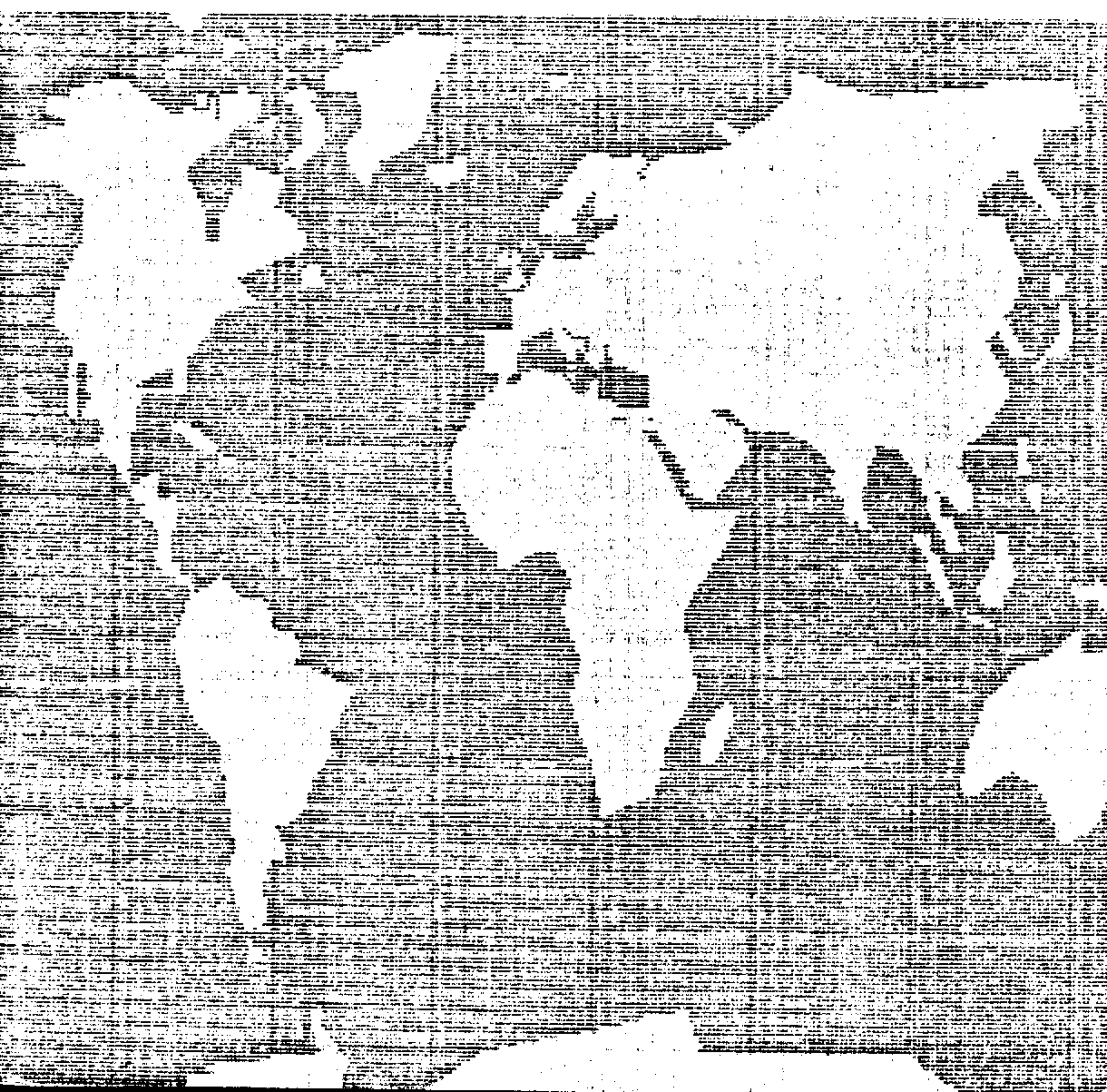


VOLUMEN I NO. 5 MAYO 1986





INFORMACION SOBRE EL CLUB

La integración en la asociación C.E.I.Q.L. se hace por suscripción anual o semestral. El C.E.I.Q.L. publica mensualmente el boletín de los socios, QLave, nombre por el cual también se conoce al Club.

Más información sobre la asociación se puede obtener desde la secretaria del Club.

Para ser miembro de QLave se requiere estar interesado por el ordenador personal Sinclair QL.

El Club mantiene una librería de Software. Una lista de los programas existentes en la librería se publicará de vez en cuando para así ir actualizándola. Los programas que se quieran aportar o sacar de la librería se deben notificar directamente al encargado de la misma.

Las contribuciones a QLave enviadlas a nombre del Presidente.

Presidente

Serafin Olcoz

Baltasar Gracián 21

5005 ZARAGOZA

Secretario

Juan Palacio

Pedro M. Ric, 19

5008 ZARAGOZA

Librero

Angel Asin

San Jorge 22, 3B

5001 ZARAGOZA

La correspondencia debe enviarse a: QLave / Apartdo correos 403 / 50000 - Zaragoza

Contribuciones a QLave

Las contribuciones a QLave deben ser Archivos de QUILL en cartuchos de microdrive, preferiblemente. Los cartuchos se devuelven a vuelta de correo.

Los programas cuya extensión no sea muy grande se incluirán en la revista, pero aquellos de gran extensión pasarán a formar parte de la librería.

Los programas que enviéis deben adjuntar una descripción de los mismos y de su funcionamiento.

=====

Se prohíbe la reproducción total o parcial del contenido de esta revista

QLave no se hace responsable del contenido de los artículos o comentarios que aparezcan firmados por sus correspondientes autores.

Sinclair, QL, QDOS, ZX microdrive son marcas registradas de Sinclair Research Ltd.



EDITORIAL

Como ya es casi habitual en las editoriales vuelvo a deciros que no ha habido ninguna noticia del Gobierno Civil sobre los estatutos.

Siguiendo con las malas noticias y como ya declamos en un flash de última hora en el pasado número de Qlave, la revista mensual QL USER se ha incorporada a QL WORLD. La empresa que publicaba QL USER era ENAP Publications y la que la ha comprado ha sido Focus Investments Ltd.

Y la noticia bomba para los seguidores de Sinclair en general y de este boletín en particular :

La compra de SINCLAIR RESEARCH Ltd. por la competencia : ANSTRAD , de la que damos amplia información en un artículo dedicado a ello exclusivamente.

En INVESTRONICA se confirmaron esta noticia el mes pasado así como desmintieron el rumor de que fuese INDESCOMP quien se encargase de la importación, para España, de los productos Sinclair a partir de ahora, por lo menos por un año (según dijeron aunque ...).

Item más no hay ninguna intención por parte de INVESTRONICA de importar nada en absoluto de hardware para el QL y quizá importen software aunque es poco probable. También se dijeron que las conversaciones con QJUMP Ltd. son agua de borrajas, desgraciadamente para los usuarios del QL españoles.

Sin embargo, INVESTRONICA junto con los colegios de arquitectos ha promocionado el QL y el software desarrollado para el mismo con fines de : Redacción de proyectos, Cálculo de estructuras e instalaciones para edificación, en Madrid, Euzkadi, Navarra, La Rioja, Burgos, y en próximamente en varios lugares más.

Debo nuevamente pedir a todos los socios que me enviéis colaboraciones para publicarlas en próximos números de Qlave , a ser posible en ficheros de QVILL en cartuchos o en discos de 3.5", ya que son y deben ser la clave de Qlave, ¡ ánimo que no decaiga el interés !.



COMENTARIO DE PROGRAMAS

PROGRAMA : 3L Microdrive Toolkit

EDITOR : COMPWARE

DIRECCION : 57 Repton Drive, Haslington, Crewe, CW1 1SA

PRECIO : 9.99 £

Antes que nada debo aclarar que Toolkit significa herramienta y que por eso hay muchos comandos toolkit en el mercado, no confundir el mdv Toolkit de Compware con el de Sinclair / @Juzp.

Una vez aclarado lo referente al nombre, decir que la versión que se comenta es la 1.00 (12 de Septiembre de 1985).

El programa se presenta en un microdrive junto con dos folios (ROJOS) de instrucciones, presentados en formato cuartilla tipo @Lave.

Los programas que contiene el microdrive son:

- boot - Un simple programa que carga el Toolkit
- mkit_bin - El código objeto del Toolkit
- backup - Un programa para hacer copias de seguridad
- read_headers - Un programa ejemplo para leer cabeceras
- read_to_screen - Ejemplo de la lectura de un sector en la memoria de la pantalla
- verify_n_read - Ejemplo de leer un sector y verificarlo
- write_a_sector - Ejemplo de escribir un sector

El contenido de este cartucho permite realizar el "formateo" de un cartucho con un específico número aleatorio en la cabecera, leer la cabecera de un sector y devolver en la pantalla el nombre del medio que se ha leído y el número variable de la cabecera, leer un sector absoluto, verificar sectores absolutos, escribir sectores absolutos, y escribir sectores absolutos sin verificación.

En las instrucciones se dan unas claras explicaciones de como realizar las prestaciones arriba indicadas, así como una referencia muy útil de como es el formato de los microdrives (sectores, cabeceras de sectores, mapa de sectores, archivos, el directorio, y localización de un sector).

El único comando que no tiene un ejemplo en el microdrive de su empleo, es el encargado del "formateo" con número prefijado pero dicho ejemplo se describe perfectamente en el manual. No se como no se ha incluido con el resto de los ejemplos ya que es realmente útil y es tan pequeño el listado necesario.

Como se puede ver por el contenido de este programa se ve claramente que no se trata de una herramienta destinada precisamente para los principiantes ya que son necesarios algunos conocimientos sobre el formato de los microdrives así como sobre la forma de almacenar los ficheros en los cartuchos para poder sacarle el jugo a estas utilidades.

Realmente se trata de un TOOLKIT muy útil y con muy buenas aplicaciones sobre todo teniendo en cuenta que a todas las utilidades se accede desde el SUPERBASIC.

El único problema, que no se debe a Comware, es el posible mal uso de este toolkit entendiéndose por mal uso el dedicado a la desprotección de programas comerciales ya que algunos presentan únicamente la protección del número aleatorio y del nombre del medio que van en la cabecera del sector.

Que más decir excepto que todas las funciones que aparecen en el toolkit son del todo necesarias para cualquier usuario con algunos conocimientos y con interés en internarse en su BL.

PROGRAMA: @MON

EDITOR: @JUMP

DIRECCION: 24 KING STREET RAMPTON CAMBS CB4 4BD ENGLAND

Este es un programa desarrollado por Tony Tebby (creador del @DOS) y por lo tanto íntimamente ligado con el @DOS en su forma de trabajar.

El paquete consta de cuatro programas principalmente @MON que es el programa principal y que sirve para la corrección de errores de los programas en ensamblador; @MIN una versión reducida del anterior; JOB que es un fichero de extensiones del Superbasic para control de jobs desde éste y CLOCKS que es lógicamente un reloj.

Los comandos de que consta el programa se pueden agrupar de la siguiente forma: un grupo dedicado a la ejecución y trace del programa objeto, así con GO podremos ejecutar un programa hasta un determinado lugar que hayamos marcado con un BREAKPOINT que como su nombre indica es un punto de ruptura en el que el control retorna al usuario esta ejecución se puede realizar desde una determinada dirección, hasta un breakpoint en otra, hasta un breakpoint o una condición especificada o hasta return.

El comando TRACE posee la capacidad de mostrarnos los efectos de cada instrucción en código objeto sobre los registros del micro posee las mismas variantes que las indicadas para el comando GO (breakpoint, condición, return...) si bien el breakpoint que reconoce este comando es distinto del de GO por lo que no hay conflicto entre ambos. Existe una modalidad de éste comando, @WICK trace que hace eso una pasada a través del código. La ejecución el TRACE se puede hacer en dos niveles, sólo el código en modo usuario o también el código en supervisor.

Estas opciones de trace sólo se pueden ejecutar hacia adelante lógicamente así en el caso de que deseemos ver los anteriores se ha preparado un buffer especial que se invoca con el comando RS, recall, así podemos acceder a los pasos anteriores sin volver sobre nuestros pasos.

Otro grupo de comandos se refieren al display en pantalla tanto en Hexadecimal y ASCII como mostrando las instrucciones desensambladas o los registros del micro.

Un tercer bloque de comandos tratan la modificación de registros y memoria así como la edición de ésta última, destacar de esta opción la facilidad con que nos movemos 'sobre la memoria' pudiendo alterar su contenido con gran facilidad.

Tenemos así mismo los comandos de búsqueda de cadenas de símbolos o bytes e incluso cadenas en una instrucción. Comandos de apertura y cierre de canales, así como de evaluación de una dirección en hex y decimal. Es posible también crear macro-instrucciones que se definen mediante CS y que posee también diversos modos de ejecución.

El programa se carga en memoria mediante LRUN MDVI_QMON y no comienza su ejecución hasta que es invocado tecleando QMON. Ver que el job puede ser llamado bien desde el Superbasic o bien desde otro job con lo que QMON se convierte en un job 'hijo' de éste último con todas las implicaciones que esto conlleva (desaparecer si lo hace el job que lo invocó ...). La llamada se puede hacer de un modo simple o no así es posible teclear QMON simplemente con lo que se entra en la ventana 0 y sobre el Superbasic por defecto QMON R2,3 monitoriza el job 3 en la ventana 2, QMON CON_256x70a0x0,1 trabaja sobre el job 0 en una ventana de reducidas dimensiones, QMON mdvi_clocks sitúa este programa para ser analizado en la ventana 1 y por fin QMON 0mdvi_clocks coloca mdvi_clocks de forma transitoria en la ventana 0 que retorna a su contenido original tras la ejecución del comando. Existen 5 líneas de ventanas transitorias que pueden ser usadas en cualquier momento.

Cuando se invoca el QMON éste redirecciona el vector de excepciones del micro de forma que el QDOS pasa el control a QMON cuando éstas ocurren, en el caso de que el job sobre el que se trabaje tenga un vector de excepciones, normalmente se copia por QMON de esta forma en el caso de producirse un error fatal se nos advierte con un mensaje por pantalla. En cualquier momento se puede interrumpir la ejecución de un comando pulsando la tecla ESC se reanuda la ejecución con ENTER.

El proceso que se sigue en la inclusión de los BREAKPOINT es ligeramente distinto para el comando GO que para el comando TRACE, así con GO se produce la sustitución de dos bytes de código objeto por la orden no legal 4AFB en el caso de TRACE el código no es alterado pero en cada momento se mira la lista de BREAKPOINT.

El fichero JOBS se incluye como ayuda para conocer el identificador del job que deseamos analizar en un determinado momento y coloca los siguientes procedimientos residentes en el Q.L.: AJOB que permite la activación de un determinado job, RJOB que libera al QDOS de un determinado job, SPJOB permite colocar a un job con una prioridad distinta a la que posea, JOBS que nos da la lista de jobs que existen en este momento en el Q.L.

Junto con el programa se incluye un manual bastante completo y que por regla general soluciona la mayor parte de las dudas que puedan surgir.

DISEÑO DE UNA SENCILLA TARJETA DE ENTRADAS Y SALIDAS PARA EL "QL"

=====

J.M. Angulo, J. Parra y K. Bañuelos

INTRODUCCION

Con objeto de potenciar al microprocesador QL y poderlo destinar a diversas aplicaciones de control industrial, se ha diseñado en el Departamento de Arquitectura de Ordenadores de la Facultad de Informática de la Universidad de Deusto, una simple tarjeta, basada en un VIA 6522, para la adaptación de periféricos.

El VIA es un circuito integrado usado en las familias de los microprocesadores de 8 bits 68000 y 65000, que dispone de dos puertas, de 8 líneas cada una, para el acoplamiento de periféricos en paralelo. Además, posee una línea de entrada y salida de información en serie y dos contadores-temporizadores o "timer". Se trata de un módulo de entrada y salida de carácter síncrono, o sea, que su funcionamiento requiere una señal de reloj, que recibe por la patita que lleva la denominación de E.

MODULOS DE ENTRADA Y SALIDA SINCRONOS Y ASINCRONOS

La familia del 68000, entre cuyas UCP está la 68000, residente en el QL, está diseñada para controlar módulos de entrada-salida asíncronos. De esta manera se alcanza una gran velocidad con la transferencia de información, puesto que la UCP espera a la finalización del trabajo del módulo, el cual comunica dicho acontecimiento, activando la línea DTACK y colocándola a nivel bajo. DTACK significa "reconocimiento de la transferencia de datos" (Data Transfer Acknowledge).

Los componentes de control de periféricos de la familia 68000, como el 68230 y el 68901, están preparados para trabajar en modo asíncrono, con un incremento notable de la velocidad, potencia y flexibilidad. Sin embargo, hoy por hoy, estas pastillas son caras y de manejo complejo, motivos por los que los fabricantes del 68000 implantaron un bus síncrono a la UCP principal para poder manipular módulos de las familias de microprocesadores de 8 bits, que a veces son hasta 100 veces más baratas y su manejo está muy divulgado.

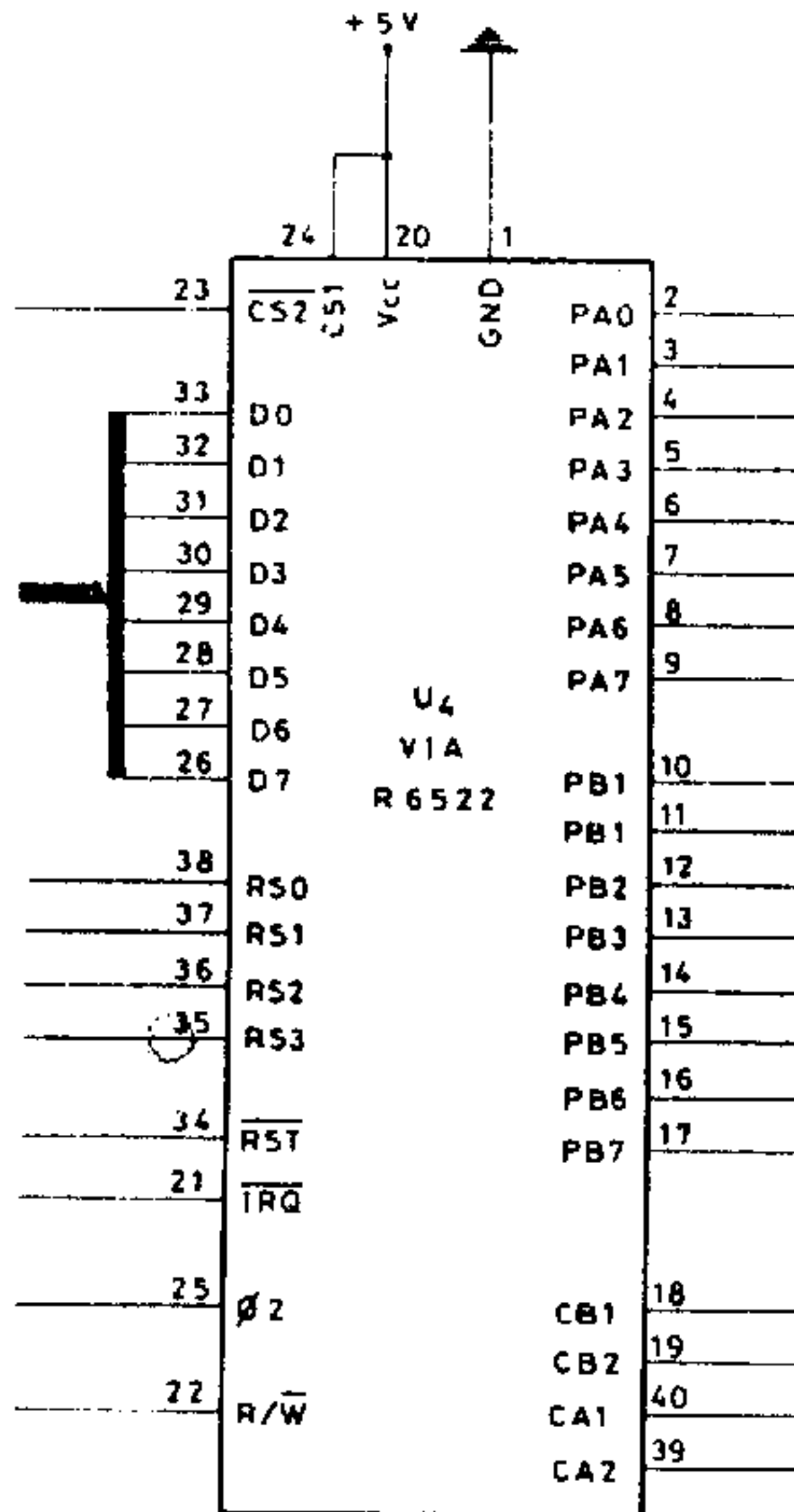


Figura 1. Estructura del VIA, que se trata de un modulo de entradas y salidas sincrono, usado en las familias 68000 y 65000.

ESQUEMA DEL CONEXIONADO Y FUNCIONAMIENTO DE LA TARJETA

Para que el 68000 cambie a modo de trabajo sincrono, necesita recibir un nivel activo por la línea VPA (Dirección válida para periférico). La circuiteria de la tarjeta a diseñar tendrá que originar dicha señal cada vez que le corresponda funcionar.

El 68000 genera una señal VMA (Dirección válida de memoria), que no genera el 68008, por lo que también la circuiteria de la tarjeta ha de crearla mediante dos flip-flops y se aplica a cada una de las patitas de "selección de chip" del VIA (CS1). Este circuito es el indicado por Motorola en el manual "16-bit Microprocessor Data Manual".

La señal de reloj, proviene de la patita E (Enable) de la UCP y su frecuencia es un décimo de la del 68008. Un ciclo de E está formado por 10 ciclos del 68008, seis con nivel bajo y cuatro con alto.

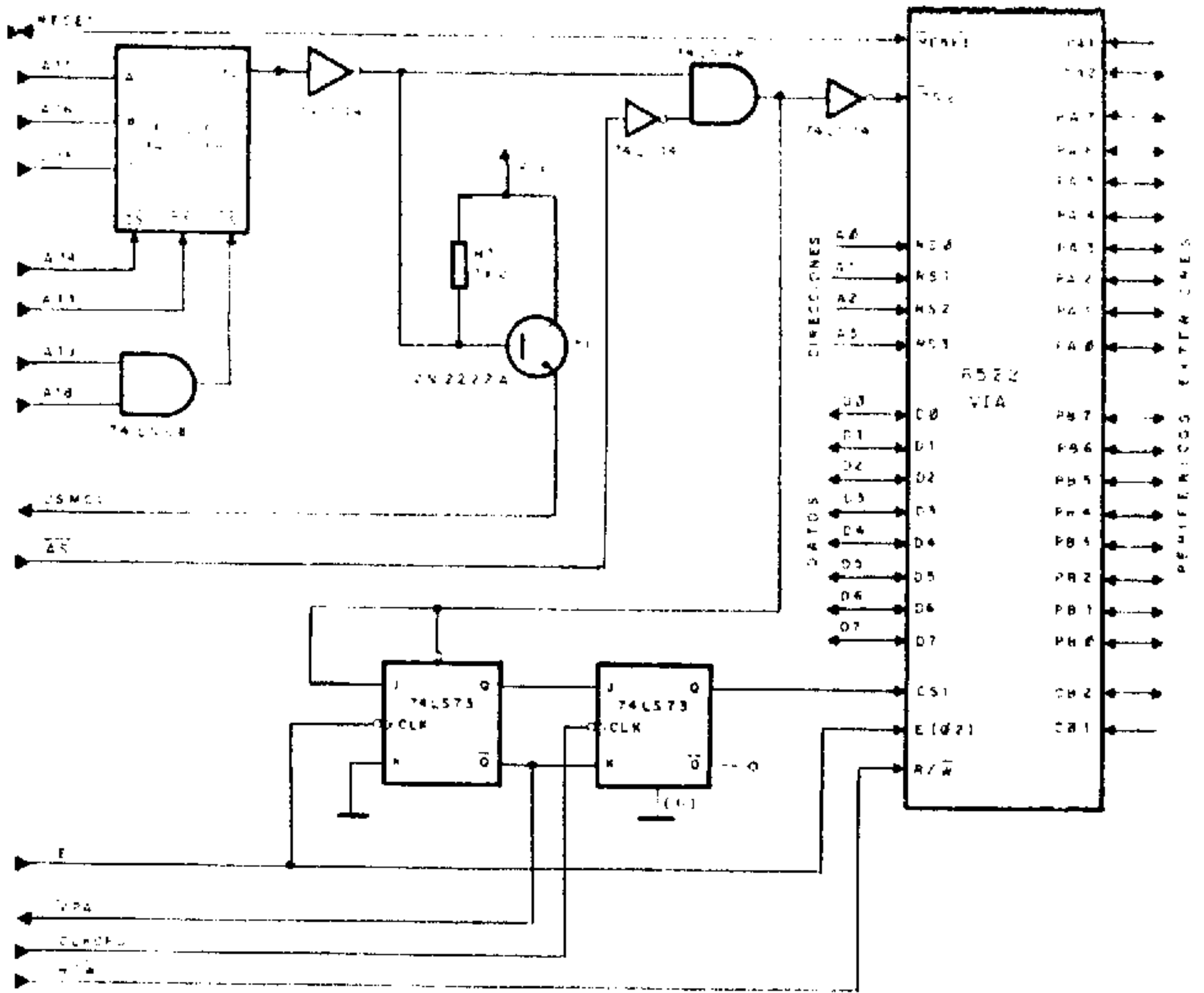


Figura 2. Esquema de conexionado de la tarjeta que adopte un VIA al QL.

Un problema particular que presenta el microordenador QL consiste en la presencia de una ULA de video, de diseño propio de SINCLAIR y que soporta diversas funciones. Una de ellas es confirmar un DTACK cuando su lógica interna establece que es necesario. Esta forma de trabajo dificulta enormemente el funcionamiento de la tarjeta, puesto que DTACK no debe activarse durante el misao; por lo cual cuando ésta recibe una dirección propia, manda una señal DSMCL, que es particular del QL y que inhabilita la ULA. Esta señal ha de ser muy rápida e interna, razones por las que se ha empleado un transistor en su generación.

Las fases de funcionamiento del VIA de la tarjeta, en un ciclo de transferencia de datos serían las siguientes:

- 1.- EL procesador 68000 comienza un ciclo normal de lectura escritura esperando la activación de la señal DTACK.
- 2.- En su lugar se recibe la señal VPA, generado por la circuitería de la tarjeta VIA

- 3.- La UCP vigila la señal E hasta que pase a baja.
- 4.- La circuitería de la tarjeta genera la señal VMA.
- 5.- El periférico espera a que E suba el nivel lógico y entonces transfiere los datos
- 6.- La UCP espera a que baje de nuevo E, lo que significa el fin de la transferencia.
- 7.- El procesador desactiva AS y DS.
- 8.- La circuitería de la tarjeta desactiva VMA y VPA.
- 9.- Comienza el ciclo siguiente.

Tanto la programación como el manejo del VIA se realiza normalmente. La base del direccionamiento del VIA corresponde a la posición \$C0000, que se cargará en uno de los registros de direcciones del 68000, lo que simplificará el manejo de los registros internos del VIA, a base de desplazamientos sobre dicha base.

Figura 3. Mapa de la memoria del 68.
La base de direcciones del VIA es \$C0000

FFFFF

C0000

40000

28000

20000

1C000

18000

10000

0C000

00000

EXPANSION I/O
EXPANSION RAM
RAM INSTALADA
RAM DE VIDEO
EXPANSION I/O
MEMORIA I/O
EXPANSION I/O
ROM OPCIONAL
ROM INSTALADA

BIBLIOGRAFIA

- 1.- "Microprocesadores de 16 bits. El 68000 y el 8026/8099".
Angulo, J.M., Paraninfo.
- 2.- "Microprocesadores. Diseño práctico de sistemas".
Angulo, J.M., Paraninfo.
- 3.- "16-bits Microprocessors Data Manual" Motorola.
- 4.- "Microprocesadores de 32 bits. El gran salto".
Angulo, J.M. Paraninfo.

INTRODUCCION AL CODIGO MAQUINA



INSTRUCCIONES DEL 68000 (II)

INSTRUCCIONES ARITMETICAS (Cont.)

-NEG

Resta el operando destino de cero. El resultado se almacena en el destino. De esta forma se obtiene el complemento a 2 del destino (v. Clave n.º. 1).

Admite longitudes de operando de 8,16,32 bits, y puede indicarse utilizando los modos de direccionamiento:

INSTRUCCION NEG

Modos posibles	op. fuente	op. destino
Reg. datos directo		x
Reg. direc. directo		
Reg. direc. indirecto		x
Reg. indirecto con postincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado.		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl		
Relat. cont. programa indexado		
Inmediato		

Ej: NEG.B D5

El ejemplo calcula el complemento a 2 del byte de menos peso de D5, y lo almacena también en el mismo byte de menos peso de D5.

Los flags del registro de estado se alteran igual que con la instrucción SUB.

-NEGX

Resta entre el contenido del operando destino y el valor del flag X de cero. El resultado se almacena en destino.

Los flags del registro de estado se alteran igual que con la instrucción SUBX.

Es igual que NEG, salvo que el contenido del flag X también se resta de 0, lo que sirve para trabajar con precisión múltiple (Z pasa a 0 si el resultado es 0, pero en caso contrario no cambia de estado, lo que permite comprobar el resultado en operaciones de múltiple precisión).

Puede trabajar con operandos de 8, 16 y 32 bits, y admite los mismos modos de direccionamiento que NEG.

-NBCD

Esta instrucción resta el operando destino y el bit X de 0, utilizando aritmética BCD (v. Glave n.º 1). Almacena el resultado en el destino. Lo que se consigue con NBCD es el complemento a 10 del destino cuando X=0 y el complemento a 9 si X=1.

Permite los mismos modos de direccionamiento que la instrucción NEG.

El tamaño de operando es siempre 1 byte.

--Alteraciones de los flags.

'C' y 'X' se ponen a '1' si se producen llevadas en la operación, y a 0 en caso contrario.

'Z' Se pone a '0' si el resultado no es cero, y si lo es NO CAMBIA

'N' y 'V' Quedan indefinidos.

-MULU

Usando aritmética binaria, multiplica dos números sin signo.

El tamaño de los operandos es de 16 bits, y el del resultado, de 32.

Los flags 'C' y 'X' están siempre a cero, independientemente de lo que ocurra en la operación; 'N' refleja el bit más significativo del resultado; 'Z' pasa a '1' si el resultado es 0 y a '0' en caso contrario; 'X' no se altera.

Uno de los números a multiplicar debe facilitarse en un registro de datos, y hará a su vez de operando destino; el otro operando puede facilitarse usando:

INSTRUCCION MULLU

Modos posibles	op. fuente	op. destino
Reg. datos directo	X	X
Reg. direc. directo		
Reg. direc. indirecto	X	
Reg. indirecto con postincr.	X	
Reg. indirecto con predecr.	X	
Reg. indirecto con desplaz.	X	
Reg. indirecto indexado.	X	
Absoluto corto	X	
Absoluto largo	X	
Relat. cont. programa con despl	X	
Relat. cont. programa indexado	X	
Inmediato	X	

-MULS

Multiplica también dos operandos de 16 bits, produciendo un resultado de 32 que se almacena en el operando destino. Es pues idéntica a MULLU, salvo que MULS trabaja con números con signo usando aritmética binaria en complemento a 2 (v. Qlave ndm 1).

Los modos de direccionamiento posibles son los mismos que para MULLU.

--Alteraciones de los flags

- 'N' Se pone a '1' si el resultado es negativo; se pone a '0' si es positivo.
- 'Z' Pasa a nivel '1' al producirse como resultado 0; en caso contrario pasa a '0'.
- 'V' y 'C' Permanecen siempre a 0.
- 'X' No se altera.

-DIVU

Divide los 32 bits de un registro de datos entre un operando fuente de 16 bits. El resultado se almacena en el registro de datos destino (el cociente en los 16 bits de menos peso y el resto en los de más peso).

Los modos de direccionamiento posibles son los mismos que para la instrucción MULLU.

--Alteración de los flags

- 'N' Repite el bit más significativo del cociente. Si se produce un desbordamiento queda en estado desconocido.
- 'Z' Se pone a '1' si el cociente es 0 y a '0' si el cociente no es 0. Si se produce un desbordamiento queda en estado desconocido.
- 'C' Permanece a 0
- 'X' No se afecta.
- 'V' Se pone a '1' al producirse un desbordamiento.

(Se produce desbordamiento cuando el operando fuente es mayor que el destino. El desbordamiento se detecta antes de realizar la operación, y en caso de originarse, no se lleva a cabo la operación y por tanto los operandos no cambian. Los flags se alteran según se ha expuesto cuando se produce desbordamiento)

-DIVS

Opera igual que la instrucción DIVU, salvo que la operación se lleva a cabo en aritmética sin signo.

Los modos de direccionamiento posibles son los mismos que los permitidos en la instrucción MULU.

Los 32 bits del resultado están compuestos de igual forma que en la instrucción DIVU (los 16 de menos peso forman el cociente, y los 16 de más peso el resto), y se almacenan también en el registro de datos destino (el que proporciona el dividendo).

--Alteración de los flags.

'N' Pasa a '1' si el cociente es negativo y en caso contrario pasa a '0'

Si se produce desbordamiento queda en estado indefinido.

'Z' Se pone a '1' si el cociente es 0 y a '0' si el cociente no es cero.

Si se produce desbordamiento queda en estado indefinido.

'V' Al producirse un desbordamiento pasa a nivel '1'

'C' Siempre a '0'

'X' No se altera.

-EXT

Produce la extensión de signo (repetición del bit de más peso), del operando, pasándolo de byte a palabra o de palabra a palabra larga, según sea el caso.

Ej: EXT.W D2

En el ejemplo se extienden en signo los 8 primeros bits de D2, hasta completar la palabra (16 primeros bits), por funcionar con una longitud de una palabra (W). Si el tamaño del operando es de palabra larga, el bit 15 del registro de datos implicado, se extiende hasta el 31.

El operando tiene que estar contenido siempre en un registro de datos.

--Alteración de los flags.

'N' Se pone a '1' si el resultado es un número negativo (el bit a extender = '1'); en caso contrario pasa a '0'.

'Z' Pasa a nivel '1' si el resultado es 0 y viceversa.

'V' y 'C' siempre a 0

'X' no se altera.

-CLR

Borra el operando especificado (lo llena de ceros). El tamaño del operando puede ser de 8, 16 o 32 bits.

--Alteración de los flags.

'Z' Pasa siempre a '1'

'N', 'N' y 'C' pasan siempre a 0.

'X' no se altera.

Los modos de direccionamiento posibles para indicar el operando que queremos borrar son:

INSTRUCCION CLR

Modos posibles	op. fuente	op. destino
Reg. datos directo		X
Reg. direc. directo		
Reg. direc. indirecto		X
Reg. indirecto con postincr.		X
Reg. indirecto con predecr.		X
Reg. indirecto con desplaz.		X
Reg. indirecto indexado.		X
Absoluto corto		X
Absoluto largo		X
Relat. cont. programa con despi.		
Relat. cont. programa indexado		
Inmediato		

-CMP

Esta operación resta dos datos, reflejando el resultado de la resta en los flags del registro de estado. Dicho con otras palabras, lo que hace es comparar los dos operandos. El operando destino es siempre un reg. de datos y hace las veces de minuendo.

Los modos de direccionamiento posibles son:

INSTRUCCION CMP

Modos posibles	op. fuente	op. destino
Reg. datos directo	X	X
Reg. direc. directo	X	

Reg. direc. indirecto	
Reg. indirecto con postincr.	X
Reg. indirecto con predecr.	X
Reg. indirecto con desplaz.	X
Reg. indirecto indexado.	X
Absoluto corto	X
Absoluto largo	X
Relat. cont. programa con despl.	X
Relat. cont. programa indexado	X
Inmediato	X

Los flags del registro de estado se alteran de la misma forma que en la instrucción CMP.

La diferencia entre CMPA y CMP estriba en que CMPA sólo puede usar tamaños de operando de 16 y 32 bits, mientras que CMP podía hacerlo con operandos de 8, 26 y 32.

-CMPI

Variante de la instrucción CMP que resta del operando destino (expresado con direccionamiento inmediato en la misma instrucción) el operando fuente, reflejando el resultado de dicha operación en los flags de estado. Dicho de otro modo, compara un dato inmediato con otro almacenado en un registro de datos o en la memoria.

Los modos de direccionamiento posibles son:

INSTRUCCION CMPI			
Modos posibles	op. fuente	op. destino	
Reg. datos directo			X
Reg. direc. directo			
Reg. direc. indirecto			X
Reg. indirecto con postincr.			X
Reg. indirecto con predecr.			X
Reg. indirecto con desplaz.			X
Reg. indirecto indexado.			X
Absoluto corto			X
Absoluto largo			X
Relat. cont. programa con despl.			
Relat. cont. programa indexado			
Inmediato	X		

El tamaño de operando puede ser byte, palabra o palabra larga.

Los flags reflejan el resultado de la comparación de igual modo que en la instrucción CMP.

En la siguiente tabla aparecen los valores de los flags del registro de estado, según haya sido el resultado de la resta:

CONDICIONES	F L A G S		
	N	Z	C
Destino > Fuente	0	0	0
Destino < Fuente	1	0	1
Destino = Fuente	0	1	0

-CMPA

Es un caso particular de la instrucción CMP. En CMPA, el registro destino es un registro de direcciones. Al igual que en CMP, también se lleva a cabo la resta entre los dos operandos y se alteran los flags, de acuerdo con el resultado.

Así pues los modos de direccionamiento permitidos son:

INSTRUCCION	CMPA	
	Modos posibles	op. fuente op. destino
Reg. datos directo		X
Reg. direc. directo		X X
Reg. direc. indirecto		X
Reg. indirecto con postincr.		X
Reg. indirecto con predecr.		X
Reg. indirecto con desplaz.		X
Reg. indirecto indexado.		X
Absoluto corto		X
Absoluto largo		X
Relat. cont. programa con despi.		X
Relat. cont. programa indexado		X
Inmediato		X

CORREO DE LOS SOCIOS



Por si a alguien pudiera interesarle, a continuación van un par de posiciones de memoria bastante útiles al encabezar un programa:

PEEK (163890) = 1 si se está en T.V. y 0 si se está en monitor.

PEEK (163892) = 0 si se está en mode 0 y 8 si se está en mode 8.

Enrique José Sanchis Borrel
ALICANTE

Cometi un bug en el artículo sobre WHEN (Número 3, pag. 26) dice:

b) b=x-1 : b=SQRT(x-1)

debería decir:

b) b=x-1 : b=SQRT(B).

José M. Guzmán
SEVILLA

Cuando se sale de los programas de Psion y se intenta editar algún programa de basic, en mi versión de programas y de Q.L. no se imprime la línea a editar. Todavía no tengo muy claras las razones pero la solución es sencilla: en los programas boot eliminar en la línea en que se abren los canales 0, 1 y 2 la apertura del canal 0.

Si se borra esto todo funciona igual y además se editan las líneas de los programas Basic.

E. de Jesús

Respecto a la posibilidad de conocer la memoria libre en el Q.L. esto se calcula por diferencia de dos variables del sistema SV_BAS y SV_FREE:

```
print peek_1(163856)-peek_1(163852)
```

9Lave



NOVEDADES

WDSsoftware

HILLTOP
 St. Mary
 JERSEY
 Channel Islands
 Tel: (0534)81392

Lista de precios de ABRIL	Peso	Precio R	Contenido
Item	Kgr.	(Sin IVA)	

HARDWARE			
Sinclair QL + Manual	7.000	173.87	Versión Británica
Sinclair QL (512 K)	7.000	282.61	QL + Silicon Express
Microvitec QL RGB Monitor T/S	15.000	239.13	BS-Col
CST Q-Disc Interface	.350	75.00	Controla 2 Drives
CUMANA CA070 Disc Interface	.350	69.48	Controla 4 Drives
CUMANA CS354 1 x d/s 3.5" drive	2.500	121.70	Fuente Propia, 720k
CUMANA CS400 1 x d/s 80 tr 5.25" dr	2.900	156.48	" " , 720K
CUMANA CD358 2 x d/s 3.5 " drive	3.500	210.00	" " , 1.4M
CUMANA CDB00/S 2 x d/s 5.25 " dr	3.900	295.61	" " , 1.4M
MIRACLE Sys 512 ExpanderRAM	.270	125.00	Doble Slot
CST Q+4 Expansion System	5.000	200.00	
CST IEEE-488 Interface		195.00	Para Standard(GPIB)Bus
CST 20Mb Winchester+dual 3.5" Disc		1000.00	Completo con Interface
QL-Centronics para Impresora	.250	26.04	
Miracle Sys Modaptor para Modems	.150	33.91	
Miracle Sys Joystick Adaptador	.030	4.99	
Eidersoft ICE	.350	52.13	ROM con WIMPs/multitarea/ RAMdisc/ART/Toolkit

SOFTWARE

Doctor Cartucho Talent	.080	19.09	Recupera archivos perdidos
GraphiQL Talent	.280	30.39	Paquete de Diseño Gráfico
QL Project Planner Triptych	.850	34.74	
QL Entrepreneur " "	.900	34.74	
QL Decision Maker " "	.900	34.74	
QL Contabilidad Integrada Sagesoft	1.100	78.22	
BCPL Metaconco	.550	52.13	
ISO Pascal Metaconco	.700	78.22	
Compilador C Metaconco	.750	86.91	
Psientific QL Keydefine	.040	8.65	Hasta 2000 carac. por Key
ARCHIVER Eldersoft	.200	16.48	
QSPELL " "	.150	17.35	
QL JARDINERO Gordian	.280	21.70	Datos sobre 1100 especies
Dialog Transact	.250	30.39	
Tasprint para QL Tasman	.250	17.35	
Tascopy " "	.250	11.22	
QL Toolkit Sinclair	.750	21.70	Extensido de Comandos Basic
QL Cajero QUEST	1.000	60.83	

JOSS | JOYSTICK - OPERATED SOFTWARE System |

Es el interface entre el Usuario y el QDOS, el potente sistema oprativo del QL. Como sistemas similares en máquinas más caras, JOSS libera al usuario del tedioso trabajo de escribir largos ficheros de comandos, o ambiguos iconos. Con las teclas del CURSOR o con JOYSTICK, se mueve el cursor sobre el comando requerido yse pulsa la barra de Espaciado o el Fuego para seleccionar la opción.

15 h en ndv o 5 1/4" flp, 17 h en 3.5 " flp

REFBL7

Para usar con ARCHIVE 2, contiene 1300 útiles referencias y un programa de busqueda/escritura.

11 h en ndv o 3.5" flp, 9 h en 5 1/4 flp

W3 TUTOR de MORSE

6 h en ndv o 5 1/4 ", 8 h en 3.5 " .

El pago se debe hacer por adelantado en £ Esterlinas en Bancos Británicos, Giro Internacional, Postal Order o ACCESS/MasterCard. Añadir 1£ para el correo aéreo fuera de Europa.

(El IVA no se cobra de ello se encarga la Aduana).

Añadir 4.65 £ si no sobrepasa 1/2 Kg, añadir por 1/2 kg .55, si no pasa el kg 4.60, si no pasa 2 kg 5.70, 3 kg 6.75, 4 kg 7.70 ,5 kg 8.30, 6 kg 9.60, 7 kg 10.30, 8 kg 10.75, 9 kg 11.20, 10 kg 11.65.



SOFTWARE PARA QL

STAT.- El mas completo y versátil paquete de programas de cálculo estadístico. Tres modos distintos de introducción de datos. Capaz de efectuar análisis seriados, con un alto nivel de automatización, de matrices de datos con múltiples observaciones por cada sujeto, y con parte de sus valores en blanco. Estadística de dos variables, con listados, resultados y representaciones gráficas (histogramas en 3D y gráfico de nube de puntos) con salida por pantalla y por impresora. Incluye programas-herramienta para modificar y combinar datos, conversión de unidades directas a derivadas (típicas, percentiles) con representación gráfica, por pantalla o impresora. Correlación parcial y múltiple (con 3 y 5 variables). Con un amplio manual que incluye explicación de los términos y técnicas estadísticas y con programas-ejemplo que ilustran sobre el modo de empleo. PVP 6.000 pts.

QLTEST.- Un fértil instrumento pedagógico que permite construir fácilmente pruebas objetivas de elección múltiple, con 3 a 7 respuestas posibles, para enseñanza individualizada o para editar rápidamente pruebas de rendimiento individuales o colectivas, de corrección rápida y objetiva. Con amplia documentación y varios programas ejemplo. PVP 3.500 pts.

CATALOGO.- Si sus grabaciones en cartuchos de microdrive comienzan a ser difíciles de localizar, Vd. necesita este programa herramienta que le permitirá tener ordenados y fácilmente localizables sus programas en BASIC, ficheros PSION y demás grabaciones. Con salida por pantalla e impresora, crea relaciones ordenadas y sistemáticas de grabaciones: listados comprimidos del contenido de cada cartucho, relación general de grabaciones por orden alfabético, con expresión del cartucho e que está cada una, y relación alfabética de ficheros PSION. Indica % de capacidad utilizada respecto de la total de los cartuchos disponibles; crea una base de datos que permite la búsqueda rápida de grabaciones a cargo del propio ordenador. Con manual de instrucciones. PVP 4.000 pts.



PREGUNTAS

Refiriendome concretamente a los capitulos que sobre código máquina han aparecido en Qlave resulta:

El primer capítulo (n. 1 sin página) , está bien, perfectamente digerible.

Donde ya se ila el asunto es en el segundo capítulo (n. 2 página 18), concretamente el esquema de dicha página que no está desarrollado.

A partir de aquí ya todo lo que se publique no sirve para nada puesto que no sabes que hacer con ello.

Concretamente el lector no experimentado que es para quien se supone que va dirigido el tema se pierde y lo deja.

A mi me gustaría que me aclararan lo siguiente:

1- ¿Cuales son las instrucciones del QL para manejar el código Máquina?, ¿Como se entra?,¿Como se sale?,¿Como se ejecuta?. ¿Son las instrucciones del QDOS?. ¿Son instrucciones de Superbasic?. Las explicaciones deberían ir acompañadas de algún ejemplillo.

2- Explicar como se fabrica un instrucción en código máquina y como se la programa. A continuación ejemplo de algún programilla con alguna tarea sencilla.

3- No queda claro según lo que se dice en la citada página 18, si hay que introducir los datos en binario, hexadecimal o decimal.

Os agradeceré resolváis estas dudas en la revista, pues creo que no seré el unico que las tenga.

José Aramendi Lizcano

MADRID

Las instrucciones que configuran el lenguaje máquina son las reflejadas en las tablas que aparecen en el número 2. En dichas tablas figura el nombre de la instrucción, significado, sintaxis que admite y tamaño de operandos con que puede funcionar.

Como ya se explica en el número 2, para llevar a cabo un programa en código máquina, es precisa la ayuda de un programa ensamblador. La manera de llevarlo a cabo es completamente distinta a cómo lo haríamos en Basic. En primer lugar, hay que confeccionar el listado fuente del programa (listado con las instrucciones que se están explicando a partir del número 4, dispuestas para realizar la tarea, fin del programa. Como decíamos en el número 1, además de las instrucciones del 68000, también pueden aparecer llamadas al QDOS --traps--). El listado fuente se lleva a cabo con la ayuda de un programa editor, que se comporta únicamente como un editor de textos. La sintaxis, respecto a columna de comentarios, separadores, depende en gran parte del programa ensamblador que utilicemos, y por tanto hay que remitirse al manual del mismo. Una vez elaborado el listado fuente con la ayuda de un editor, se graba, y se pasa a través de un programa ensamblador. Dicho ensamblador producirá un fichero que es por fin el programa en código máquina listo para ejecutarse en el QL. Este proceso se refleja en la fig.1, pag. 18, n.º 2.

Respecto al repertorio de instrucciones, éstas no tienen nada que ver con el Superbasic. Cada microprocesador tiene un juego de instrucciones. En el caso del 68000 éstas son las reflejadas en las tablas del número 2 de Qlave. Aunque parezca excesivamente tedioso, programar a nivel máquina es esto exactamente, hablarle directamente al microprocesador del ordenador.

El desarrollo de cada instrucción se empieza a exponer a partir del número 4.

Los datos que aparecen en el listado fuente (el que llevamos a cabo nosotros), pueden facilitarse en uno u otro sistema de numeración, según el programa ensamblador que utilicemos. Hay algunos que permiten usar cualquiera de los tres sistemas (indicando, por supuesto, el sistema que queremos usar).

Así por ejemplo, la instrucción `ADD.W $45000,D3` quiere decir: Sumar los 16 bits ("W" indica un tamaño de operando de 16 bits, como ya se ha explicado) situados en la dirección de memoria `$45000` --hexadecimal-- (para la mayoría de los lenguajes ensambladores, el símbolo \$ delante de un número indica que éste está en hexadecimal), con los contenidos en el registro D3. `$45000` indica una dirección de memoria, ya que no se indica ningún otro tipo de direccionamiento, y entonces se sobreentiende que utilizamos direccionamiento absoluto (V. Qlave n.º 3).

No obstante, somos conscientes de que el lenguaje ensamblador, o el código máquina, resultan muy arduos para aquellas personas que no han oído hablar nunca de ellos. Los artículos que aparecen en el boletín, están enfocados, desde el supuesto de que se conocen básicamente los fundamentos sobre lenguajes ensambladores y estructura de microprocesadores.

Tal y como aconsejamos en el n.º 1, resulta útil el practicar con un lenguaje ensamblador, y a partir de unos conocimientos básicos sobre el tema (los que intentamos dar en la sección) ir descubriendo las posibilidades del mismo.

Agradecerla si me pudiesen informar sobre la existencia de un compilador de COBOL.

Sin más que felicitarles por la extraordinaria labor que están llevando a cabo y esperando no se desanimen y que esto dure, me despido de Vds.

Angel Trigueros Romero
CASTELLÓN

Lamentablemente todavía no hay en el mercado ningún compilador de COBOL, que nosotros sepamos.

Por lo visto varios de nuestros socios sienten interés por trabajar en este lenguaje pero por ahora no es posible.

Aunque en el boletín de Marzo hay una interesantísima información sobre los productos que comercializa "Compware", no estoy aun seguro de algunos detalles que quisiera saber antes de hacer mi pedido. Son los siguientes:

1.- ¿Las expansiones de memoria "Miracle Expanderam" permiten insertar el interfaz/controlador de disco de "MicroPeripherals" - que es lo que yo tengo - o bien solo sirven para el de "Cumana"?

2.- ¿Sabes que es el "Expansión Module (Sinclair QL) power supply" ?

3.- ¿Las fundas vacías de microdrives sirven acaso para "tapar" y proteger las ranuras de inserción de los mismos en el QL, o son otra cosa?

4.- En la página 20 del citado boletín de Marzo cuando indicáis lo que hay que pagar por gastos de envío para diversos artículos se hace referencia a "modems", sin embargo no hay referencia similar en el "catálogo" de Compware reproducido en la página 17. Mi pregunta es: ¿resulta posible, por tanto, importar modems acústicos para el QL a través de Compware y cuánto valen?

En otro orden de cuestiones me atrevo a sugerir-preguntar lo siguiente:

1.- ¿No sería interesante introducir una sección informativa sobre el hardware disponible-disponible en nuestra querida Inglaterra, claro- para el QL ?

2.- ¿No sería posible presionar a Investronica para que se ocupe algo más del hardware y del software "disponible" para el QL que tanto anuncia?

3.- ¿Tenéis algunas intenciones sobre redes locales o sobre conexión con modems entre los socios que quisieran? ¿Es esto posible?

4.- ¿No podríais publicar en el boletín los datos precisos para suscribirse a QL'users y a QLWorld ya que aquí -que yo sepa-nadie las importa? Antonio Ramos

Las expansiones de memoria "Miracle ExpanderAM" permiten insertar el interfaz de Micro Peripherals, comprobado en Zaragoza, no es propaganda.

El "Expansión Module power supply" se trata efectivamente de una fuente supletoria de energía, pero en concreto para la expansión de memoria de ExpanderAM no es necesaria (comprobado).

Las fundas vacías no sirven para lo que pregunta y no entiendo el sentido de la misma.

La respuesta es evidente después de ver la sección de novedades del pasado mes, pero respecto a lo publicado en la página 17 del número 3 hay que decir que se incluyen los modems y sus precios en la segunda línea.

En la sección de novedades tratamos de dar a conocer lo "disponible-disponible" para el QL y conste que aunque se puede entender como publicidad no lo es, ya que no la cobramos y la publicamos por la información que en ella se da .

No es posible presionar a INVESTRONICA y el motivo es que dicen que en España no hay demanda de hardware para el QL, aunque en QLave pensemos lo contrario.

Redes locales ya hay varias funcionando aunque no para la comunicación entre usuarios. Respecto a la instalación de los modems, depende de la CTNE y de cómo se regularicen y normalicen los modems en España, más información en próximos números de QLave .

La forma de suscribirse a QL WORLD/incorporating QL USER es enviando un cheque por valor de 30 £ a nombre de FOCUS INVESTMENTS Ltd. cuya dirección es :

OAKFIELD HOUSE PERRYMOUNT ROAD HAYWARDS HEATH WEST SUSSEX RH16 3DH

Tel. 0444 459188

Tengo una impresora BMC, la DP100 de matriz de puntos de impacto, y al trabajar con los programas Archive y Quill de QL (Psion), tengo problemas:

1.- En el QL Archive:

Para imprimir los caracteres ã y ñ, de tal forma que impriman ese carácter y no otro. Que procedimiento puedo utilizar para que el ordenador al detectar el carácter " lo imprimiera.

2.- En el QL Quill:

Cuando se agota la memoria del microdrive y estamos trabajando con un documento, cómo podemos continuar el mismo documento. Cómo imprimir los caracteres ã y ñ.

3.- En el QL Easel:

Forma de hacer gráficos en forma de pirámide.

Carlos Sánchez Plaza
Madrid.

La impresora Admate DP100, posee varios juegos con caracteres locales (U.S.A, Francia, Alemania, Inglaterra, Dinamarca, Suiza, Italia y España). Si la impresora tiene activado el juego español, varían los caracteres de códigos: 35, 91, 92, 93, 123, 124 ("ã" es el carácter 92 y "ñ" el 124).

Lo más seguro es que tu impresora tenga por defecto activado el juego español; en este caso, normalmente puedes imprimir el carácter "ã", pero no puedes hacerlo desde "Quill". Para solucionarlo, tienes que ejecutar el programa "Install_bas", que está en el mismo cartucho que "Quill". Hecho esto, selecciona la opción "EDITAR", con lo que te aparecerán todos los códigos de control y posibilidades que usa "Quill" para el manejo de la impresora. Pues bien, basta con borrar los códigos que aparecen en "COD. PREAMB.", (el programa dejará la anotación "NINGUNO"). Una vez hecho esto, selecciona la opción instalar, y el mismo programa se encargará del resto. En realidad lo que has hecho es evitar que cada vez que "Quill" vaya a imprimir, seleccione en la impresora el juego U.S.A.

Es posible que no tengas seleccionado por defecto el juego castellano; en este caso, debes cambiar los códigos que aparecen junto a "COD. PREAMB." por "ESC.R.7", que son los encargados de activar el juego castellano.

Para conseguir que también el programa Archive imprima la ñ, debemos hacer las mismas modificaciones con el programa "Install_bas" de "Quill", e "instalar" los datos en el cartucho de "Archive".

Al trabajar con el programa "Quill", éste va grabando automáticamente el texto que rebasa la memoria del QL, ocupando de este modo el cartucho que tengamos en el microdrive 2. Así pues, si hemos confeccionado un texto que ocupe más de la mitad de un microdrive, "Quill", tendrá abierto un fichero en "mdv2_" para poder trabajar con ese texto. Si en estas condiciones le decimos que nos lo grabe; como la opción por defecto del comando salvar funciona en el microdrive 2, nos encontraremos con que no tenemos suficiente espacio libre en el cartucho. En este caso basta con alterar la opción que nos ofrece por defecto "Quill" al ir a salvar: "mdv2_" por "mdv1_", y meter en el microdrive 1 un cartucho formateado. De esta forma podremos trabajar con textos de hasta 100 Kbytes (aprox.).

Lógicamente para cargar un texto que ocupe más de la mitad de un cartucho, tenemos que hacerlo desde el microdrive 1, ya que si no al ir a abrir "Quill" su fichero de trabajo en el microdrive 2 no encontraría espacio suficiente.

Para trabajar con textos que superen los 100 Kbytes, es preciso hacerlo por partes no superiores a dicha cifra.

De todas formas no es recomendable trabajar con textos muy largos, ya que el programa tiene que acceder constantemente al microdrive 2, ocasionando unas pérdidas de tiempo y un desgaste del cartucho evitables.

Resulta más práctico dividir los textos en fragmentos cortos y grabarlos por separado. Para imprimir el texto basta con unirlos todos. Además proporciona mayor seguridad, ya que en el caso de perder información en un cartucho, podríamos llegar a perder todo el texto si no lo tuvieramos troceado.

El carácter " en tu impresora es el carácter de código 91 del juego francés. Para poder imprimirlo, el QL debes activar el juego de caracteres francés y luego imprimir chr\$(91).

Print (canal), chr\$(27);"R";chr\$(1);chr\$(91).

Un sistema para conseguir imprimirlo sin mayores complicaciones puede ser el activar dicho juego en la impresora (chr\$(27);"R";chr\$(1)), y cada vez que queramos imprimirlo, teclear en su lugar el carácter de código 91 (abrir paréntesis cuadrado). En este caso tienes que resignarte a no poder imprimir la "ñ". Para poder imprimir ambos caracteres, tienes que cambiar al juego francés para imprimir " y al castellano (chr\$(27);"R";chr\$(7)) para imprimir la "ñ"

Las graficas de poblacion consisten en dos graficas, una a derecha y otra a izquierda (como en la figura), esto se puede realizar en el programa EASEL metiendo por separado las graficas de derecha y de izquierda, y convirtiendo una de ellas en negativa, mediante formulas, es decir, suponiendo que una grafica se llamase 'Izquierda', y que la otra se llamase 'Derecha', se pondria (tras haber introducido los valores de ambas graficas):

"Izquierda=-Izquierda"

Tras esto se visualiza la grafica conjunta por medio del comando 'view' (visualizar) introduciendo:

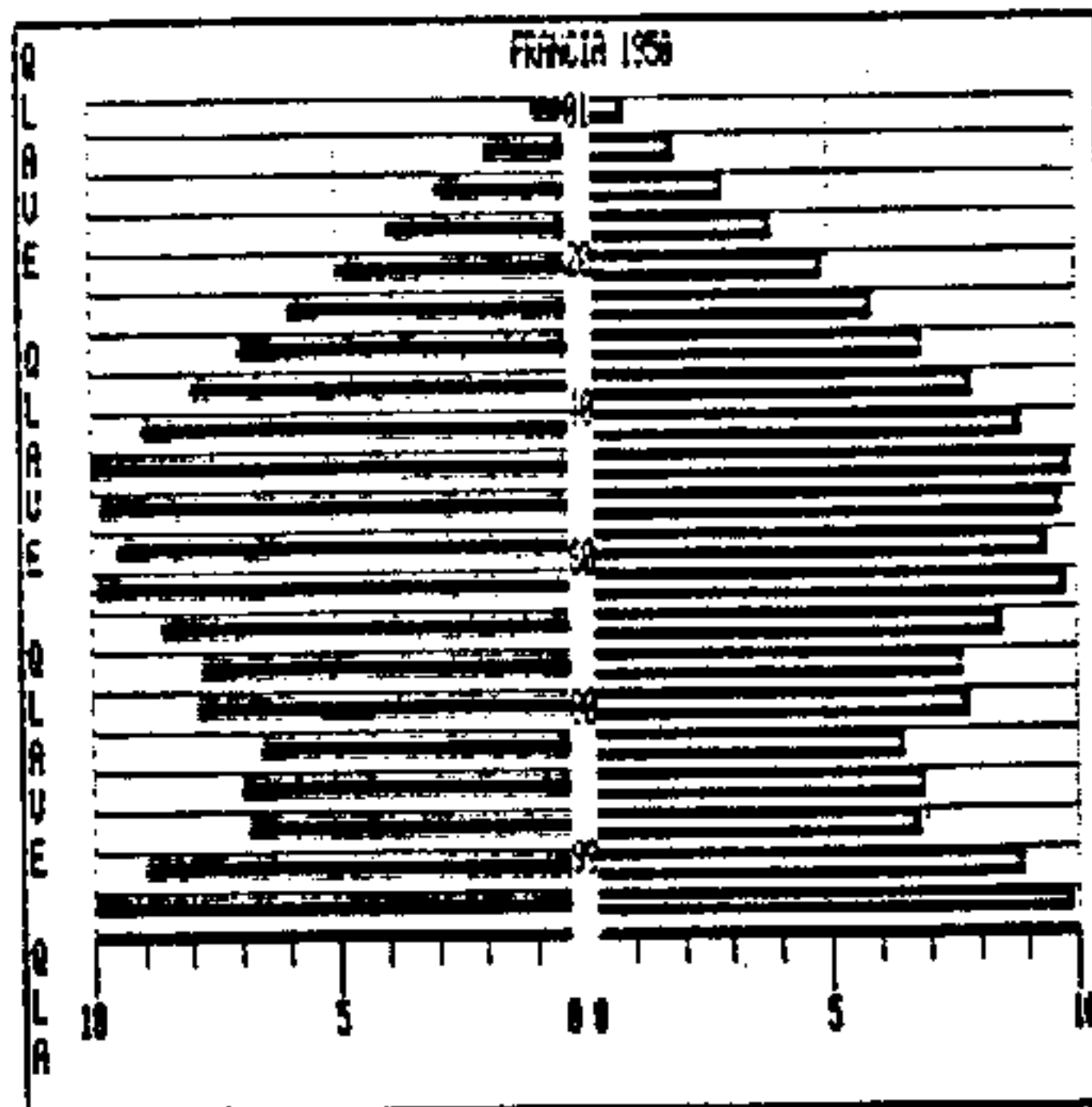
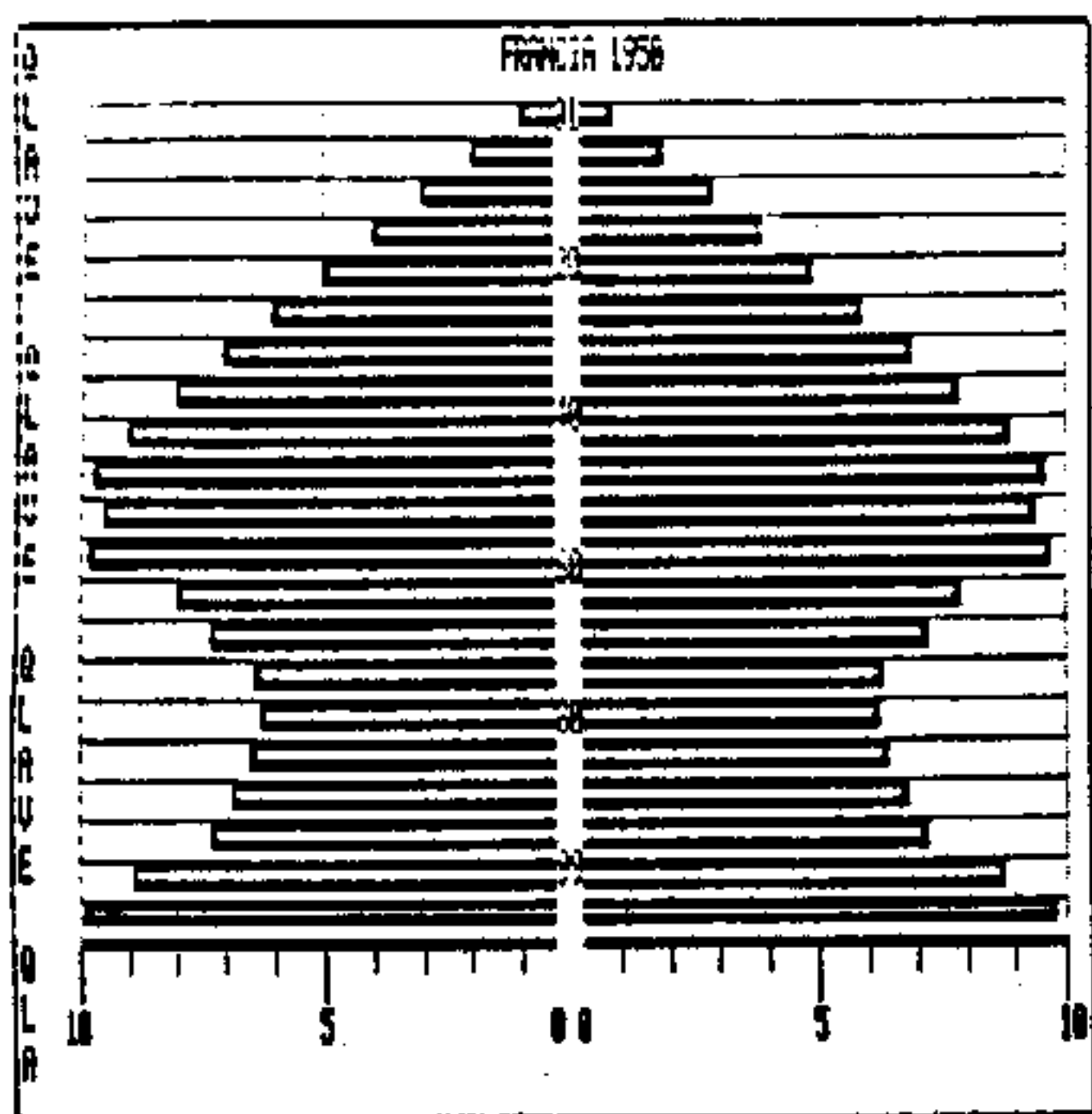
"view", "Izquierda, Derecha"

y dandole finalmente el formato numero 6.

Despues se elige una regleta sin valores y se meten como texto los tuyos propios.

Entre los dos graficos se puede colocar texto en modo vertical y sobre fondo negro para dar los valores de edades de la poblacion.

Las barras, ejes, regleta, fondo, indicador etc. se pueden regular despues a voluntad.



He podido comprobar que en el número 3 de Qlave aparece una oferta de Software y Hardware de la casa Comercial Comware. En la citada oferta se dan todos los datos para hacer el pedido: Precio, Gastos de envío, Demora aproximada del envío, etc. Pero he observado que no se dice nada acerca de los posibles problemas relacionados con el paso de la Aduana, ¿existen estos problemas? o estos productos tienen paso libre por la Aduana.

Ruego que me clarifiquéis esta cuestión ya que sería lamentable que un socio se gastara una gran cantidad de dinero en Hardware para que luego quedara retenido en la aduana al no venir nada indicado en la citada oferta.

Luis José García
JAEN

Hasta el momento los socios que han pedido algo a Comware lo han recibido al precio indicado y en el tiempo esperado. Al recoger el paquete en correos no ha surgido ningún problema excepto que el paquete estaba abierto pero con el contenido intacto y en perfectas condiciones.

He comprado una impresora Seikosha SP-800 con interface centronics acoplado al R.L. pero no consigo que funcione correctamente. Lo hace con los programas de PSION Quill, Easel y Abacus aunque falla con algunos caracteres castellanos pero no consigo que funcione con Archive cuando éste es el que más me interesa.

También desearía que me indicaseis el modo de obtener COPYS de pantalla.

Jose Miquel Alvarez Torregrosa

En el caso de que tengas un Q.L. castellano la obtención del conjunto de caracteres castellanos es sencillo, sólo has de incluir el comando TRAl. En el caso de que poseas el modelo inglés has de realizar correctamente el install de los programas de Psion, tienes una descripción completa del método a seguir en la sección de información del manual. Deberás de ejecutar el programa install_bas y seleccionar una impresora cualquiera en la que colocarás los datos de tu impresora, los baudios a 9600, suprimir el código si lo hubiere así como el de postámbulo, rellenando las opciones de negrilla... Para colocar los caracteres castellanos lo debes hacer con las opciones traducir (hay 10) colocando primero el carácter a transformar del set del Q.L. y a continuación el que corresponda en tu impresora (ñ..).

Una vez realizada esta instalación por ejemplo en el cartucho de Quill debes colocar en el mdv2 el Archive e instalar en el mdv2 en respuesta a las preguntas del programa. Para la obtención de COPYS de pantalla te remitimos al programa en ensamblador publicado en el boletín num. 4.

QLave.

- ¿Qué marcas y modelos de ampliaciones de memoria y floppis se conocen en el mercado?
- ¿Cuales tienen posibilidad de conectar el floppy a la ampliación?
- ¿Se puede conectar cualquier floppy a cualquier ampliación que disponga de conector?
- ¿Es posible encontrar en el mercado esp. estos productos?
- ¿Si se piden a U.K. cuanto habría que sumar al coste que allí tienen?
- ¿Existen ampliaciones compatibles con el floppy oficial presentado por INVESTRONICA ?
- ¿Cómo queda la conexión al interface?

José Manuel Nuñez Carrillo.

Te podemos decir que existen bastantes marcas distintas aunque las más conocidas son CST, PCML, Technology Research, Silicon Express, Cumana y Micro Peripherals. Las diferencias se suelen centrar más en la interface que llevan de conexión al Q.L. que en ellas mismas así por ejemplo la comercializada por Investronica dejaba bastante que desear si bien el Upgrade de Tony Tebby ha solucionado muchos de estos problemas. En U.K. la más votada fue la de Silicon Express. La mayoría de estas casas incorporan en versiones de sus interfaces ampliación de memoria, con lo que queda todo más recogido.

Para la conexión de ampliación en otros números se habla ya de estos temas (Expanderas que posee doble conector y que puede conectarse a los demás floppys, si bien el interface queda algo al aire.

En España apenas si hay disponible algo de esto y es preciso pedirlo a U.K. la tarifa a añadir depende de a quien lo pidas en nuestras páginas se indican algunas posibilidades.

QLave

SINCLAIR

AMSTRAD US

Amstrad compra Sinclair. Fue una noticia que nos inquietó a todos desde el momento que nos comunicaron que una emisora de radio la había dado. Nos extrañó mucho a todos y como la prensa "especializada" española no decía nada, pensamos que todo había sido un error.

Hace algunos días la noticia era firme e incluso Ricardo García Gete, director comercial de Investrónica confirmaba telefónicamente a nuestro presidente que la noticia era cierta, pero que Investrónica seguiría distribuyendo Sinclair por algún tiempo, debido a los acuerdos firmados con anterioridad a la operación.

Alan Michel Sugar compra la marca Sinclair en 5 millones de libras y los stocks existentes en 11 millones. Como primeras declaraciones Sugar dijo que los productos Sinclair se beneficiarían de la red de distribución global que posee Amstrad.

Esta red debe ser maravillosa dentro de Europa, pero realmente por Europa es muy fácil mover algo que se está fabricando aquí; por la parte europea Sinclair no gana nada, únicamente el desconcierto que de seguro se producirá con los cambios en las distribuciones. Tenemos que ver la distribución en los U.S.A. la "red global" de Amstrad en Estados Unidos; pues bien, Amstrad comienza a vender en el Nuevo Continente hace escasamente 3 meses. Desde el mes de febrero, en el que comenzó a servir sus dos "maravillas":

PCW 8256

CPC 6129

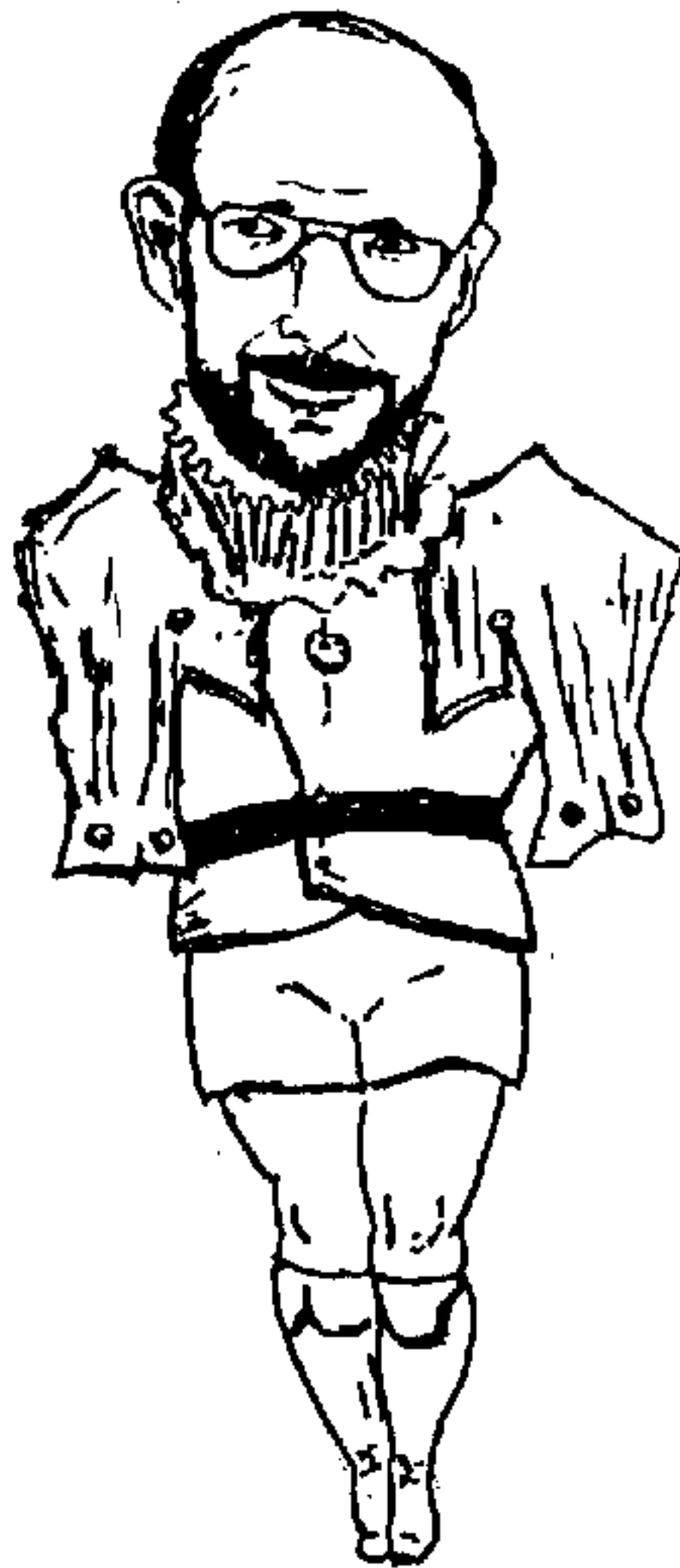
Sin embargo Sinclair se encuentra en Estados Unidos desde 1981 a través de su distribuidor Timex. Tampoco parece que gane gran cosa con el "contacto americano" de Amstrad.

Otra curiosa declaración del señor Sugar es que "posiblemente" desaparecerá la gama de productos Sinclair y que se lanzará para la campaña navideña un producto que complementaría "por debajo" la gama Amstrad.

Conocemos la gama Amstrad, pero no entendemos que puede haber debajo, realmente no se emplean microprocesadores de 2 bits ni de 4 bits; que es lo único que puede estar por debajo de dicha "gama".

También es curioso que Mr. Alan piense que él está fabricando ordenadores de empresa (es precisa la aclaración de que el modelo a que se refiere es el PCW 8256). Todos los ordenadores Amstrad llevan un Z-80, que es un buen microprocesador de 8 bits para regir 64 Kbytes de memoria, pero es muy malo para gestionar varios cientos de Kbytes; de hecho todas las compañías serias han abandonado este micro y están trabajando con micros de 16 (Comodore Amiga, Atari 520 ST, 1040 ST, Sinclair QL...)

Poco más cabe decir, porque de poco más nos hemos enterado, si cabe recordar el afán de Sir Clive por abandonar la gerencia de la empresa en Junio del año pasado, cuando dijo "yo no soy la persona adecuada para la gerencia".



Sir Clive no es un hombre de negocios, no entiende de dinero, sólo entiende de innovación tecnológica, de adelantarse a su tiempo, de apurar al máximo las posibilidades tecnológicas del momento. Así y por este camino nos sorprendió con el ZX80 a comienzos del año 80, con el ZX81 en marzo del 81, con el ZX Spectrum en abril del 82 y un poquito más cerca con el QL; "el Salto Cuántico".

También hay que recordar aquí la pantalla plana que incorporó en la televisión de bolsillo.

Los resultados económicos de Sinclair han sido verdaderamente extravagantes, ya que entre el 82 - 83 arroja un beneficio de más de 14 millones de libras, en el ejercicio siguiente la expansión fue muy escasa, pese al crecimiento de las ventas; fueron unos beneficios de algo menos de 14.3 millones de libras. El periodo marzo 84 y marzo 85 fue un verdadero record, Sinclair consiguió perder 18.3 millones de libras, algo verdaderamente portentoso, teniendo en cuenta la tendencia de beneficios de los años anteriores.

Sin embargo hemos de tener en cuenta que en este momento es cuando se producen fuertes gastos y los ingresos siguen siendo más o menos los mismos. En este tiempo se comienza a comercializar el C5, el coche eléctrico de Sinclair Vehicles; se crea la compañía independiente MSI para el estudio de la alta escala de integración, monta en la universidad e Strathclyde una red de RL (más o menos 1 por alumno), destinados a investigar en el campo de la inteligencia artificial. También en este periodo se está desarrollando el "Wafer-Drive", cuyo proyecto completo cuesta en torno a los 50 millones de libras. Con tantos gastos, la política de ventas de Sinclair tiene que ser muy agresiva, y consigue una participación en el mercado inglés de un 60%, pero ello no le logra sacar de la ruina económica; demasiados gastos en investigación.



Alan Michel Sugar no es un científico, no entiende de tecnología, sólo de dinero. Sugar es capaz de vender ruedas cuadradas, nada importa la calidad del producto, es necesario venderlo, y de eso sí que sabe.

Los ordenadores que vende Sugar son innovadores, o mejor, serían innovadores si nos encontrásemos en 1983, pero estamos en 1986 y no podemos decir que haya inventado nada, que ha hecho un "600", cuando la tecnología permite hacer un "Ferrari".

Ha tenido este hombre bastantes negocietes y todos ellos parece ser que le han ido bien, pero, desde luego, ninguno tan bien como el último.

Después de todas estas cosas nos queda, todavía, preguntas como:

- ¿Qué será del espíritu innovador de Sinclair después de la compra?
- ¿Será cierto que piensa "cargarse" don Alan Michel la gama Sinclair y ponerle un cassette pegado (como el 464) a alguno de los Spectrum?
- ¿Alguien se gastaría 11 millones de libras en stocks para acabar con un modelo?
- ¿Qué será de la cartera de proyectos en marcha que tenía Sinclair y sobre todo con el Wafer-Drive?
- ¿Podrá acabar el Señor Michel con la inquietud que ha producido 16 "ZX Microfairs", miles de programas, cientos de periféricos y millones de páginas impresas.?

Algo sí está claro, Sir Clive está en vida en la historia de la difusión informática, eso no podrá evitarlo el tío Michel.

De cualquier forma no podemos decir otra cosa que bienvenido sea don Michel "azúcar" si hace las cosas bien, lo cual es verdaderamente difícil, porque es algo que no le interesa. Sólo quiere vender.

FLASH . . . FLASH . . .

Leon Heller, presidente de QUANTA, preguntó a Alan Michel Sugar, presidente de Amstrad, acerca de cuales eran sus planes respecto al futuro del QL, y que si estaría dispuesto a vender los derechos de fabricación del QL a "cualquiera". Sugar contestó que "le haría feliz el poder venderlos", pero que por el momento aún no conocían la máquina, y que primero la iban a estudiar.

Leon Heller comenta, que aunque Amstrad deje de lado el QL, el QL seguirá fabricándose.



TRUCOS Y RUTINAS

Con el programa QUILL suministrado por PSION no es posible la realización de varias copias de un mismo documento como no sea usando una y otra vez la opción de impresión con lo que esto implica si deseamos obtener una cantidad respetable de copias.

Este problema es solucionable fácilmente si usamos la opción de impresión ofrecida por QUILL en lugar de dirigirla a impresora lo hacemos a un fichero de microdrive en esta situación se obtiene un fichero terminado con .lis que está preparado para volcarlo a impresora en el momento que lo deseemos así sería factible construir un programa en SuperBasic que controlase el volcado de este fichero a impresora tantas veces como se desee.

El fichero tratado así por QUILL incluye todos los códigos de control que tengamos en nuestro INSTALL salvo los caracteres castellanos. Los que posean el G.L. castellano solucionarían este problema incluyendo el comando TRAI.

Ahora bien en el caso de imprimir muchas veces un mismo documento supone una pérdida excesiva de tiempo para el usuario por lo que he hecho esta rutina que se encargaría de hacer todo lo anteriormente expuesto pero en multitarea con lo que podemos aprovechar así un tiempo de proceso que en otro caso se perdería.

Es posible imprimir no sólo ficheros de QUILL sino textos o listados tratados con otros editores como pueden serlo el de Metacomco o Sinclair. Los ficheros obtenidos de QUILL con la opción INPRIMIR EN FICHERO, que es el tipo aceptado por el programa, llevan una palabra al comienzo del fichero a modo de indicador esto se testea en la zona etiquetada como CICLO.

El programa permite la impresión de caracteres castellanos para ello se han construido dos tablas paralelas una correspondiente a los códigos correspondientes en el G.L. y otra correspondiente a los que se imprimirán, aparecen etiquetadas por ACENT y NACENT el proceso de comparación se realiza a partir de la zona etiquetada BIFURCAR donde se recorre la tabla para buscar si coincide el carácter en curso con los de ella y en cada caso se pasa a DOBLEA, DOBLEB, DOBLEC y DOBLED que es la que lleva el carácter a imprimir a DI.

Al comenzar se procede a la apertura de una serie de ventanas en pantalla la primera nos sirve exclusivamente para colocar el sombreado, las dos siguientes se refieren al nombre del programa y la salida de consola asociada al job. Si se produce un error en la apertura de ventanas el mensaje de error correspondiente se generará por el canal 0, se da la opción de corregir el error producido y tecleando F1 se reanudará la ejecución del programa.

Posteriormente se nos pregunta el fichero a imprimir en caso de cualquier error se produce un mensaje en la ventana del job, ver que en este caso se decidió no testear el dispositivo de entrada de forma que puede ser mdv, flp, ser2, net o pipe con lo que el usuario deberá de usar únicamente aquel medio que sea legítimo en cada momento.

Seguidamente se nos interroga acerca del número de copias a realizar y número de líneas por página de forma que debemos conocer este dato antes de llamar al programa.

Posteriormente se pregunta si se va a usar papel continuo o no, en el caso de que se responda que si deberemos indicar después el número de líneas entre páginas, ver que esta opción pudiera no ser necesaria si ya se indicó al formatear el texto con @UILL pero por si esto no se realizó se decidió incluir esto.

Posteriormente en la zona etiquetada como SIMP se procede a abrir la salida serie en el caso de que no sea posible por estar en uso es posible corregir el error desde Basic (cerrando el canal que tengamos abierto) y pulsando F1 se reinicia el proceso.

Al finalizar el proceso se nos indicará la posibilidad de imprimir otro fichero en caso de no desearlo se procede a cerrar todos los canales y a liberar al QDOS del job.

Ver que en el caso de querer abortar el programa se debe de pulsar ESC, esta tecla se la testea constantemente durante la impresión por lo que no se deberá usar en el desarrollo de los otros programas que se ejecuten en multitarea con este job (Superbasic) lo mismo se puede decir de F1 cuando el programa está testeando errores en la apertura de canales.

El proceso de testeo del teclado es algo a lo que no nos habíamos referido en ocasiones anteriores y su explicación deja bastante que desear en el manual de Adrian Dickens el formato de llamada se puede observar por ejemplo en la zona etiquetada como cero en las subrutinas de error donde A3 apunta al bloque de definición, he usado dos bloques uno para cada una de las dos filas que se exploran así el primer byte incluye el comando 9 que indica teclado, el segundo byte toma el valor 1 ya que no se le va a mandar nada, la palabra larga siguiente está puesta a 0 ya que no se manda ningún parámetro, el siguiente byte se refiere al número de fila a testear y el último byte indica que se espera una respuesta de 8 bits.

En la redacción del programa se han usado los macros QDOS, FIN, INPUT y ESCRIBIR que son fácilmente usables en otros programas si se pretende hacer ejercicios con los comandos que se tratan en la sección de código máquina ya que para cualquier trabajo que se realice es preciso mantener informado al QDOS.

Se incluye también un cargador de código para aquellos que no deseen teclear todo el listado de ensamblador.

```

100 inicio=RESPI(1856)
110 tempdir=inicio
120 FOR f=1 TO 47
130 intro
140 END FOR f
150 SEEXEC mdv2_multimp_bin,inicio,1856,500
160 STOP
170 DEFine PROCedure intro
180 REPeat aintro
190 check=0
200 cont=1
210 adir=tempdir
220 INPUT ahex$
230 FOR r=1 TO 20
240  bhex$=ahex$(cont TO cont+3)
250  ww=hexdec(bhex$)
260  POKE_W adir,dec
270  check= check+dec
280  cont=cont+4
290  adir=adir+2
300 END FOR r
310 INPUT checksum
320 IF check=checksum THEN PRINT 'correcto':EXIT aintro
330 tempdir=adir
340 END REPeat aintro
350 END DEFine
360 DEFine FuNction hexdec(hex$)
370 dec=0
380 FOR n=1 TO LEN(hex$)
390 LET v=CODE(hex$(n))
400 IF v>64 THEN v=v-55
410 IF v>47 THEN v=v-48
420 dec=dec+2^((LEN(hex$)-n)+4)*v
430 NEXT n
440 RETURN dec
450 END DEFine

```

- 1 '6018000000004AFB000F4D554C54492D494D50524553494F4E0072FF7440700B4E4149FA070672FF', 355779
- 2 '760241FA06CE70014E42B0BC0000000067066100048660BA720076FF43FA0548702E4E43720076FF', 406752
- 3 '43FA0544702E4E4370024E4243FA0518347800C84E92B0BC0000000067066100045260E82948000E', 301607
- 4 '70204E43741576FF43FA05A670074E4343FA04F8347800C64E92B0BC0000000067066100042660B8', 351203
- 5 '288870204E43740976FF43FA063270074E4347FA04FA43EB0002742470024E43927C0001368172FF', 383408
- 6 '760041FA04E270014E42B0BC0000000067082054610003BE60C029480004205476FF70204E437412', 355273
- 7 '76FF43FA05D870074E4347FA04B243EB0002740470024E43927C000136816100041AB07C000066CE', 362335
- 8 'B87CFFFF67C843FA061C3284205476FF70204E43741976FF43FA04D070074E4347FA047443EB0002', 423194
- 9 '740370024E43927C00013681610003BCB07C000066CEB87CFFFF67C8B87C000145FA05BC14842054', 454874
- 10 '205476FF70204E43741676FF43FA054070074E4347FA043043EB0002740170024E43927C00013681', 381866
- 11 '43FA041E45FA05A4121114810C110053670000180C110073670000100C11004E67400C11006E673A', 166064
- 12 '60AE76FF70204E43741D76FF43FA04D270074E4347FA03E043EB0002740370024E43927C00013681', 398159
- 13 '61000348B07C000066D045FA054E1544000172FF760241FA051470014E42B0BC00000000671A2054', 328349
- 14 '76FF70204E43740D76FF43FA03E070074E436100028060D229480008720D76FF70054E43206C0004', 383992
- 15 '76FF70014E43B23C001B6708720070424E436006720270424E4370014E43206C0008B0BCFFFFFFF6', 565021
- 16 '670000EC45FA04E0B23C000A666C062A00010001142A0001B4126702605C157C0000000143FA04BC', 245618
- 17 '0C11006E670C0C11004E67066100027A6040720D76FF70054E43205470204E43741D76FF43FA03D4', 353183
- 18 '70074E4347FA030043EB0002740170024E43927C0001368143FA02EE0C11005367080C1100736702', 292431
- 19 '60CA45FA043E47FA04467A00B21A670A5245BA3C000C66F46040BA7C00046F2ABA7C00056708BA7C', 484310
- 20 '00076F10602B727B70054E43720870054E43601A722C70054E43720870054E43600C722770054E43', 491947
- 21 '720870054E431233500070054E4347FA026E70114E410801000366000126206C00046000FF0643FA', 363551
- 22 '03F6137C0000000143FA03EA045100010C51FFFF660000A6205476FF70204E43741F76FF43FA0266', 286163
- 23 '70074E4347FA023843EB0002740170024E43927C0001368143FA022612110C110053670000180C11', 270187
- 24 '0073670000100C11004E672A0C11006E672460B4205470024E42206C000470024E42206C00087002', 261413

ESCRIBIR	MACRO		SS1	LEA.L	VENT1,A1
	MOVEQ	#1,D2		MOVE.W	UT_SCR,A2
	MOVEQ	#-1,D3		JSR	(A2)
	LEA.L	\2,A1		CMP.L	#0,D0
	QBOS	IO_SSTR6,3		BEQ.S	S2
	ENDM			BSR	ERROR0
INPUT	MACRO		S2	BRA.S	SS1
	LEA.L	BUFFER,A3	SS2	MOVE.L	A0,12(A4)
	LEA.L	2(A3),A1			
	MOVEQ	#1,D2		QBOS	SB_CLEAR,3
	QBOS	IO_FLINE,3		ESCRIBIR	21,MENB
	SUB	D1,D1		LEA.L	VENT2,A1
	MOVE.W	D1,(A3)		MOVE.W	UT_CON,A2
	ENDM			JSR	(A2)
				CMP.L	#0,D0
				BEQ.S	S3
				BSR	ERROR0
{Programa principal				BRA.S	SS2
	BRA.S	RUTINA	S3	MOVE.L	A0,(A4)
	DS.L	1	VUERO		
	DC.W	\$4AFB		QBOS	SB_CLEAR,3
	DC.W	15		ESCRIBIR	9,MEN1
	DC.B	'MULTI-IMPRESION',0		INPUT	36
RUTINA	MOVEQ	#-1,D1		MOVEQ	#-1,D1
	MOVEQ	#64,D2		MOVEQ	#0,D3
	QBOS	MT_PRIOR,1		LEA.L	BUFFER,A0
	LEA.L	ID,A4		QBOS	IO_OPEN,2
	MOVEQ	#-1,D1		CMP.L	#0,D0
	MOVEQ	#2,D3		BEQ.S	S4
	LEA.L	VENTO,A0		MOVE.L	(A4),A0
	QBOS	IO_OPEN,2		BSR	ERROR
	CMP.L	#0,D0	S4	BRA.S	VUERO
	BEQ.S	S1		MOVE.L	A0,4(A4)
	BSR	ERROR0	VUER1		
	BRA.S	RUTINA		MOVE.L	(A4),A0
S1	MOVEQ	#0,D1		MOVEQ	#-1,D3
	MOVEQ	#-1,D3		QBOS	SB_CLEAR,3
	LEA.L	BLOCK1,A1		ESCRIBIR	18,MEN2
	QBOS	SB_FILL,3		INPUT	4
	MOVEQ	#0,D1		BSR	RUTCON
	MOVEQ	#-1,D3		CMP	#0,D0
	LEA.L	BLOCK2,A1		BNE.S	VUER1
	QBOS	SB_FILL,3		CMP	#-1,D4
	QBOS	IO_CLOSE,2		BEQ.S	VUER1
				LEA.L	NUMREP,A1
				MOVE	D4,(A1)

```

ESPIMP      MOVE.L      (A4),A0
            MOVEQ      #1,D3
            QDOS      SD_CLEAR,3
            ESCRIBIR  25,MEN10
            INPUT     3
            BSR       RUTCON
            CMP       #0,D0
            BNE.S     ESPIMP
            CMP       #1,D4
            BEQ.S     ESPIMP
            ADD       #1,D4
            LEA.L     CLIN,A2
            MOVE.B    D4,(A2)
            MOVE.L    (A4),A0

CIIMP       MOVE.L      (A4),A0
            MOVEQ      #1,D3
            QDOS      SD_CLEAR,3
            ESCRIBIR  22,MEN3
            INPUT     1
            LEA.L     BUFFER+2,A1
            LEA.L     LINEAS,A2
            MOVE.B    (A1),D1
            MOVE.B    D1,(A2)
            CNPI.B   #83,(A1)
            BEQ      NIMP
            CNPI.B   #115,(A1)
            BEQ      NIMP
            CNPI.B   #78,(A1)
            BEQ.S    S12
            CNPI.B   #110,(A1)
            BEQ.S    S12
            BRA.S    REPETIR

NIMP        MOVEQ      #1,D3
            QDOS      SD_CLEAR,3
            ESCRIBIR  29,MEN4
            INPUT     3
            BSR       RUTCON
            CMP       #0,D0
            BNE.S    NIMP
            LEA.L     LINEAS,A2
            MOVE.B    D4,1(A2)

SIMP        MOVEQ      #1,D1
            MOVEQ      #2,D3
            LEA.L     IMPRESORA,A0
            QDOS      IO_DFEX,2
            CMP.L     #0,D0
            BEQ.S    S10
            MOVE.L    (A4),A0
            MOVEQ      #1,D3
            QDOS      SD_CLEAR,3
            ESCRIBIR  13,MEN11
            BSR       ERROR
            BRA.S    SIMP
            MOVE.L    A0,8(A4)
            MOVEQ      #90,D1
            MOVEQ      #1,D3
            QDOS      IO_SBYTE,3
            MOVE.L    4(A4),A0
            MOVEQ      #1,D3
            QDOS      IO_FBYTE,3
            CMP.B     #0,D1
            BEQ.S    S12
            MOVEQ      #0,D1
            QDOS      FS_POSAB,3
            BRA.S    REPETIR
            MOVEQ      #2,D1
            QDOS      FS_POSAB,3
            QDOS      IO_FBYTE,3
            MOVE.L    8(A4),A0
            CMP.L     #10,D0
            BEQ      ACABAR
            LEA.L     CLIN,A2
            CMP.B     #90A,D1
            BNE.S    BIFURCAR
            ADDI.B    #1,1(A2)
            MOVE.B    1(A2),D2
            CMP.B     (A2),D2
            BEQ.S    IPAG
            BRA.S    BIFURCAR
S10
CICLO
S12
REPETIR
ACABAR
IPAG
BIFURCAR

```


			TESTED	
O TRAV	MOVE.L	(A4),A0		QDOS
	MOVEQ	#-1,D3		ESCRIBIR
	QDOS	SD_CLEAR,3		INPUT
	ESCRIBIR	31,MEN12		LEA.L
	INPUT	1		CMPI.B
	LEA.L	BUFFER+2,A1		BEQ
	MOVE.B	(A1),D1		CMPI.B
	CMPI.B	#83,(A1)		BEQ
	BEQ	OTRA		BRA.S
	CMPI.B	#115,(A1)		
	BEQ	OTRA	CONTINUO	
	CMPI.B	#78,(A1)		BSR
	BEQ.S	NOOTRA		BRA
	CMPI.B	#110,(A1)		
	BEQ.S	NOOTRA	ABORTAR	
	BRA.S	O TRAV		MOVEQ
				MOVE.L
				QDOS
				ESCRIBIR
				FIN
O TRA	FIN			MOVEQ
	BRA	RUTINA		MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR
				FIN
				MOVEQ
				MOVEQ
				QDOS
				ESCRIBIR

Espacio de datos

VENT1	DC.B	248,1,2,0
	DC.W	249,20,256,0
VENT2	DC.B	248,1,2,0
	DC.W	249,40,256,20
BLOCK1	DC.W	244,7,10,60
BLOCK2	DC.W	5,50,249,10
ENTER	DC.B	9,1
	DC.L	0
	DC.B	1,2
FUNC	DC.B	9,1
	DC.L	0
	DC.B	0,2
BUFFER	DS.W	1
	DS.B	36
MEN12	DC.B	'¿Desea imprimir otro fichero?: ',0
MEN11	DC.B	'Salida serie ',0
MEN10	DC.B	'num. de líneas por pag.: ',0
MEN9	DC.B	'F1 continuar'
MEN8	DC.B	'MULTI-IMPRESION 1.0',0
MEN7	DC.B	'E/S completada. Fin de trabajo'
MEN6	DC.B	'E/S incompleta, prog. abortado'
MEN5	DC.B	'Introduzca hoja y pulse s: ',0
MEN4	DC.B	'Número de líneas entre pag.: ',0
MEN3	DC.B	'Papel continuo (s/n): '
MEN2	DC.B	'Número de copias: '
MEN1	DC.B	'Fichero: ',0
MENR	DC.B	'Número incorrecto',0
VENTO	DC.W	16
	DC.B	'SCR_256X70A256X0'
IMPRESORA	DC.W	4
	DC.B	'SER1'
ACENT	DC.B	\$8C,\$83,\$93,\$96,\$99,\$87,\$A8,\$88,\$B4,\$B3,\$A9,\$89
NACENT	DC.B	\$61,\$65,\$69,\$6f,\$75,\$75,67,99,93,91,92,124
ID	DS.B	16
LINEAS	DS.B	2
NUMREP	DS.B	2
CLIN	DC.B	2

END

NOTIFICACIONES



RENOVACION DE LA CONDICION DE SOCIO

El próximo mes de Junio finaliza el plazo de abono de aquellos socios que se suscribieron al Club para el primer semestre del año. Dichos socios, si desean seguir perteneciendo @Lave, deberán renovar el abono de la cuota, para el segundo semestre del año.

La renovación puede realizarse enviando un giro postal al tesorero del Club (indicando en la parte del mismo destinada a texto, el número de socio, nombre y apellidos, así como la palabra 'RENOVACION'), o enviando un cheque junto con el número de socio, nombre y apellidos.

El importe de la cuota para los meses de Agosto a Diciembre es de 1.300 pts.

(Os recordamos que al no tener aún un número de sociedad en el registro de asociaciones, no podemos abrir una cuenta bancaria a nombre de @Lave, por lo que tampoco podemos cobrar talones o giros a dicho nombre; por ello deben ir a nombre del tesorero)

Las renovaciones de cuotas deben enviarse antes del 31 de Junio. Se entenderá que aquellos socios, abonados para el primer semestre del año, que no renueven la cuota, manifiestan su voluntad de causar baja en el Club.

Este mes han entrado a formar parte de la librería los siguientes programas:

- 1.- Recibos de navidad ; longitud: 5282 bytes.
- 2.- Regresiones ; longitud: 6250 bytes.

También son accesibles por librería todas las rutinas que se publiquen en el boletín en la sección del mismo nombre.

El programa Recibos de navidad se encarga de la confección de este tipo de recibos y la obtención de una copia impresa de los mismos. En esta misma sección se puede ver una muestra de sus resultados.

El programa Regresiones se encarga de realizar el ajuste de una nube de puntos a una función lineal, exponencial o logarítmica por los métodos tradicionales en estos casos.

SUMARIO

- 1 . - PORTADA
- 2 . - INFORMACION SOBRE EL CLUB
- 3 . - EDITORIAL
- 4 . - COMENTARIO DE PROGRAMAS
 - QMON
 - COMWARE TOOLKIT
- 7 . - HARDWARE
- 11 . - INTRODUCCION AL C. MAQUINA
- 18 . - CORREO DE LOS SOCIOS
- 19 . - NOVEDADES
- 22 . - PREGUNTAS
- 29 . - AMSTRAD VS SINCLAIR
- 33 . - TRUCOS Y RUTINAS
 - MULTI-IMPRESION
- 43 . - NOTIFICACIONES
- 44 . - SUMARIO