

1. Menú principal de CRONOS

Este programa le permite la generación de cronogramas a partir de un fichero de instrucciones que tiene que haber sido cargado previamente. Asimismo podrá obtener el flujo de la información sobre los buses. Todo lo anterior referido a la arquitectura que podrá observar al elegir la opción flujo del menú

1.1. Menú de edición

Desde este menú podrá realizar todas las acciones que desee sobre el fichero de instrucciones. Estas acciones son: Editarlo, Nombrarlo, Cambiar el directorio. Debe tener en cuenta que no podrá salir al menú principal a menos que tenga un fichero con instrucciones correctas. Si desea salir del programa sin realizar acción alguna, deberá, o bien, entrar en edición del fichero y escribir END. (con lo cual podrá salir al menú principal), o bien, pulsar ALT-X para salir del programa de forma incondicional.

1.2. Editor de CRONOS

Aquí podrá editar el fichero de instrucciones. Las operaciones básicas que puede realizar son:

- Cursores para mover el puntero.
- ENTER para línea nueva.
- BORRAR para eliminar una letra.
- RETROCESO para borrar hacia la izquierda.
- PGUP, PGDN para mover 20 líneas arriba o abajo.
- INS para cambiar modo insertar.
- CTRL-F1 para obtener ayuda sobre una instrucción.
- Podrá salir incondicionalmente con ALT-X.

1.3. Cambio de directorio

Usted puede cambiar de directorio simplemente tecleando el nuevo camino y pulsando ENTER al final. Si el nuevo directorio no es correcto, aparecerá un aviso en la pantalla y no será efectivo el cambio. Se puede anular la acción pulsando ESC en cualquier momento

1.4 Escribir en fichero

Puede asignar al fichero de instrucciones un nombre determinado. Si desea anular la acción, simplemente deber pulsar ESC en cualquier momento, y tanto el nombre del fichero anterior como su contenido se mantendrán intactos.

2 Ayuda sobre instrucciones

Las instrucciones correctas son:

ADC	CALL	END.	OR
AND	CLC	ENDL	RET
BC	CLE	INC	SHR
BNC	CLS	LD	ST
BNS	CLZ	LOOP	SUB
BNZ	DEC	MOVE	XOR
BS	DEF	MULT	
BZ	DIV	NOT	

ADC

Esta instrucción produce la suma teniendo en cuenta el bit de acarreo de los operandos. Los modos de direccionamiento válidos son:

```
ADC .X .Y
ADC .X .Y++
ADC .X AAA
ADC .X [AAA]
ADC .X (.Y) AAA
```

```
ADC .X  [(.Y) AAA]
ADC .X  (.Y++) AAA
ADC .X  [(.Y++) AAA]
```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

AND

Esta instrucción produce la operación lógica AND sobre los operandos. Los modos de direccionamiento válidos son:

```
AND .X  .Y
AND .X  .Y++
AND .X  AAA
AND .X  [AAA]
AND .X  (.Y) AAA
AND .X  [(.Y) AAA]
AND .X  (.Y++) AAA
AND .X  [(.Y++) AAA]
```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

BC

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de acarreo es 1. Es obligatorio colocar la directiva {C0} o {C1} al final. Los modos de direccionamiento son:

```
BC .X
BC .X++
BC AAA
BC [AAA]
BC (.X) AAA
BC [(.X) AAA]
BC (.X++) AAA
BC [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

BNC

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de acarreo es 0. Es obligatorio colocar la directiva {C0} o {C1} al final. Los modos de direccionamiento son:

```
BNC .X
BNC .X++
BNC AAA
BNC [AAA]
BNC (.X) AAA
BNC [(.X) AAA]
BNC (.X++) AAA
BNC [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

BNS

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de signo es 0. Es obligatorio colocar la directiva {S0} o {S1} al final. Los modos de direccionamiento son:

```
BNS .X
BNS .X++
BNS AAA
```

```
BNS [AAA]
BNS (.X) AAA
BNS [(.X) AAA]
BNS (.X++) AAA
BNS [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

BNZ

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de Cero es 0. Es obligatorio colocar la directiva {Z0} o {Z1} al final. Los modos de direccionamiento son:

```
BNZ .X
BNZ .X++
BNZ AAA
BNZ [AAA]
BNZ (.X) AAA
BNZ [(.X) AAA]
BNZ (.X++) AAA
BNZ [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

BS

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de signo es 1. Es obligatorio colocar la directiva {S0} o {S1} al final. Los modos de direccionamiento son:

```
BS .X
BS .X++
BS AAA
BS [AAA]
BS (.X) AAA
BS [(.X) AAA]
BS (.X++) AAA
BS [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

BZ

Esta instrucción produce una bifurcación al lugar indicado por el operando sólo si el bit de cero es 1. Es obligatorio colocar la directiva {Z0} o {Z1} al final. Los modos de direccionamiento son:

```
BZ .X
BZ .X++
BZ AAA
BZ [AAA]
BZ (.X) AAA
BZ [(.X) AAA]
BZ (.X++) AAA
BZ [(.X++) AAA]
```

Siendo X un registro (0..9) y AAA es un número decimal.

CALL

Esta instrucción llama a una subrutina. Es necesario haberla definido previamente y colocar su nombre al final de la instrucción. Modos de direccionamiento:

```
CALL .X          {NOM_SUB}
```

```
CALL .X++          {NOM_SUB}
CALL AAA           {NOM_SUB}
CALL [AAA]         {NOM_SUB}
CALL (.X) AAA      {NOM_SUB}
CALL [(.X) AAA]    {NOM_SUB}
CALL (.X++) AAA    {NOM_SUB}
CALL [(.X++) AAA]  {NOM_SUB}
```

Donde X es un registro (0..9) y AAA es un número decimal.

CLC

Coloca a cero el bit de acarreo. No lleva parámetros

CLE

Coloca a cero el bit de interrupción externa. No lleva parámetros

CLS

Coloca a cero el bit de signo. No lleva parámetros

CLZ

Coloca a cero el bit de cero. No lleva parámetros

DEC

Esta instrucción decrementa el contenido de un registro en una unidad. Sólo dispone de un modo de direccionamiento:

```
DEC .X
```

Donde X es un registro numerado del 0 al 9

DEF

Este comando define una subrutina. Es necesario colocar el nombre de la subrutina detrás de éste y debe ir siempre acompañada de la instrucción RET. La sintaxis de este comando es:

```
DEF NOM_SUB
INSTRUCCION
INSTRUCCION
...
RET
```

DIV

Esta instrucción produce la división de los operandos. Los modos de direccionamiento válidos son:

```
DIV .X .Y
DIV .X .Y++
DIV .X AAA
DIV .X [AAA]
DIV .X (.Y) AAA
DIV .X [(.Y) AAA]
DIV .X (.Y++) AAA
DIV .X [(.Y++) AAA]
```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

END.

Es obligatorio cerrar el fichero de instrucciones con este comando. Toda instrucción que se encuentre después de este comando no será tomada en cuenta.

ENDL

Este comando debe cerrar un conjunto de instrucciones encabezado con un LOOP. Si no cerramos correctamente el análisis terminar indicando el error.

INC

Esta instrucción incrementa el contenido de un registro en una unidad. Solo dispone de un modo de direccionamiento:

```
INC .X
```

Donde X es un registro numerado del 0 al 9.

LD

La instrucción LD se encarga de cargar el contenido de una dirección de memoria en un registro determinado. Los modos de direccionamiento correctos son:

```
LD .X AAA
LD .X [AAA]
LD .X (.Y) AAA
LD .X [(Y) AAA]
LD .X (.Y++) AAA
LD .X [(Y++) AAA]
```

Donde X e Y son registros del 0 al 9 y AAA es un número decimal.

LOOP

Esta instrucción genera un bucle que se repetirá tantas veces como indique el direccionamiento. Los modos de direccionamiento correctos son:

```
LOOP .X
LOOP .X++
LOOP AAA
LOOP [AAA]
LOOP (.X) AAA
LOOP [(X) AAA]
LOOP (.X++) AAA
LOOP [(X++) AAA]
```

Donde X es un registro (0..9) y AAA es un número decimal.

MOVE

La instrucción MOVE se encarga de cargar el contenido de un registro en otro registro. Sólo dispone de un modo de direccionamiento que es el siguiente:

```
MOVE .X .Y
```

donde X e Y son registros numerados del 0 al 9.

MULT

Esta instrucción produce la multiplicación de los operandos. Los modos de direccionamiento válidos son:

```
MULT .X .Y
MULT .X .Y++
MULT .X AAA
MULT .X [AAA]
MULT .X (.Y) AAA
MULT .X [(Y) AAA]
MULT .X (.Y++) AAA
MULT .X [(Y++) AAA]
```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

NOT

Esta instrucción produce la operación lógica NOT sobre el operando. Sólo dispone de un modo de direccionamiento:

NOT .X

Donde X es un registro numerado del 0 al 9.

OR

Esta instrucción produce la operación lógica OR sobre los operandos. Los modos de direccionamiento válidos son:

OR .X .Y
OR .X .Y++
OR .X AAA
OR .X [AAA]
OR .X (.Y) AAA
OR .X [(.Y) AAA]
OR .X (.Y++) AAA
OR .X [(.Y++) AAA]

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

RET

Este comando es necesario al finalizar un conjunto de instrucciones encabezado por DEF. Estas instrucciones comprenden una subrutina. Cuando se encuentra se produce un retorno al programa principal.

SHR

Esta instrucción produce un desplazamiento de un bit a la derecha en un registro. Solo dispone de un modo de direccionamiento:

SHR .X

Donde X es un registro numerado del 0 al 9.

ST

La instrucción ST carga el contenido de un registro en una posición de memoria determinada. Los direccionamientos válidos para esta instrucción son:

ST .X AAA
ST .X [AAA]
ST .X (.Y) AAA
ST .X [(.Y) AAA]
ST .X (.Y++) AAA
ST .X [(.Y++) AAA]

Donde X e Y son registros del 0 al 9 y AAA es un número decimal.

SUB

Esta instrucción produce la resta de los operandos. Los modos de direccionamiento válidos son:

SUB .X .Y
SUB .X .Y++
SUB .X AAA
SUB .X [AAA]
SUB .X (.Y) AAA

```

SUB .X  [(.Y) AAA]
SUB .X  (.Y++) AAA
SUB .X  [(.Y++) AAA]

```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

XOR

Esta instrucción produce la operación lógica XOR sobre los operandos. Los modos de direccionamiento válidos son:

```

XOR .X  .Y
XOR .X  .Y++
XOR .X  AAA
XOR .X  [AAA]
XOR .X  (.Y) AAA
XOR .X  [(.Y) AAA]
XOR .X  (.Y++) AAA
XOR .X  [(.Y++) AAA]

```

Donde X e Y son registros numerados del 0 al 9 y AAA es un número decimal.

3. Ayuda sobre directivas

Las directivas del traductor permitidas son:

{CDR}	{P}	{Z0}	{Z1}
{D}	{C0}	{C1}	
{E}	{S0}	{S1}	

{CDR}

Esta directiva nos permite decidir si queremos un cálculo del direccionamiento relativo automático en el periodo de decodificación. Si no es puesta no se producirá el cálculo, pero si se coloca una (o más veces) sí se producirá. Es necesario que esté escrita en una línea aparte.

{D}

Esta directiva permite cuando esté presente que se produzca un trap de desbordamiento en la instrucción a la precede. Su sintaxis es la siguiente:

```
INSTRUCCION {D}
```

Se debe tener en cuenta que sólo será utilizable en instrucciones que utilicen la ALU en sus operaciones.

{E}

Esta directiva permite cuando esté presente que se produzca un trap de interrupción externa en la instrucción a la que precede. Su sintaxis es la siguiente:

```
INSTRUCCION {E}
```

Siempre puede ser utilizada.

{P}

Esta directiva permite cuando esté presente que se produzca un trap de paridad en la instrucción a la que precede. Su sintaxis es la siguiente:

```
INSTRUCCION {P}
```

Se debe tener en cuenta que sólo será utilizable en instrucciones que generen lecturas en memoria.

{C0}

Esta directiva permite indicar que el acarreo será cero para la instrucción a la que precede. Su sintaxis es:

INSTRUCCION {C0}

Esta directiva sólo ser utilizada por instrucciones que verifiquen el estado del bit de acarreo.

{S0}

Esta directiva permite indicar que el signo será positivo para la instrucción a la que precede. Su sintaxis es:

INSTRUCCION {S0}

Esta directiva sólo ser utilizada por instrucciones que verifiquen el estado del bit de signo.

{Z0}

Esta directiva permite indicar que el resultado de la operación anterior resultó cero. Su sintaxis es:

INSTRUCCION {Z0}

Esta directiva sólo ser utilizada por instrucciones que verifiquen el estado del bit de cero.

{C1}

Esta directiva permite indicar que el acarreo será uno para la instrucción a la que precede. Su sintaxis es:

INSTRUCCION {C1}

Esta directiva sólo ser utilizada por instrucciones que verifiquen el estado del bit de acarreo.

{S1}

Esta directiva permite indicar que el signo será negativo para la instrucción a la que precede. Su sintaxis es:

INSTRUCCION {S1}

Esta directiva sólo será utilizada por instrucciones que verifiquen el estado del bit de signo.

{Z1}

Esta directiva permite indicar que el resultado de la operación anterior resultó distinto de cero. Su sintaxis es:

INSTRUCCION {Z1}

Esta directiva sólo ser utilizada por instrucciones que verifiquen el estado del bit de cero.

4 Flujo de la información

En esta pantalla podrá observar el flujo de la información por los buses. Puede utilizar solamente las teclas de cursor derecha e izquierda para moverse por el

flujo. Para imprimir la pantalla de flujo deber pulsar la tecla P. En cualquier momento podrá observar el fichero de instrucciones pulsando CLTR-F1 (para salir pulsar ESC); para moverse por el texto usar las teclas de cursor arriba y abajo o páginas arriba y abajo. Para salir de esta pantalla no tiene más que pulsar ESC.

5 Presentación del cronograma

Aquí podrá ver el cronograma generado a partir del fichero de instrucciones que usted creó. Para moverse por él podrá hacerlo periodo a periodo con las teclas de cursor derecha e izquierda o instrucción por instrucción con las teclas página arriba y página abajo. En cualquier momento podrá observar el fichero de instrucciones pulsando CLTR-F1 (para salir pulsar ESC); para moverse por el texto usar las teclas de cursor arriba y abajo o páginas arriba y abajo. Para salir de esta pantalla no tiene más que pulsar ESC.

6 Generador de cronogramas manual.

Aquí podrá generar el cronograma que desee de forma que podrá ver el flujo de información generado por éste, guardarlo en disco para su posterior utilización y utilizar dicho cronograma para actualizar los ficheros .DAT de CRONOS. Las teclas utilizadas son:

Q: Salir.
D: Cambio directorio
N: Cronograma nuevo.
F2: Grabar
I: Imprimir.
F: Flujo inform.
V: Vaciar el cron. desde cursor
L: Cargar un fichero de cronog.
C: Cambiar CDR
P: Periodo de decodificación.
R: Periodo de ruptura.
BARRA ESPACIADORA: Cambio señal