que utilizan operandos (ST, LD, ADD, SUB) y para aquellas que indican una bifurcación (BR, BZ) utilizando un ejemplo.

Modo de Direccionamiento	Instrucciones con operandos	Instrucciones con saltos
Inmediato*	LD A, #10 (SUB A, #1)	No se usa
Directo	LD A, /10	BR /10
Indirecto	LD A, (/10)	BR (/10)
Indexado	LD A, /10 (.X)	BR /10 (.X)
Indirecto e Indexado	LD A, (/10) (.X)	BR (/10) (.X)

Tabla 4 - Sintaxis de los Modos de Direccionamiento

* Recordar que el direccionamiento inmediato implica el uso únicamente de las instrucciones LD (CO = 101) y SUB (CO = 110)

A3 - Directivas del Ensamblador

Todas las Directivas deben ser precedidas por un punto.

.ORG <valor>

Hace que el programa se cargue en memoria a partir de la dirección indicado por <valor>.

<etiqueta> **.DATA** <valor>

Asigna a <etiqueta> una dirección específica y carga en la misma el dato <valor>.

<etiqueta> **.EQU** <valor>

Asigna una dirección de memoria, <valor>, a un nombre, <etiqueta>.

.END

Directiva de finalización del programa fuente.

.SUBR <valor>

Carga un bloque de instrucciones (subrutina) a partir de la dirección <valor>.

.FINSUBR

Indica la finalización de la Directiva " .SUBR " .
