

MSX88: UNA HERRAMIENTA PARA LA ENSEÑANZA DE LA ESTRUCTURA Y FUNCIONAMIENTO DE LOS ORDENADORES.

Rubén de Diego Martínez

Universidad Politécnica de Madrid. E.U.I.T. de Telecomunicación.
Departamento de Ingeniería y Arquitecturas Telemáticas.
Carretera de Valencia Km 7, 28031 MADRID.

Tfno: (91) 336 78 02, FAX: (91) 336 78 17
Correo Electrónico: rdiego@diatel.upm.es

RESUMEN

Cuando se aborda la enseñanza de la estructura y funcionamiento de los ordenadores en sus aspectos básicos y primarios, el profesor se encuentra con la dificultad de tener que realizar modelos dinámicos que permitan visualizar y hacer entender lo que ocurre dentro de una Unidad Central de Proceso (CPU), o en el propio ordenador. MSX88 permite solucionar en parte el problema, dado que es un simulador gráfico de un ordenador didáctico basado en la familia iAPx86, que muestra los flujos de información existentes entre los diversos elementos que los componen.

MSX88 permite al profesor, bien a través de la utilización de las modernas aulas informatizadas, bien utilizando los nuevos recursos audiovisuales (pantallas de cuarzo líquido para retroproyector, cañones de video...), plantear sus clases pudiendo analizar de forma cómoda la estructura y funcionamiento del ordenador. La herramienta permite igualmente al alumno repasar las clases del profesor y realizar prácticas, dirigidas o no dirigidas, tendentes a reforzar lo aprendido en las clases teóricas.

1. BREVE HISTORIA DEL MSX88

MSX88 no es una herramienta nueva, se dispone de una amplia experiencia en su utilización dentro de la enseñanza reglada impartida por el Departamento de Ingeniería y Arquitecturas Telemáticas de la E.U.I.T. de Telecomunicación de la U.P.M. y en diversos cursos de formación, siempre combinada con prácticas sobre *hardware* real. Están disponibles varias versiones de la herramienta, incluyendo versiones de demostración¹.

1 En la presentación contenida en esta ponencia existe la misma dificultad que MSX88 trata de solucionar, por tanto se aconseja al lector interesado que solicite una copia del programa, el cual contribuirá a aliviar las posibles lagunas existentes en la comprensión de su contenido.

El MSX88 nace en la antigua Cátedra de Ordenadores de la E.U.I.T. de Telecomunicación de la UPM en 1988 cuando el ordenador personal (PC) hace su irrupción en el mundo informático a precios asequibles y es posible dotar un aula completa para la realización de prácticas.

Previo a la era del PC, la enseñanza básica de la estructura y funcionamiento de los ordenadores se soportaba con el microprocesador Intel 8085 y con un microordenador didáctico basado en esta CPU: el SDK85. Sobre este microordenador el alumno realizaba distintos programas en código máquina, que se completaban con programación en Lenguaje de Ensamble (L.E.) mediante ensambladores cruzados.

Con la llegada del PC es prácticamente obligado realizar un cambio de CPU, pasando de la CPU 8085 a la Intel 8086/88 [1], con el correspondiente incremento de complejidad. Había, y hay, un problema didáctico importante: abordar el estudio de una CPU comercial de 16 bits como el 8086 de forma global es algo muy difícil. Era necesario plantear una simplificación que permitiera afrontar el problema de forma parcial, sin enfrentarse a toda la complejidad del 8086 desde el momento inicial.

Por otra parte existía, y existe, el problema de aportar visibilidad a lo que ocurre dentro de la CPU, y en última instancia dentro del ordenador. A este respecto, en la mencionada Cátedra ya existían diversas experiencias plasmadas en el Ordenador Didáctico MicroSimplex [8] inspirado a su vez en otro más antiguo desarrollado en Standard Eléctrica: el Simplex [7], también utilizado años atrás en la docencia. Ambos ordenadores didácticos tenían como característica común mas interesante el permitir visualizar los flujos de información de los buses del ordenador, además de la información contenida en los distintos registros.

Con todos estos antecedentes no resultará extraño el inicio del desarrollo del MSX88, un programa que emula el funcionamiento de un ordenador, basado en una CPU, el SX88, que a su vez se puede ver como un 8088 al que se le han ocultado todas aquellas características que resultarían tanto de difícil impartición, por parte del profesor, como de difícil entendimiento por parte del alumno.

Como cualquier programa el MSX88 ha sufrido una notable evolución, dado que desde la versión 1.0, en la que únicamente se ofrecía la visión de la CPU y memoria, se ha pasado a la versión 3.0 en la que el MSX88 es ya un completo ordenador que incorpora una periferia y admite diversos conexiones y configuraciones, tal como se verá más adelante.

Desde sus primeras versiones el MSX88 está diseñado de forma que sus elementos tengan un equivalente lo mas parecido posible en CPUs y periféricos de la familia iAPX86/88, o al menos, con la idea que el alumno, al pasar a trabajar sobre elementos reales, sólo tenga que añadir, nunca quitar u olvidar, a lo aprendido sobre el MSX88.

2. EL CONJUNTO DE HERRAMIENTAS DEL ENTORNO MSX88

En la versión 3.0 el MSX88 está formado por el siguiente conjunto de programas:

- **ASM88**: Ensamblador del conjunto de instrucciones que el SX88 es capaz de ejecutar. Se mantiene, en su mayor parte, la sintaxis que Intel propone para el L.E. del 8086/88. El código producido por el ensamblador es no reubicable, el alumno deberá elegir una ubicación fija para los programas y datos dentro del espacio de memoria definido en el MSX88, de esta manera siempre se tiene un control total sobre lo que se está haciendo.

Este ensamblador es un ensamblador cruzado, es decir se ejecuta en una máquina distinta -el PC bajo MS-DOS- de en la que correrán los programas que ensambla -el MSX88- .

Un posible aspecto de los listados que genera se puede observar en la figura 1.

```

                                página 1
Fichero Listado: DEMOPIO.LST
Programa Fuente: DEMOPIO.ASM
Fecha: Tue Mar 15 18:14:10 1994

Dir. Código Máquina      Línea Código en Lenguaje de Ensamble
1      ;Programa que muestra el funcionamiento del PIO en el modo...
2      ;...de conexionado I (PIO conectado a LEDS y microinterrupt.)
3      ;Programador: Rubén de Diego
4      ;
5      ;Este programa configura el PIO (conexionado I) con el puerto A
6      ;...de entrada y el puerto B de salida. A continuación se queda
7      ;...en un bucle infinito leyendo del puerto asociado a los mi-
8      ;...crointerruptores, escribiendo a continuación el valor leído
9      ;...en el puerto asociado a los leds.
10     ;
11     ;      ORG 2000H
12
2000 B8 FF 00      13      MOV AX,0FFH      ;Programa los puertos...
2003 E6 32        14      OUT 32H,AL      ;... PA de entrada...
2005 32 C0        15      XOR AL,AL
2007 E6 33        16      OUT 33H,AL      ;... PB de salida.
17
2009 E4 30        18      BUCLE: IN AL,30H
200B E6 31        19      OUT 31H,AL
200D E9 09 20     20      JMP BUCLE      ;Bucle infinito, se parará con ESC.
21
22      END

S I M B O L O S:

Nombre:      Tipo:      Valor:
BUCLE      Label      2009h

```

Figura 1. Listado generado por el MSX88

- **LINK88:** Montador de los programas objeto producidos por el ASM88. No tiene funciones de enlazador, dado que por el tamaño de los programas a ejecutar en el MSX88, no tiene sentido realizar una programación modular. Esta herramienta se mantiene sobre todo para mantener una coherencia con el resto de entornos de desarrollo software existentes y que el alumno ya maneja o manejará. Al igual que el ASM88, el LINK88 es un montador cruzado.
- **MSX88:** Emulador que constituye el núcleo del entorno de herramientas. Es, como se ha mencionado anteriormente, un emulador de un ordenador basado en una CPU denominada SX88.

Tras arrancarlo -tecleando MSX88- en la pantalla aparece la denominada pantalla principal o *pantalla 0* (figura 2), donde se muestra la CPU, el bloque de memoria, un bloque de periféricos y la ventana de comunicación con el usuario, que se comporta como una pantalla de un ordenador.

Existe un conjunto de periféricos, además de la pantalla, que el MSX88 comparte con el PC donde corre. Estos son: el teclado, las unidades de disco que se dispongan, y, si existe una, la impresora (LPT1:).

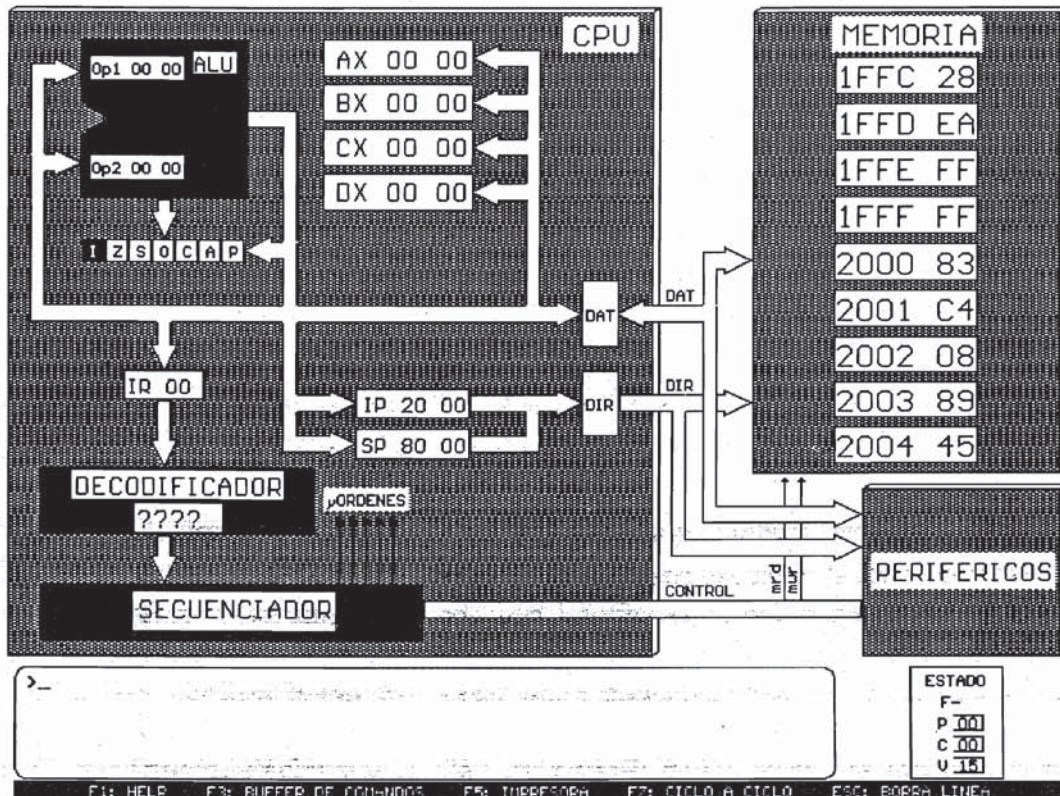


Figura 2. Pantalla principal del MSX88

El MSX88 posee además los siguientes periféricos:

- **PIO:** Similar al Intel 8255 (modo 0).
- **Periférico de Handshaking:** Similar al modo 2 del Intel 8255 -denominado en adelante *hand-*.
- **Controlador de Interrupciones (PIC):** Similar al Intel 8259
- **Timer:** Conectado a un reloj de periodo 1 segundo.
- **Barra de Leds y Microconmutadores** accionables desde el teclado.
- **Controlador de Acceso Directo a Memoria (CDMA):** Similar al I8237.
- **Impresora:** Se conecta a diversos elementos a través de un interfaz Centronics que deberá ser programado por el alumno.

Estos periféricos admiten diversos conexiones, siendo la mayor parte de ellos totalmente programables.

3. DESCRIPCION DETALLADA DEL MSX88.

Tratando de distinguir grandes bloques en el MSX88, se podría decir que está compuesto por: CPU SX88, memoria, periféricos y programa monitor.

3.1. CPU SX88

CPU con una arquitectura similar a la del 8088, sus características fundamentales son:

- Arquitectura interna de 16 bits y externa de 8 bits.
- Bus de direcciones de 16 bits, capaz por tanto de direccionar 64 Kbytes de memoria.
- 32 Kbytes de memoria disponibles para el usuario (0-7FFFH).
- Mantiene el mismo esquema de interrupciones que el 8086/88: Líneas NMI e INT y las interrupciones software INT xx e INTO. La línea NMI no está, de momento, asociada a evento alguno. El conexionado de INT se describe en el apartado dedicado al PIC.
- Registros de uso general AX, BX, CX y DX. Se pueden tratar también como registros de 8 bits (AH, AL, BH, BL....).
- ALU de 16 bits capaz de ejecutar las operaciones: ADD, SUB, AND, OR, XOR, NOT, INC y DEC.
- Registro de indicadores con los flags de: Cero, Paridad, Paridad Auxiliar, Signo, Overflow e indicador de interrupciones permitidas.
- Registros contador de programa (IP) y puntero de pila (SP) de 16 bits.
- Soporta los siguientes modos de direccionamiento: Dato Inmediato, Registro, Directo e Indirecto basado en el registro BX.
- Bus de control integrado por las líneas: INT, INTA, NMI, RW, RD, IO/M, HOLD, HLDA.
- Existen además dos registros internos de almacenamiento temporal de datos que aparecen en pantalla únicamente cuando son necesarios, tratando de esta manera de reforzar el impacto visual que haga comprender al alumno su necesidad.

Se puede sintetizar el esfuerzo de diseño de la CPU diciendo que es similar a un 8088 en el que se abstrae la Unidad de Interfaz con los Buses (BIU), asimilándola a una interfaz más simplificada en el que se elimina la cola de instrucciones y los registros de segmento. La EU se ve también simplificada, eliminando los registros BP, DI y SI y definiendo una arquitectura clásica con un Registro de Instrucción (RI) un decodificador y un secuenciador.

Se ha elegido el 8088 como base, en vez del 8086, debido a que su arquitectura de 8/16 bits se considera, didácticamente, más completa.

3.1.1. Conjunto de Instrucciones

El siguiente conjunto de instrucciones se ha demostrado suficiente para resolver los sencillos algoritmos necesarios para conseguir los objetivos que se pretenden con el MSX88. Su código máquina es como el del 8086/88. Este código máquina es altamente ineficiente, debido a que el conjunto de intrucciones es considerablemente menor en el SX88, pero esta circunstancia da pie a interesantes ejercicios de rediseño de dicho código.

En la versión 3.0 se han definido las siguientes instrucciones:

- **Transferencia de Datos:** MOV, IN, OUT, PUSH, POP, PUSHF y POPF
- **Aritmético-Lógicas:** ADD, ADC, SUB, SBB, AND, OR, XOR, NOT y NEG.
- **Incremento-Decremento:** INC y DEC.
- **Comparación:** CMP.

- **Cambio de Flujo de Programa:** JMP, JZ, JNZ, JS, JNS, JC, JNC, JO, JNO, CALL y RET.
- **Gestión de las Interrupciones:** INT, IRET, INTO, STI y CLI.
- **Control:** NOP y HLT.

Los lectores no familiarizados con la CPU 8086/88 y/o su lenguaje de ensamblaje, en [2], [4] y [5] podrán encontrar información complementaria

Obsérvese que conforme a lo manifestado, pasar del MSX88 al 8088, tanto en el plano estructural, como en lo relacionado con el lenguaje de ensamblaje e instrucciones, es tan sencillo como añadir nuevos elementos y funcionalidades asociadas que complementen las ya existentes.

3.2. LA MEMORIA DEL MSX88

El MSX88 posee un total de 64 Kbytes de memoria direccionados linealmente. El usuario tiene disponibles un total de 32Kbytes desde la posición 0 a la 7FFFH, memoria en la que cabrá holgadamente cualquier programa pensado para el MSX88. Escrituras en memoria por encima de esta dirección pueden tener efectos graves en la ejecución del MSX88, dado que, para ofrecer más realismo, se ha hecho coincidir la ubicación de esta zona con el ejecutable del propio programa MSX88. Al alumno se le justifica diciendo que por encima de la dirección 7FFFH está cargado el programa monitor.

El primer Kbyte está reservado para los vectores de interrupción tal como ocurre en el microprocesador en el que se inspira.

En la pantalla principal del MSX88 (figura 2) el bloque de memoria se representa como un cuadrado que contiene una ventana de que permite visualizar 9 posiciones de memoria. La celda central siempre representa la posición de memoria seleccionada.

Existe la posibilidad de abrir una segunda ventana (figura 7) del mismo tamaño, pero ésta permanece siempre fija centrada en una posición especificada. De esta manera el alumno podrá ver (trazar) la variación de un conjunto de posiciones de memoria durante la ejecución de un programa.

3.3. LOS PERIFÉRICOS DEL MSX88

El MSX88, aparte de la pantalla principal, tiene la posibilidad de proporcionar una segunda visión de su estructura, denominada *pantalla 1*, centrada ésta en el estudio de los periféricos, tal como se ilustra en la figura 3.

La mencionada figura representa en la parte superior derecha el conjunto CPU-Memoria junto con la lógica de selección de la periferia y la generación de las señales de lectura/escritura. Además, en la parte superior derecha se pueden ver el PIC y el Timer.

En la parte central de la pantalla se observa el PIO conectado a una barra de LEDs y microconmutadores.

En la parte inferior se sigue manteniendo la ventana de comunicación con el usuario.

La configuración mencionada se corresponde con el denominado *conexiónado 0*, que sería el conexiónado más simple -sin contemplar los elementos relacionados con las interrupciones- Sobre este conexiónado se pueden realizar múltiples prácticas, destacando por su sencillez las relacionadas con la programación de un PIO conectado a unos LEDs y microconmutadores (ver el ejemplo de la figura 1).

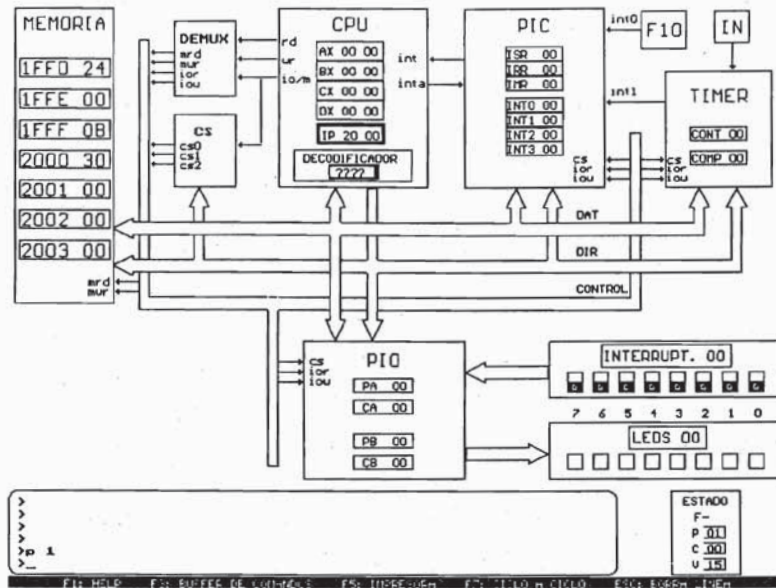


Figura 3: Conexión básica del MSX88

El *conexiónado 1* se muestra en la figura 4, donde la parte superior de la pantalla permanece invariante -será una constante para el resto de conexiones- habiéndose sustituido las barras de LEDs y microconmutadores por una impresora con una interfaz *centronics*, que el alumno deberá implementar.

En el *conexiónado 2*, figura 5, se mantiene la impresora pero ahora conectada a un dispositivo que hace *handshaking*, implementando igualmente la interfaz *centronics* necesario para conectarse a la impresora. El Hand puede relacionarse con la CPU a través de interrupciones o por consulta de estado.

En el *conexiónado 3* (figura 6), último por el momento, aparece el CDMA conectado al Hand. Por supuesto no es un buen ejemplo de utilización de un CDMA y así se debe hacer ver

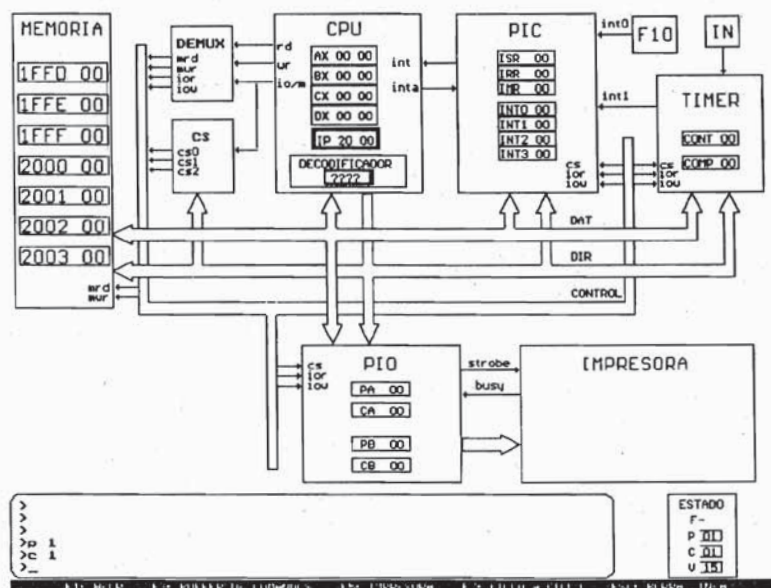


Figura 4. Conexión 1 de la periferia del MSX88

a los alumnos, pero toda la problemática relacionada con la programación de estos dispositivos aparece en este conexionado, con lo que quedan cubiertos, tal como se pretende, los aspectos didácticos.

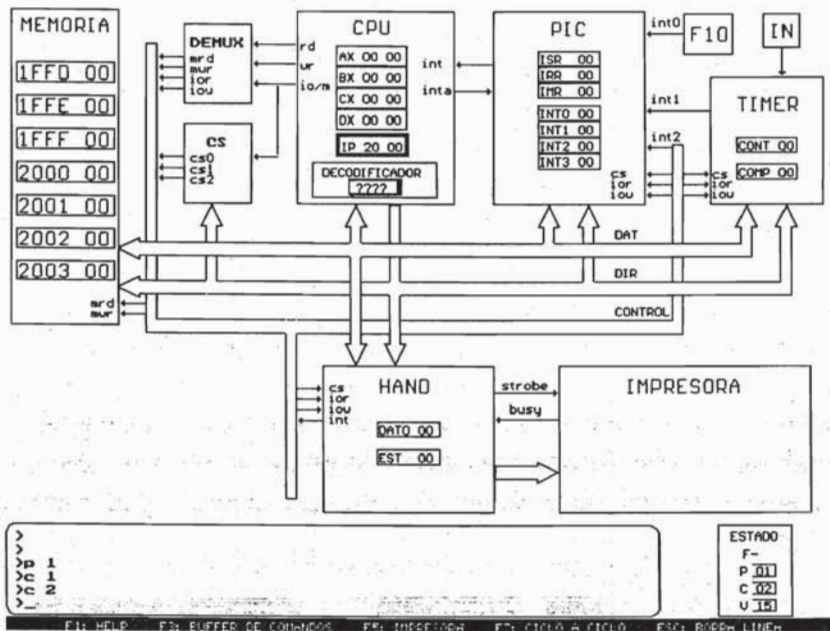


Figura 5. Conexionado 2 de la periferia del MSX88

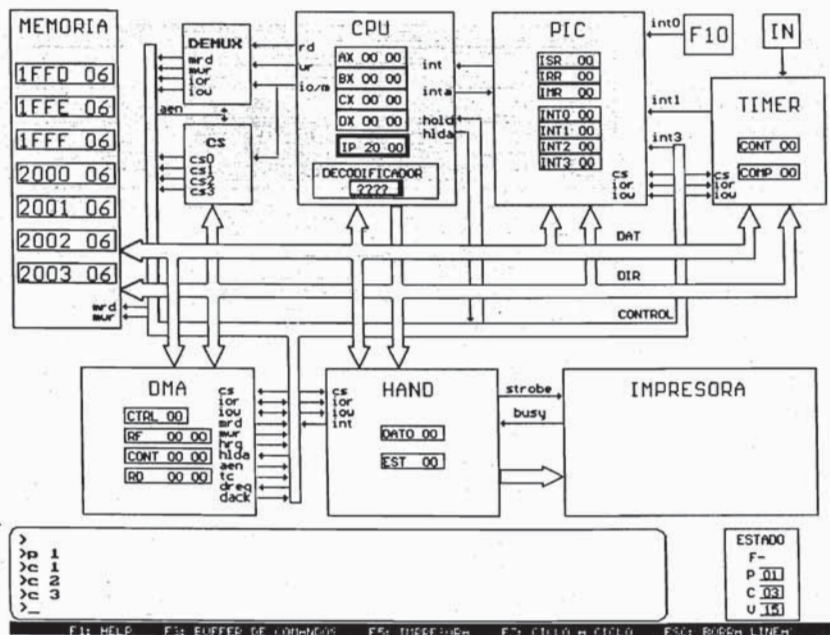


Figura 6. Conexionado 3 de la periferia del MSX88

3.3.1. PIO

Presenta dos puertos paralelo programables bit a bit. Posée cuatro registros internos, dos de control y dos de datos. Tiene dos posibles conexiones dependiendo del modo de conexionado en que se trabaje. (ver figuras 3 y 4).

3.3.2. Hand

Disponible en el modo de conexionado 2 (figura 5) conectado a la impresora a través de una interfaz *centronics*. No admite modo de programación alguno. El registro de estado se utiliza únicamente para conocer si está listo para transmitir otro dato.

3.3.3. Timer

Conectado a un reloj de 1Hz, básicamente se comporta como un contador de eventos. Posee dos registros de ocho bits, registro de cuenta y y registro de comparación. Ambos son de lectura escritura.

Cuando el valor del registro de cuenta alcanza el valor del registro de comparación, se activa la señal de salida del timer conectada a la entrada de interrupción *int1* del PIC.

3.3.4. Controlador de interrupciones

Similar al Intel 8259, proporciona la posibilidad de tener hasta 8 entradas de interrupción de las que en la configuración actual sólo se utilizan como máximo 3 a la vez.

Los registros permiten una total programación de las distintas características funcionales.

La conexión de las entradas de interrupción es la siguiente:

- **int0**: Tecla F10. Cada vez que se pulsa la tecla F10 se produce una interrupción.
- **int1**: Salida del timer.
- **int2**: Hand, indicando que está listo para aceptar un nuevo dato.
- **int3**: CDMA. Indica fin de transferencia.

3.3.5. Controlador de DMA

Inspirado en el Intel 8237, ofrece la posibilidad de realizar transferencias memoria-memoria o memoria-periférico y a la inversa.

Posee un sólo canal de DMA, pudiendo realizar las transferencias en modo bloque (una vez iniciada la transferencia, transmite datos sin parar hasta que se agota el bloque a transmitir) o por demanda del periférico al que está conectado.

3.3.6. Impresora

Impresora ASCII de 20 columnas, capaz de mostrar el contenido que se le envía. Posee una interfaz *centronics* gestionado únicamente por las líneas *strobe* y *busy*.

Es posible fijar el número de caracteres por segundo que imprime, poseyendo igualmente un buffer de caracteres. Aunque muy lenta, para que sean observables los efectos en los accesos de los ordenadores a periféricos de este tipo, el comportamiento es totalmente similar a una impresora real.

3.3.7. Leds y Microconmutadores

Se comportan como la típica barra de LEDs y microconmutadores. Los 8 microcomputadores numerados de 0 a 7 pueden tomar los valores '0' y '1'. Estos valores se cambian, cuando se está ejecutando un programa, pulsando las teclas '0' a '7'. Los LEDs, igualmente numerados de '0' a '7' se encienden cuando a su entrada tienen el valor '1' lógico.

3.4. EL PROGRAMA MONITOR DEL MSX88.

Como cualquier ordenador el MSX88 tiene un sistema operativo (S.O.) encargado de realizar diversas tareas. Obviamente en este caso el sistemas operativo es mínimo, y por ello lo denominaremos *monitor*.

Ofrece la posibilidad de ejecutar diversos comandos e incluso soporta algunas utilidades, equivalentes a las *llamadas al sistema* presentes en cualquier S.O., que permiten a un programa de usuario realizar operaciones de entrada/salida sobre el teclado y la pantalla de usuario.

Los comandos del monitor son los equivalentes a los de cualquier programa depurador, adaptados a la filosofía MSX88.

La figura 7 ilustra el aspecto de la pantalla principal tras teclear el comando *help*. En la parte central de la pantalla se abre una ventana que muestra todos los comandos existentes. Se puede obtener aún un nivel más de ayuda.

Existen además algunos comandos asociados a las teclas de función - por ejemplo el *help* se puede obtener también presionando la tecla F1 -. En la parte inferior de todas las pantallas se puede observar la presencia de una barra de información, en la que se indican las funcionalidades asociadas a las teclas. El contenido de esta barra de información cambia en función de si se está bajo el control del monitor o de un programa de usuario.

El monitor también da información -ventana de estado- de la pantalla en la que nos encontramos, del conexionado seleccionado, de la velocidad de ejecución y del modo de ejecución.

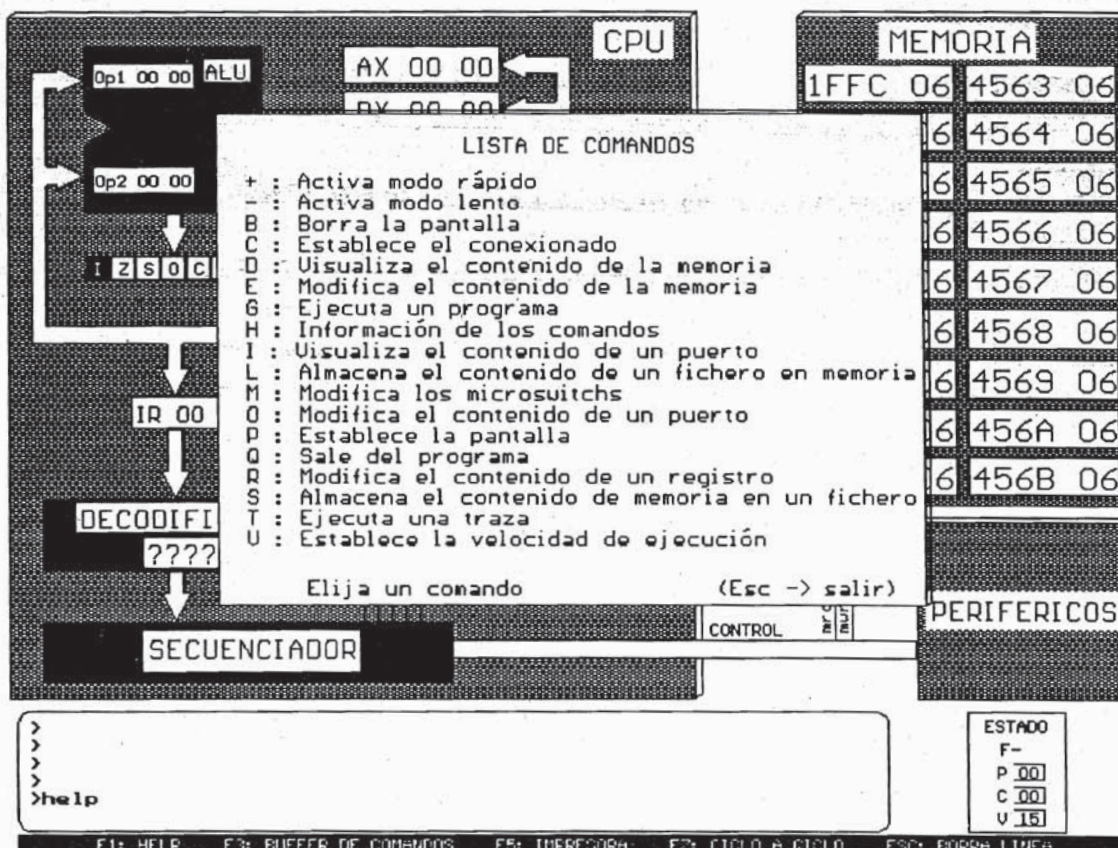


Figura 7: Aspecto del help con los comandos del monitor

3.4.1. Comandos relacionados con el modo de ejecución.

Como se ha comentado la característica que resume al MSX88 es la de 'hacer visible lo invisible'. De lo expuesto hasta el momento el MSX88 no es sino un emulador de un ordenador sin característica alguna que lo diferencie de cualquier depurador gráfico. Estas diferencias aparecen en el momento que se habla de los distintos modos en los que el MSX88 puede ejecutar los programas: *modo rápido* y *modo lento*.

En el *modo rápido* (comando +) el comportamiento del MSX88 no es diferente al de un depurador gráfico: es capaz de cargar programas contenidos en un fichero, modificar el contenido de los registros y de las posiciones de memoria, ejecutar programas con puntos de ruptura, instrucción a instrucción...

Cuando se elige el *modo lento* (comando -) el MSX88 empieza a mostrar sus cualidades didácticas diferenciadoras. A partir de este momento es como si observáramos lo que ocurre dentro del ordenador desde una nueva perspectiva: el MSX88 empieza a mostrar los flujos de información de los buses y señales existentes (figuras 8 y 9). Los buses y líneas se van rellenando, de forma continua, con colores distintos a los que tienen cuando no están activados, siendo posible variar la velocidad con la que se pintan, entre 32 valores posibles. -0: más lenta, 31: más rápida-

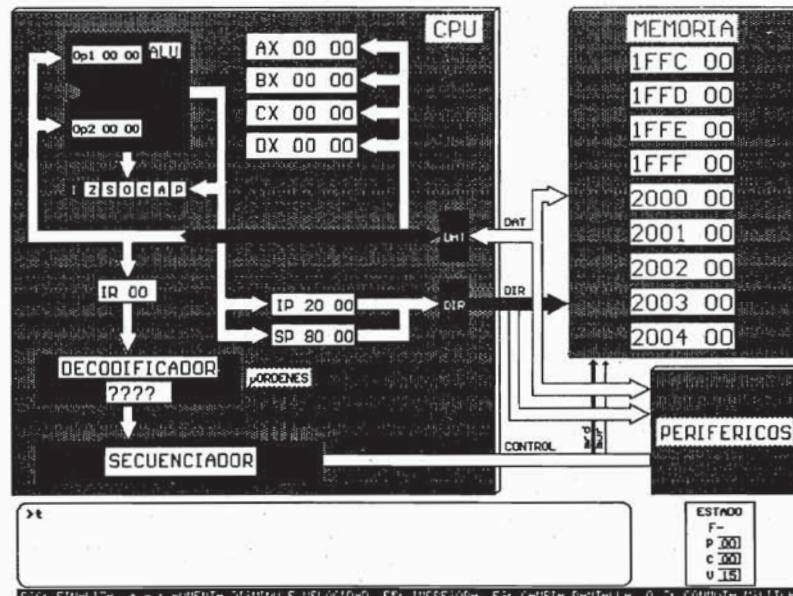


Figura 8: Ejecución en modo lento. Pantalla principal

Cuando un alumno cambia de la pantalla principal a la de la periferia, obviamente cambian los objetivos de estudio, y con ellos los elementos y señales que interesa visualizar. Así, en la pantalla de periferia, aunque se pueden ejecutar programas en modo lento, se ofrecen otros posibles modos de visualización donde se filtra de diferentes formas la información ofrecida en modo lento, para permitir centrarse únicamente en el estudio de las señales que interesen. El monitor ofrece los siguientes comandos para variar los modos de visualización:

- **+F:** El ciclo de búsqueda de la instrucción queda reducido a un único acceso a memoria. La representación que se hace en los buses es equivalente a que todas las instrucciones sean de un único octeto.

- **+P:** Se visualizan únicamente aquellas señales relacionadas con la periferia y las escrituras y lecturas de los puertos, junto con el contenido de los registros y posiciones de memoria.
- **+L:** Se visualizan únicamente las líneas de control asociadas a los periféricos y el contenido de los registros y posiciones de memoria.
- **+R:** Se visualiza únicamente la variación del contenido de los registros y posiciones de memoria.

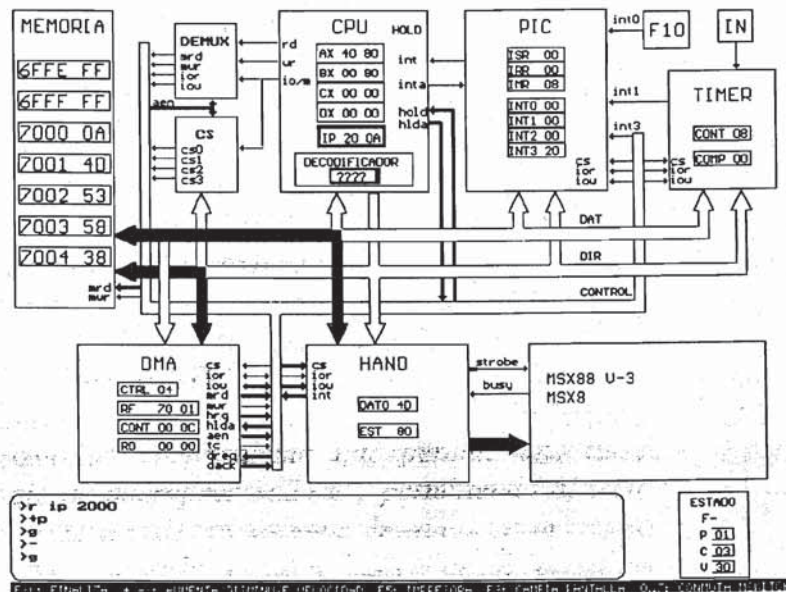


Figura 9: Ejecución en modo-lento. Pantalla de Periferia

4. CONCLUSIONES Y TRABAJOS PRESENTES Y FUTUROS

Hasta el momento el MSX88 se ha mostrado como una eficaz herramienta para abordar el estudio de la estructura y funcionamiento de los ordenadores. El efecto producido tanto en profesores como en alumnos que han trabajado sobre la pantalla principal, única disponible en versiones anteriores, ha sido valorado como altamente favorable.

La utilización de la periferia ha sido también probada parcialmente durante el curso 93/94 en la asignatura Arquitecturas de Ordenadores I para ilustrar el tema de Acceso Directo a Memoria, siendo igualmente evaluados los resultados como favorables.

Pese a que por su estructura el MSX88 puede cubrir, sin necesidad de acudir a otras máquinas, los contenidos de cualquier asignatura básica de Arquitectura de Ordenadores o Microprocesadores, es aconsejable usarlo como introducción a arquitecturas reales las cuales no tienen por qué ser de la familia iAPX86.

El desarrollo del MSX88 no se ha detenido. A parte de en mejorar todas aquellas partes con un funcionamiento no óptimo, actualmente se están realizando trabajos en las siguientes áreas:

- Elaboración de un manual de usuario del conjunto de herramientas y de un manual de prácticas.

- Dotarlo de una ayuda más potente, de forma que se pueda obtener fácilmente información referente al código máquina de las instrucciones, mapeado y estructura y funcionamiento de los periféricos, hipotética estructura del decodificador y secuenciador...
- Definir una hipotética evolución temporal de los distintos ciclos máquina y señales definidos en la actualidad. El objetivo es ofrecer una representación gráfica temporal de los diversos eventos ocurridos en el ordenador, similar a la de un analizador lógico.
- Creación de un Sistema de Autor basado en el MSX88 y de un Lenguaje de Autor [9] con el que el profesor pueda generar presentaciones para las clases de teoría y además generar Unidades Didácticas de autoaprendizaje.

Los tres primeros puntos estarán listos hacia finales de 1994, el cuarto se estima estará finalizado en el plazo de año y medio.

En un futuro está previsto abordar los siguientes temas:

- Ofrecer la posibilidad de que el alumno tras rediseñar el código máquina de las distintas instrucciones existentes, pueda modificar el decodificador del MSX88 para que éste entienda el nuevo código.
- Apoyado parcialmente en el punto anterior, ofrecer posibilidades de ofertar un simulador con arquitectura basada en la familias de Motorola, National...

5. REFERENCIAS

- [1] Intel. "The 8086 Family User's Manual". Intel Corporation 1980.
- [2] Intel. "iAPX 86/88, 186/188 User's Manual, Programmer's Reference". Intel Corporation 1985.
- [3] Intel. "iAPX 86/88, 186/188 User's Manual, Hardware Reference". Intel Corporation 1985.
- [4] M.A. Rodríguez-Rosello. "8088-8086, 8087 : programación ensamblador en entorno MS DOS". Anaya Multimedia. Madrid, 1987
- [5] J. Uffenbeck. "The 8086/8088 family: design, programing, and interfacing". Prentice-Hall International London, 1987
- [6] J.M. Angulo Usátegui. " Microprocesadores 8086, 80286, 80386". Paraninfo, 1990.
- [7] J. Carracedo y E. Pastor. "Ordenadores y Microprocesadores Parte I". E.U.I.T. de Telecomunicación. Madrid, 1980.
- [8] J. Carracedo, M.A. Nuñez y E. Pastor. "MICROSIMPLEX: a didactic microcomputer specially designed for computer education". *Euromicro 83. Microprocessing and Microprograming*. 1983.
- [9] J. Carracedo, C. Ramos & R. de Diego. "Author System for creating educational programmes". *Second biennial meeting of the Community of Mediterranean Universities on Microcomputers and their applications*. Malta, October 1988. Education and Application of Computer Technology.