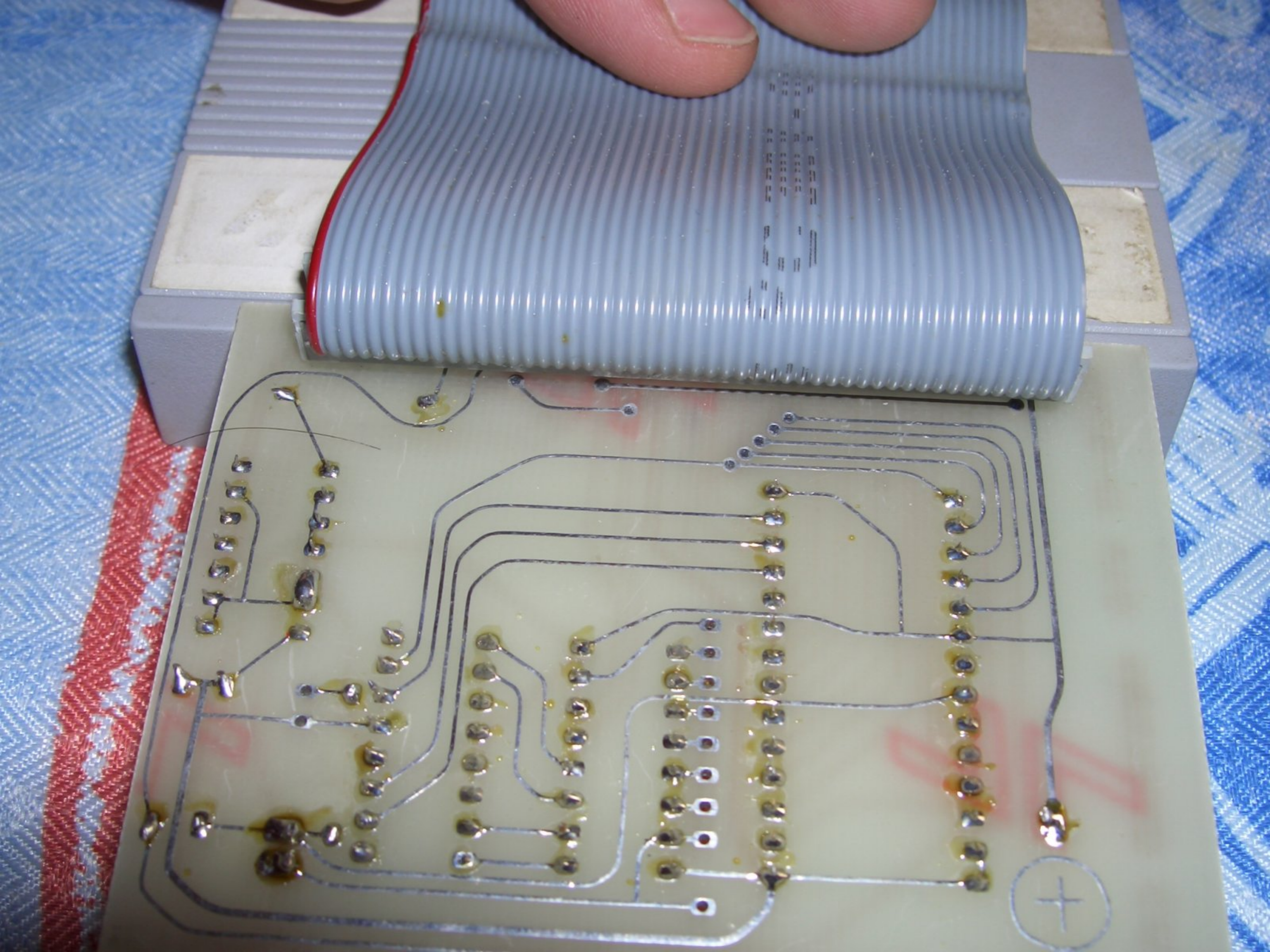
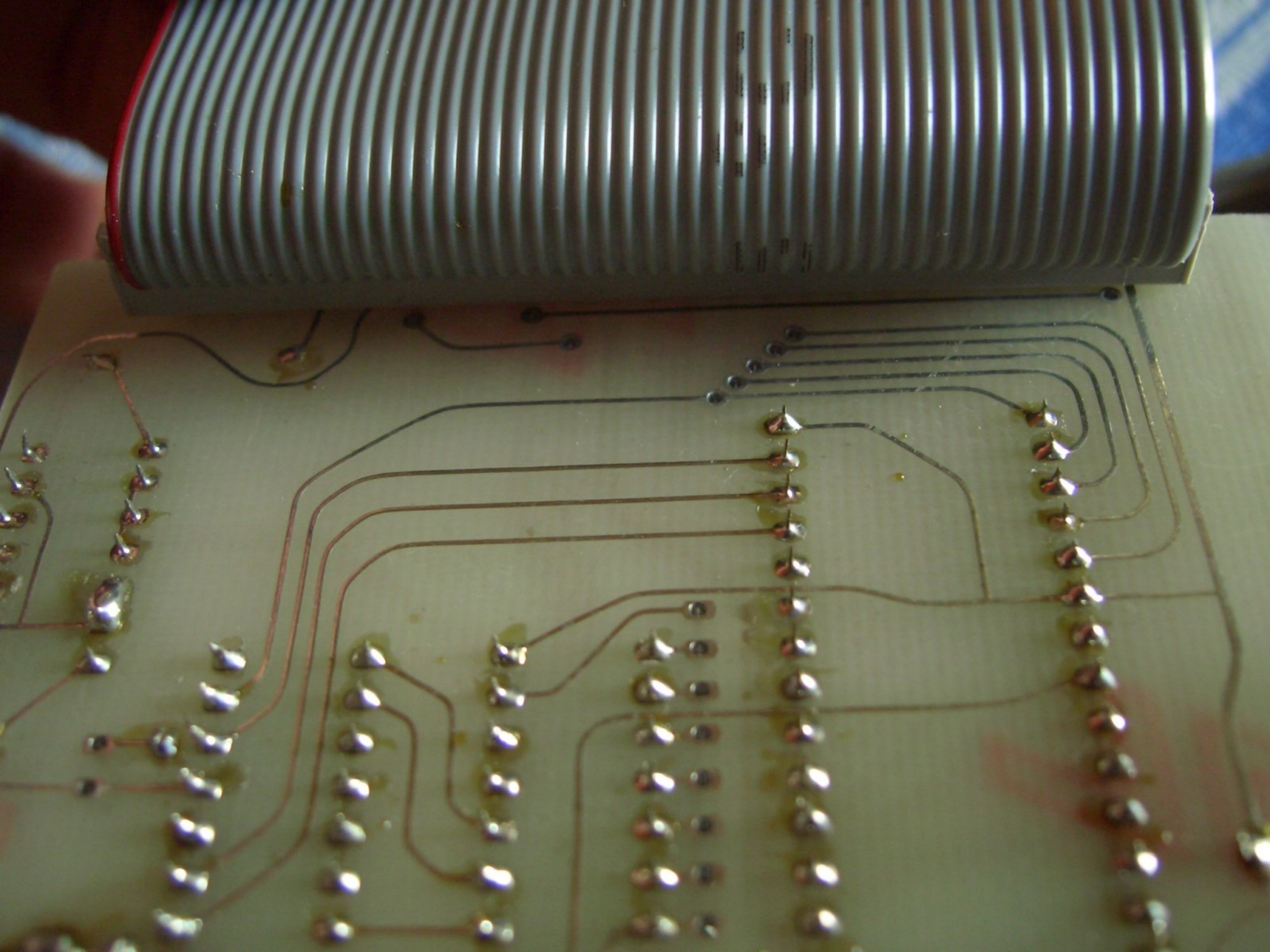
A vintage computer floppy disk is shown vertically. The disk is silver with a black top edge. A custom label is affixed to the front. The label has two sections, both containing the text "LE HACKER" and "SIREN SOFTWARE". The top section of the label is white with black text, and the bottom section is a light-colored strip with black text. The disk is set against a background of colorful, patterned fabric.

LE HACKER  
SIREN SOFTWARE

LE HACKER  
SIREN SOFTWARE





(C) S... SOFTWARE 1978.



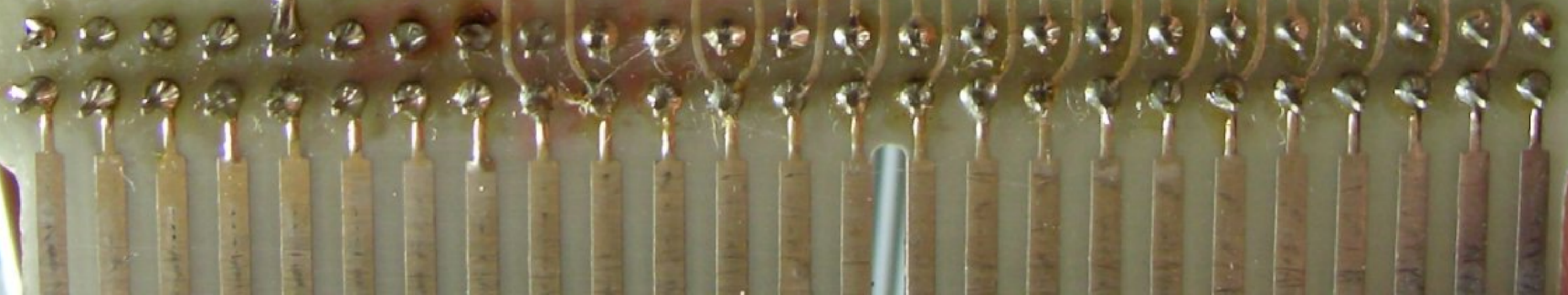
F 8948 CR  
SN74LS27N

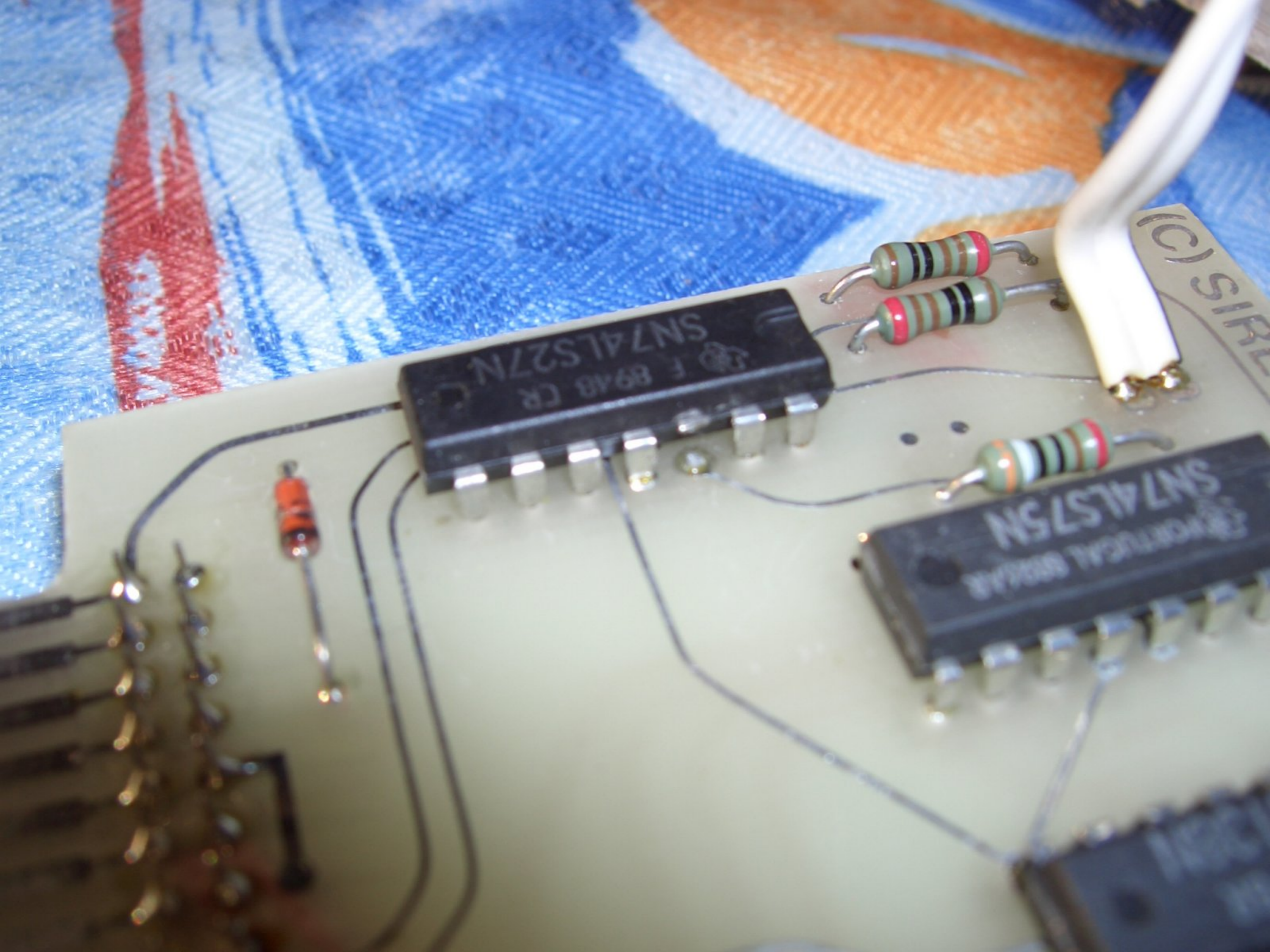
PORTUGAL 8826AR  
SN74LS75N

F 9001 BR  
SN74LS138N

D27128A-20  
K8522203  
© i '83

HACK-IT V1.0 AMSTRAD

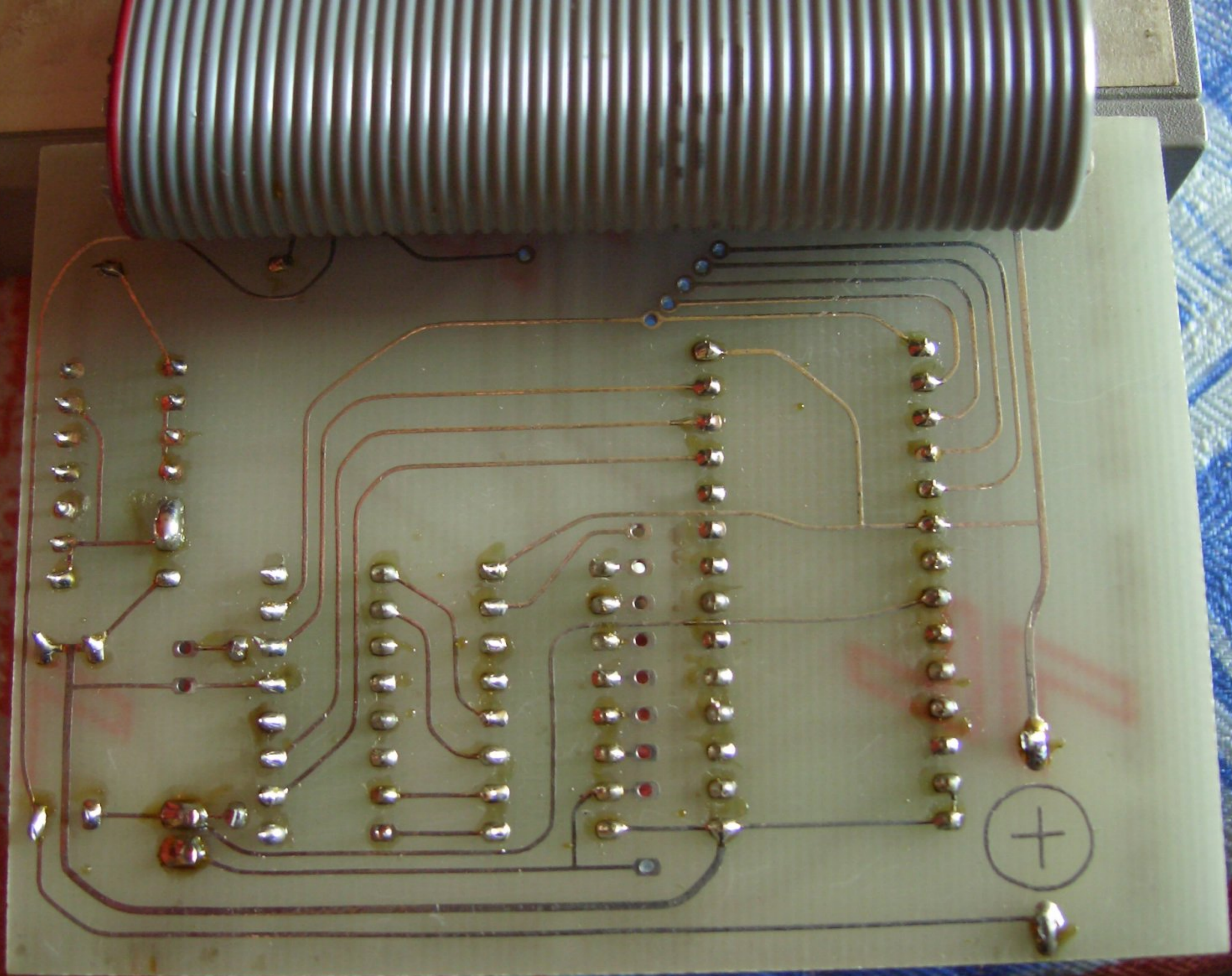




SN74ALS27N  
E 8948 CR

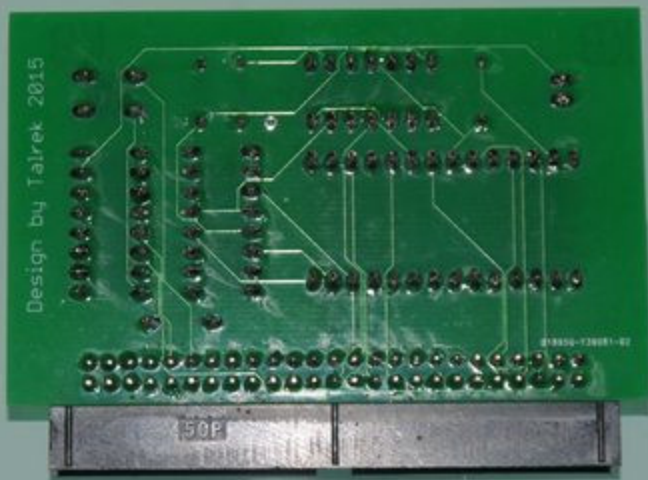
SN74ALS75N  
E 8948 CR

(C) SIRI



(C) SIREN SOFTWARE 1978.





Design by Talrek 2015

01850-12001-01

50P





## Le HACKER Pro

Copyright (1989) SIREN Software & DUCHET Computers  
Version Française Copyright (1989) DUCHET Computers

Nous vous remercions d'avoir fait l'acquisition de l'interface Le HACKER. Vous avez bien eu raison et vous ne le regretterez pas.

Nous rappelons que Le HACKER a été conçu pour les programmeurs avertis sachant parler couramment en langage machine, en hexadécimal et ayant développé une divine passion pour jongler avec les assembleurs. Nos circulaires et publicités préviennent sans ambiguïté de ce détail!

### ATTENTION !

Avant de connecter Le HACKER, assurez-vous bien que votre CPC ne soit pas sous tension.  
Sinon, Le HACKER risque d'être tragiquement endommagé et sa garantie de 3 mois est automatiquement invalidée.

### DESCRIPTION

Le HACKER se présente sous la forme d'un boîtier interface muni de 2 interrupteurs et de 2 connecteurs.  
Le HACKER se connecte en reliant la prise terminant son câble souple au port d'extension de votre CPC.  
Le connecteur arrière permet de relier d'autres périphériques de votre choix.  
Le HACKER se branche donc à plat, le bouton rouge étant du côté de la touche "ESC" et l'interrupteur métallique étant vers la touche "9".

L'interrupteur métallique sert à valider/invalider Le HACKER. Le bouton rouge est le bouton magique qui permet de stopper tout programme à tout moment et de l'explorer, modifier, etc...

Selon votre choix, votre tâche présente et le programme "victime", vous validerez ou invaliderez d'avance l'interrupteur métallique.  
S'il est validé d'avance, Le HACKER prend directement le contrôle lorsque vous mettez le CPC sous tension ou lorsque vous appuyez simultanément sur les touches CONTROL/SHIFT/ESC, ou enfin lorsque le Z80 exécute un CALL 0 ou RST 0.  
S'il est validé après ou durant chargement d'un programme, Le HACKER prendra le contrôle lorsque vous presserez le bouton rouge. A ce moment, le Z80 fait un reset, le firmware AMSTRAD (jumpblocks, etc..) s'initialise. L'écran se vide, le message habituel AMSTRAD se montre pendant un court instant (c'est normal), puis enfin le message du HACKER apparaît.

Toute la mémoire de &A40 à &A67F est intacte et à votre entière disposition.

Le HACKER parle uniquement en hexadécimal. Pour vous faciliter la tâche, deux commandes permettent de traduire Hexadécimal/décimal et vice-versa.

Toutes les commandes doivent être tapées après le ">". D'autres ">" permettent ensuite de taper les paramètres. Rappelons: Le HACKER travaille en hexadécimal.

Avec l'habitude et en appuyant sur le bouton rouge durant différentes phases du chargement d'un logiciel, vous découvrirez les messages cachés et les secrets jalousement gardés de programmes inaccessibles (ou pire!)

Par exemple (et ceci n'est qu'un exemple, ne nous téléphonez pas en hurlant "Moi je ne trouve pas la même chose!"), en appuyant sur le bouton rouge pendant le chargement de notre version de DISCOLOGY, nous trouvons à l'adresse &3FAC que les auteurs de ce progiciel ont utilisé le "HISOFT GENA3.1 ASSEMBLER AMSTRAD CPC 464" pour assembler DISCOLOGY.

Maintenant chargeons l'écran de présentation de DISCOLOGY, appuyons sur le bouton et éditons l'adresse &2105 où figure un message se rapportant à nos malheureux concurrents de chez Romantic Robot!

Tapons "M" pour voir la carte de la mémoire... Nous remarquons quelque chose à &8000.

Chargeons le COPIEUR de DISCOLOGY, allons à l'adresse &3952 après avoir appuyé sur le bouton et nous trouvons le message "Autre écriture du buffer (O)ui ou (N)on ?"

Si vous chargez le programme "DRUID" (version disquette), vous découvrirez à &1DA9 un message laissé par l'auteur de XAVIOR, DRUID et COVENANT dans lequel il accuse la compagnie P.S.S. de l'avoir escroqué!

Vous devinez donc quelles possibilités s'offrent à vous en appuyant sur le bouton rouge au "bon" moment!

#### TRAVAILLONS SERIEUSEMENT

Lorsque vous pressez le bouton rouge, Le HACKER vous apporte 34 commandes !

Passons-les en revue par ordre alphabétique:

L'abréviation minimale de chaque commande est indiquée entre ( )

##### ASSEMBLE (AS)

L'assembleur.

Toutes les instructions doivent être tapées en MAJUSCULE.

Tous les nombres seront en HEXADÉCIMAL et précédés du signe &

Tous les "opcodes" standards et publiés du ZILOG 80 sont supportés.

Les instructions RST prennent les nombres RST 0,8,10,18,20,28,30 et 38 (méthode normale) plutôt que la méthode moins orthodoxe 0,1,2,3,4,5,6 et 7.

Prenons un exemple:

Taper AS <ENTER> pour valider l'assembleur.

On vous demande une adresse de départ, tapez (par exemple) 8000  
Le > apparait.

Tapez (par exemple) les commandes suivantes, suivies de <ENTER> à la fin de chaque ligne. La ligne suivante donnera l'adresse de la commande suivante.

A la fin, tapez Q pour quitter.

```
LD HL,&4000
LD A,(HL)
OR A
RET Z
CALL &BB5A
INC HL
JR &F7
Q
```

Cette routine affichera sur l'écran tout texte se trouvant à la location &4000. Pour mettre du texte à &4000, passer en mode EDIT (voir commande EDIT plus bas). Entrez le texte en Hexadécimal (ou pressez TAB et entrez-le en ASCII - ça ira plus vite!)

Tapez (par exemple) "Ca marche !"

Passez par la commande CALL (voir plus bas) pour appeler &8000.  
Le texte situé à &4000 s'affichera.

Si vous tapez une bêtise, le message "INVALID COMMAND" (qui veut dire "commande non-valide"!) s'affichera lorsque vous pressez <ENTER> en fin de ligne.

#### BANK (BAN)

Donne accès à la deuxième BANK de 64K d'un CPC 6128 ou d'un 464/664 étendu avec une extension 64K DK TRONICS.

Les blocs de 16K sont numérotés de 1 à 4 dans la portion de mémoire &4000/&8000.

Lorsqu'une extension mémoire DK TRONICS de 256K est branchée, il est donc possible d'accéder à 16 blocs de 16K.

#### BASIC (B)

Cette commande retourne au BASIC, à l'AMSDOS, et vide la mémoire (attention au vidage de mémoire par inadvertance!)

#### BORDER (BO)

Pour changer la couleur de la bordure

#### CALL (CAL)

Pour appeler une routine.

Cette commande permet à tous les registres du Z80 d'être chargés et affichés au retour de la routine.

Notez bien que, tous les jumpblocks du firmware étant initialisés, il est possible d'utiliser cette commande pour accéder à toutes les routines du firmware.

#### CAT (CA)

Catalogue la cassette ou la disquette suivant le système initialisé.

#### CLEAR (CLE)

Remplit de zéros toute la mémoire de &40 à &AB80 ainsi que celle des extensions mémoire.

#### CLS (C)

Vide l'écran et place le curseur à son sommet gauche.

#### COLOURS (CO)

Affiche les 16 INKS et comment elles ont été réglées. Utile si les INKS doivent être examinées après qu'une routine ait été appelée.

#### COPY (COP)

Pour déplacer une portion de mémoire d'une adresse à une autre. La portion originale est laissée intacte.

#### DECTOHEX (DE)

Pour convertir les nombres décimaux (0-65535) en hex.

#### DISASSEMBLE (DISA)

Désassemble une portion spécifiée de mémoire en mnémoniques Z80. Si vous tapez un titre de fichier en plus des adresses de départ et de fin, un FICHER ASCII contenant le désassemblage sera créé sur cassette ou disquette. Par la suite, vous pourrez le charger dans votre assembleur favori pour ré-assemblage. Sinon, tapez sur RETURN et aucun fichier ne sera créé.

#### DISC (D)

Lorsque Le HACKER entre en action, le système disquette n'est pas initialisé. La portion de mémoire (&A67B à &AB7F) normalement utilisée par la ROM du disque (n° 7) peut donc être examinée, déplacée, etc...

Cette commande permet d'initialiser le disque à l'adresse de défaut en appuyant sur RETURN, ou à une adresse de votre choix que vous spécifierez.

#### EDIT (E)

Pour examiner et éditer la mémoire. Après que vous aurez tapé une adresse, &188 octets apparaîtront sur l'écran à partir de l'adresse choisie. Déplacez-vous avec le curseur sur l'écran et faites le dérouler pour monter ou descendre en mémoire. TAB permet de passer de HEX à ASCII pour éditer. Pour sortir de l'éditeur de mémoire: ESC <RETURN>

#### FILL (F)

Pour remplir un bloc de mémoire avec une valeur donnée. Utile pour nettoyer des portions de mémoire.

#### HEADER (HEA)

Affiche l'information contenue dans l'en-tête d'un programme. L'adresse, la longueur, l'entrée et le type de programme (BASIC Protégé ou non, binaire, ASCII)

#### HELP (H)

Vous donne la liste des commandes disponibles avec Le HACKER

#### HEXTODEC (HEX)

Pour convertir les nombres hex 16-bit (0-FFFF) en nombres décimaux

#### IN (I)

Lit un octet à partir d'un port

#### LOAD (L)

Charge un fichier à partir d'une cassette/disquette après que vous en aurez tapé son titre.

Une adresse de chargement peut être spécifiée.

Si vous appuyez sur RETURN au lieu de spécifier une adresse, le fichier se chargera à l'adresse affichée dans son en-tête.

#### MEMORY (M)

Cette ingénieuse commande permet d'afficher la carte complète de la mémoire!

Les blocs de mémoire inoccupés sont en bleu.

Les blocs occupés sont en jaune.

En général, un bloc jaune "solide" représente des portions de programme ou de texte, alors qu'un bloc jaune et bleu représente les portions de graphisme dans les jeux.

Il est évidemment conseillé de taper cette commande pour commencer afin d'avoir une idée générale de la mémoire.

#### MOVE (MO)

Pour déplacer une portion de mémoire d'une adresse à une autre. La portion originale laissée vacante est aussitôt remplie de zéros.

#### OUT (O)

Envoie un octet à un port

#### PAPER (PA)

Pour changer la couleur de l'arrière-plan

#### PEEK (PEE)

Le PEEK des anciens

#### PEN (P)

Pour changer la couleur de l'avant-plan

#### POKE (PO)

Le POKE des anciens

#### PROFF

Invalide la commande "PRON".

#### PRON

Pour envoyer sur imprimante le data affiché sur l'écran.

Une exception est avec la commande EDIT. Il est cependant possible de faire une copie de l'écran à tout moment en appuyant simultanément sur les touches CONTROL et COPY.

Notez toutefois que la carte de la mémoire (commande "M") ne s'imprime pas.

#### RESTORE (RE)

Ramène les couleurs au noir et blanc. C'est plus lisible d'une part, et d'autre part ça rend bien service lorsque vous avez appelé une routine qui vous donne un fouillis monstre de couleurs!

#### RSX (R)

Pour accéder aux ROMS qui pourraient être reliées à l'ordinateur (MAXAM, UTOPIA, etc...). RSX vous demande le titre RSX et essaiera d'exécuter la commande.

Notez que la ROM en question doit tout d'abord avoir été initialisée. Pour détails, consultez (par exemple) le COMPLETE FIRMWARE AMSTRAD (Soft 968) à la référence du Jumpblock 154: KL INIT BACK à l'adresse &BCCE où l'initialisation des ROMS est expliquée.

La commande "RSX" permet aussi de passer de cassette à disque (!TAPE / !DISC)

#### SAVE (S)

Pour sauvegarder une portion spécifique de mémoire sur cassette ou disque.

Tapez le titre, adresse de départ, longueur, adresse d'entrée et une adresse de chargement, puis spécifiez la nature du fichier (0 ou 1).

Il est donc possible (par exemple) de prendre une portion de mémoire à &8000 et la faire se charger à &4000 si vous en voyez l'utilité.

Les options d'adresse, etc.... s'appliquent aux fichiers à sauvegarder sur disquette et non à ceux sauvegarder sur cassette.

#### SEARCH (SE)

Pour faire des recherches en mémoire.

Taper "H" pour recherche hexadécimale

Taper "A" pour recherche de donnée ASCII

Les adresses contenant la donnée recherchée seront affichées (appuyez sur ESC pour abandon). Notez que même si la donnée n'existe pas, une adresse sera indiquée; vous avez en effet tapé la donnée, ce que vous avez tapé au clavier se trouvera donc forcément en mémoire!

Une commande toute particulière:

#### ALTERNATE (A)

Cette commande fonctionne uniquement avec un 464/664 sur laquelle une extension mémoire 64K DK TRONICS est branchée. Elle ne fonctionnera pas de façon satisfaisante avec un 6128 ou un 464/664 de base.

Sur un 464 étendu, il est possible de faire tourner un programme dans la deuxième portion de mémoire 64K et de faire croire à l'ordinateur que le programme est en fait dans la première portion de 64K!

Marche à suivre :

1. Taper "ALTERNATE" (A)
2. Le curseur disparaît. Ne pleurez pas.

3. Invalidez Le HACKER (interrupteur métallique)
4. Appuyez simultanément sur CONTROL, SHIFT et ESC
5. Un bazar incroyable apparaît sur l'écran
- 5a. Si l'ordinateur fait un reset complet, c'est parce que vous avez un 6128 ou un 464/664 non muni de 64K DK TRONICS
6. Tapez les instructions pour charger votre jeu.
7. Vous ne verrez pas ce que vous tapez, mais votre ordinateur comprendra fidèlement. Toutefois, si vous faites des fautes de frappe il ne vous obéira plus!
8. Une fois le jeu chargé, validez Le HACKER (interrupteur métallique)
9. Appuyez sur le bouton rouge
10. Commande BANK
11. Choisir BANK 4
12. Commande COPY
13. Copier 4000 octets de l'adresse 4000 à l'adresse C000
14. L'écran du jeu sera copié sur l'écran présent
15. Le code du jeu peut alors être édité en examinant les banks 1, 2 et 3 de la deuxième portion de 64K.

\*\*\*\*\*

Si le chargement d'un programme présente un problème alors que Le HACKER est validé, il suffit de l'invalider, de charger le programme puis de valider Le HACKER avant d'ensuite presser le bouton rouge.

Le HACKER n'a pas été conçu dans le but de pirater les logiciels ni de transférer des programmes de K7 à disquette.

Le HACKER a été développé pour permettre aux programmeurs enthousiastes et sérieux d'examiner, modifier et donner une touche personnelle à leurs programmes, ainsi que de travailler directement en assembleur.

Vu le coût très modique du HACKER et la complexité des opérations informatiques permises par cette interface, nous regrettons de ne pouvoir assurer un support technique par téléphone ni par courrier.

Pour utiliser Le HACKER vous devez être un familier de l'assembleur et du langage machine. Nos circulaires et publicités insistent sur ce point. Nous n'aurons donc aucune pitié pour les pleurnicheurs habituels qui se précipitent sur le téléphone et se lamentent "Je n'y arrive pas et pourtant je m'y connais car mon beau-frère a un ami qui est presque allé au SICOB en 1983". A ceux-la nous leur conseillerons de mettre leur ordinateur dans un bain de naphthaline et (en hex):  
 44,27,41,4C,4C,45,52,20,43,55,4C,54,49,56,45,52,20,44,45,53,20,53,41,  
 4C,41,44,45,53,21,21

Chaque HACKER est individuellement testé avant son expédition.

Nous remplaçons bien sûr sous garantie un HACKER défectueux,

**MAIS MAIS MAIS :**

Au cas extrêmement improbable où votre HACKER semblerait être



défectueux, ne commencez pas par nous le retourner avec une note "Ca marche pas, j'en veux un qui marche".

Relisez d'abord les instructions.

Si votre HACKER ne fonctionne toujours pas, contactez-nous par écrit ou par téléphone en décrivant très exactement les symptômes S.V.P.

Nous ignorerons des réclamations telles que:

"J'obtiens un RESET avec la 74ème partie du programme xyz.bin"

"Il fonctionne sur le CPC de mon copain mais pas sur le mien"

"Mon ordinateur est un SPECTRUM PLUS 2, je n'arrive pas à brancher Le HACKER. Remboursez-moi" (Riez, mais 3 possesseurs de Spectrum +2 mal informés nous ont acheté des MIRAGE IMAGERS en espérant un miracle)

Ne retournez Le HACKER que si nous avons au préalable donné notre accord avec un numéro de retour par écrit ou par téléphone. Tout HACKER retourné sans numéro de retour sera automatiquement refusé.

Nous regrettons d'avoir à prendre cette attitude, mais l'expérience nous a prouvé que sur 10 articles retournés, 9 étaient en parfait état de fonctionnement!!! D'autre part, plus de la moitié de ceux s'avérant défectueux avaient subi des abus.

Attention! Le HACKER est un appareil long à tester. Nous ne renvoyons pas à nos frais un appareil retourné comme défectueux alors qu'il ne l'était pas. Nous facturons un minimum de 100 FF pour dédommager du temps perdu en tests inutiles et des frais de retour (en recommandé avec accusé de réception).

De même nous facturons les réparations de tout appareil abusé, vandalisé, grillé "par mégarde", rongé par le chien (un IMAGER retourné portait des marques de crocs!), etc....

\*\*\*\*\*

Nous vous remercions de votre clientèle fidèle et vous souhaitons de nombreuses années de manipulations informatiques avec Le HACKER Pro!

**42, 4F, 4E, 20, 43, 4F, 55, 52, 41, 47, 45, 21**

ADDITIF A LA NOTICE  
DU HACKER VERSION 4.00  
POUR LA VERSION 4.80

-----

Certaines options "inutiles" ont été supprimées sur cette version (pour gagner un peu de place sur l'EPROM). Les options BORDER, PAPER, PEN, COLOURS et RESTORE n'existent plus.

L'affichage du texte a été accéléré. Dans les options SAVE et LOAD, les noms de fichiers peuvent comporter des caractères autres que lettre ou chiffre. L'option SEARCH marche dans tous les cas (il y avait un bug dans la recherche de chaîne HEX se terminant par &00). L'option DISASSEMBLE a été débuggée (certaines instructions n'étaient pas décodées, ou mal décodées).

Voici la liste des nouvelles options avec leur commande abrégée entre parenthèse:

- ERASE (ER): permet d'effacer un fichier sur disc
- RENAME (RE): permet de renommer un fichier sur disc.
- LDASC (LD): permet de charger un fichier ASCII.
- PROGIMP (PR): permet le paramétrage de l'imprimante (courrier, condensé, etc).
- DFORMAT (DF): formatage d'un disc au format DATA sur 42 pistes.
- SFORMAT (SF): formatage d'un disc au format SYSTEME sur 42 pistes.
- RAM (RA): permet un retour au basic sans effacer le contenu de la RAM.
- INRAM (IM): hardcopy de la carte mémoire.
- EDIMP (EDIM): permet l'édition de la mémoire par page (ESC pour sortir).
- COMPARE (CO): permet de faire une comparaison entre 2 blocs de la mémoire. Il ne restera dans le 1er bloc que les octets différents du 2nd bloc.
- CLOG (CLO): permet de charger le catalogue à une adresse déterminée. L'option CAT charge le catalogue en &03E8, ce qui peut effacer certains programmes.
- SECT (SEC): permet de charger en mémoire un ou plusieurs secteurs consécutifs en mémoire. Il charge les secteurs de taille 1 à 6, avec données normales ou effacées.

L'option SECT nécessite de savoir le numéro du 1er secteur à lire, sa taille, le numéro du 1er secteur de la piste, le nombre de secteurs par piste et si le secteur est en données normales ou effacées (mapping avec DISCOLOGY). Les secteurs taille 6 sont particuliers. Ils font en théorie &2000 octets. Mais en réalité, on ne peut lire que &1800 octets. Si vous chargez plusieurs secteurs taille 6 consécutifs, ils seront chargés tous les &1800 octets. Par contre, le dernier secteur chargé fera lui &2000 octets.

INSTRUCTIONS

The Hackit interface allows any program to be stopped and hacked. The interface connects to the expansion port on the back of the computer and other peripherals can be attached to the Hackits own expansion port.

Before connecting Hackit, please ensure that the computer is switched off. The Hackit must be connected with the ribbon cable coming down from underneath the edge connector that fits the expansion bus on the computer.

The only two distinguishing features of the box itself are the Red reset button to the left of the box and the on/off switch at the right of the box.

The on/off switch is self explanatory, when switched on, every time the computer is reset by pressing the Hackits reset button, control will pass to Hackit. (Control will also pass to Hackit when the computer is reset by pressing the CONTROL-SHIFT & ESC keys together, or when the computer is turned on or finally when a CALL 0 or RST 0 instructions are executed by the Z80 Processor.)

The reset button will either completely reset the computer when the Hackit is switched off, or will stop the currently running program and pass control to Hackit if it is switched on.

When Hackit is switched on and the reset button is pressed, the following will happen:- The Z80 will reset, the Amstrad firmware (jumpblocks etc) will initialise. The screen will clear and the Amstrad power up message will show for a split second before the Hackit comes into action.

All memory from &40 to &A67F will remain intact for examination, alteration etc. The Hackit works solely in Hexadecimal, however it does have built in commands for hex to decimal conversion and vice-versa.

All commands must be entered at the ">" prompt and further prompts will allow parameters such as address, filenames etc to be entered. (in hex only).

Below is a list of commands and what they do.

**DISC** Initialises the disc rom (rom 7) allowing access to the disc system as opposed to the cassette system. When Hackit comes on, the disc system is not initialised in order for memory that the disc rom uses, can be examined, moved etc. The disc rom normally occupies memory from A67B to AB7F. The disc command allows you to specify at what address the disc rom will use.

**EDIT** This allows memory to be examined and altered. After specifying an address, the screen will fill with 188 bytes starting from the specified address. The cursor can be moved around the screen which will scroll up and down when the cursor reaches the top or bottom of the screen. The TAB key is used to toggle between hex & ascii editing whilst the ESC key will exit from the editor.

**BASIC** This command simply exits Hackit and reverts back to Basic and Amsdos. All memory will be cleared by this command.

**CLS** Clears the screen and moves the cursor to the first line on the screen.

**CAT** Catalogue a tape or disc depending on whether the tape or disc system is currently in operation.

**RSX** This command can be used to access any Roms that may be attached to the computer (i.e. Maxam, Utopia etc). This command will prompt you for the RSX name and will try and find and execute the relevant command. Please note that the relevant ROM must first be initialised.

Please see the Amstrad Firmware guide with reference to jumplock 154: KL INIT BACK at address BCCE in order to initialise a rom. This command can also be used for switching between tape and disc. i.e. !TAPE / !DISC

- PEN** Allows the foreground colour to be altered.
- PAPER** Allows the background colour to be altered.
- BORDER** Allows the border coloured to be changed.
- CALL** Allows a routine to be called. This command allows the individual Z80 registers to be loaded and displayed on return from the routine. Please note that because all the firmware jumpblocks are initialised, it is possible to use this command to access all the firmware routines.
- MEMORY** This useful command allows the whole user memory (0-C000) to be viewed on one screen. The display will show which blocks of memory are not occupied (blue areas) and which parts are occupied (yellow areas). Can be usefull for determining which areas of memory are actually used by the program and which are not. Patchy yellow/blue areas are often used for graphics data for games, whereas virtually solid yellow areas are usually program areas.
- HELP** Displays a list of available commands along with the version number of the Hackit in use.
- LOAD** Loads a file from tape/disc after the filename has been entered, an address can be specified as to where the file will load into. Pressing the return key at this prompt will load in the file at the address specified in its header.
- HEADER** Will display header information of the specified file on tape/disc.
- SAVE** Will save a specified area with a specified name onto tape or disc. Besides name, start address, length and entry address, and optional load address can be entered which will be the address that the file will normally load in at. It is therefore possible to save an area of memory from say 8000 that will normally load in at 4000. If you want the load address to be the same as the start address, simply press the return key when prompted for the load address. This option only works on files being saved to disc and not tape.
- COLOURS** Displays the 16 available inks and what they are set to. Can be usefull if a routine is called and on returning the inks need to be examined.
- MOVE** Moves a specified area of memory from one address to another. The original block of memory is filled with 0's as soon as it has been moved.
- COPY** Performs a similar job to move except then original block of memory remians intact.
- FILL** Fills a block of memory with a specified value. Can be used to clear areas of memory.
- DISASSEMBLE** Disassembles the relevant area of memory into Z80 mnemonics. Besides a start and end address, if a filename is given, an ascii file will be created on tape or disc containing the disassembly. This can then be loaded into an assembler for re-assembling.
- HEXTODEC** Will convert 16 bit hexadecimal (0-FFFF) numbers to decimal.
- DECTOHEX** Will convert decimal numbers in the range 0-65535 to

hexadecimal.

**SEARCH** Will search for an ascii or hex string in memory. Any addresses that contain occurrences of the string will be displayed. At least one occurrence will always show, this is because the original string must be stored in memory for comparison, and the search will obviously find this string.

**PRON** Printer On. This command will print onto your printer any data that Hackit will display on the screen. The only exception to this is when using the EDIT command. However, pressing the CONTROL and COPY keys together at anytime will perform a screen dump.

**PROFF** Turns off the above mentioned printer on command.

**BANK** Will allow access to the 2nd 64K bank on an 6128 or expanded 464/664. Only 16K blocks can be paged in at address 4000-8000 and are numbered 1-4. Up to 16 of these 16K blocks can be accessed if a 256k memory expansion is fitted.

**ALTERNATE** Please note that this command will only work on a 464/664 with a DK Tronics 64K memory expansion. It will not function correctly on a 6128 or standard 464/664. On an expanded 464/664, it is possible to get a program running in the second bank of memory and let the computer think it is running in the normal bank of RAM. Once the program is running, it is possible to enter Hackit (by pressing the reset button) and the whole of the 2nd 64K bank will still be intact. This includes the screen memory etc. After typing in the ALTERNATE command, the cursor will disappear. At this stage turn off the HACKIT and press the CONTROL-SHIFT and ESCAPE keys together. Anything displayed on the screen will become jumbled as the mode is changed. At this point you must type in the load instructions to load in the game. You will not be able to see what you typing, but the computer will still be accepting the information. Once the game has loaded, turn on the Hackit and press the reset button to access the hackit. if you now use the BANK command to select bank 4 and then do a COPY command to copy from 4000 bytes from address 4000 to address C000 then the screen from the game will be copied onto the current screen. The rest of the games code can be edited by examining banks 1,2 and 3 of the 2nd 64K of ram.

**OUT** Outputs a byte to a port.

**IN** Reads in a byte from a port.

**CLEAR** Fills memory from 40 to AB80 and all extra banks of memory with 0.

**RESTORE** Restores the colours to black on white. Usefull if you call a routine that sets the colours to an illegible combination.

**POKE** As the basic POKE command

**PEEK** As the basic PEEK command.

**ASSEM** The assembler is an in-line Z80 assembler and assembles a line of code when it is entered. It is only a basic assembler and does not support labels. All instructions must be entered in UPPER CASE and all numbers must be preceded by a & sign. All numbers must be signed hex integers. To quit from the assembler, simply enter Q as an instruction. Example instructions:-

CALL &3200	BIT &1,L	LD A,(IX&28)	DEC (HL)	JR &34
JP &5000	OR &78	INC DE	RES &6,A	DJNZ &56

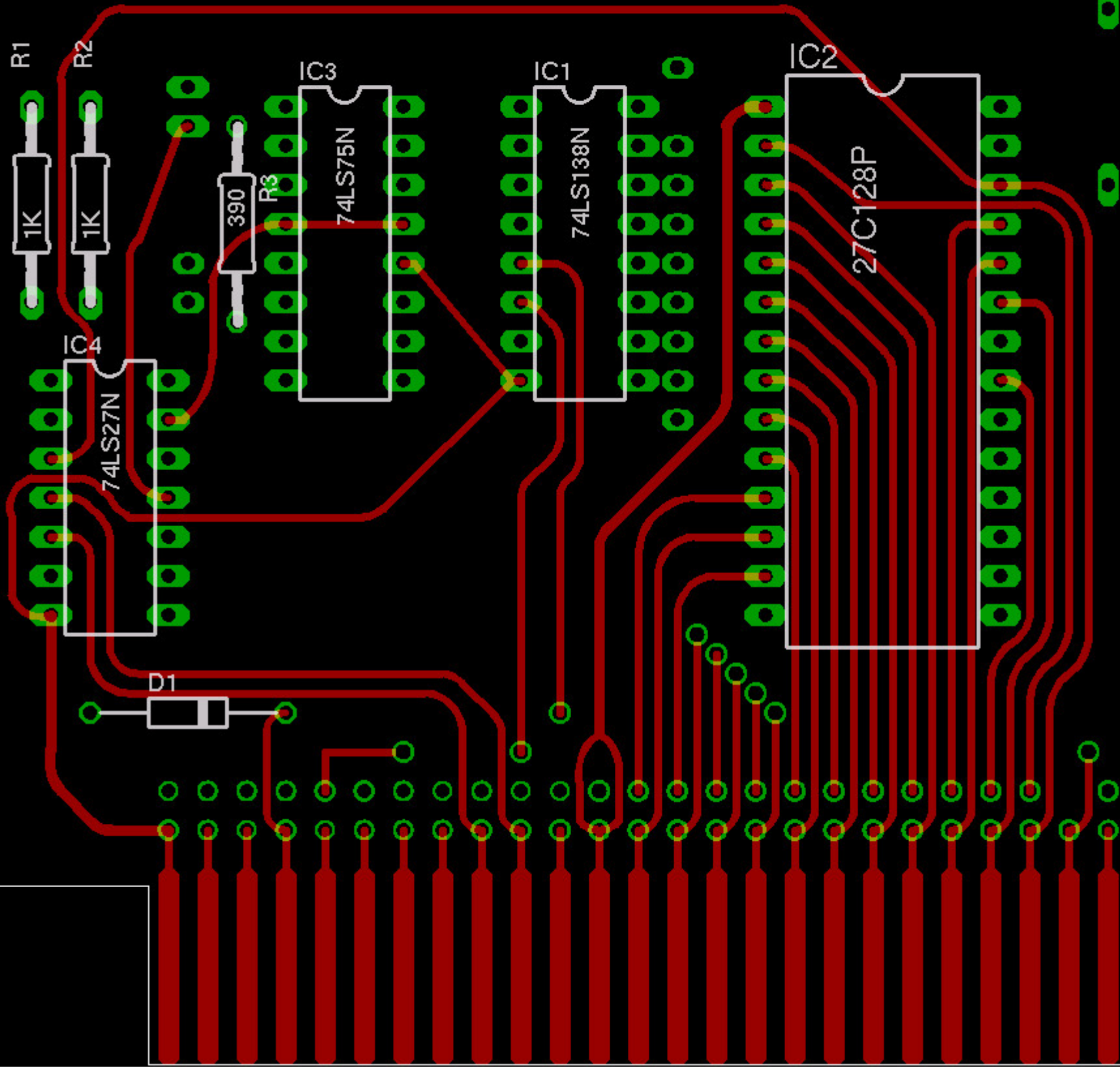
Obviously in order to make the most of HACKIT it is important to have a working knowledge of machine code. Hackit has not been developed in order to facilitate automatic tape to disc transfer, it has been developed to enable users to stop

and hack programs. Obviously with the ability to save areas of memory, it may be possible to copy programs, however this is not the prime reason for designing Hackit.

Some programs will not work with Hackit turned on. To remedy this, load the program with Hackit turned of, and turn it on when the program has loaded. Because any program can be stopped at any time, even loaders can be stopped to be disassembled to see how they work.

We hope that you enjoy using Hackit.

<C> SIREN SOFTWARE 1978.



HACH-IT V1.0 AMSTRAD CPC

