TRS8BIT

PRODUCED BY TRS-80.ORG.UK

HELLO EVERYONE, AND WELCOME TO THE MARCH 2016 EDITION; THE START OF OUR 10TH YEAR.

I'VE QUITE A FEW LITTLE 'SNIPPETS' OF NEWS AND INFORMATION, SO LET'S GET TO IT!

GOOD NEWS FOR THOSE WANTING MORE OPTIONS TRYING TO READ MARGINAL 5.25" TRS-80 DISKETTE MEDIA.

I READ THAT THE CATWEASEL CONTROLLER IS STILL PROBABLY THE BEST TOOL FOR READING OLDER TRS-80 DISKETTES, ESPECIALLY WHEN USED PROPERLY, BUT THE KRYOFLUX SEEMS TO BE CATCHING UP.
AN UPDATE FROM MIKE GORE AND KEIR FRASER ALLOWS CONVERTING THE KRYOFLUX CT STREAM INTO THE JV3 FORMAT SUPPORTED BY MANY TRS-80 EMULATORS.
FURTHER DETAILS AT -

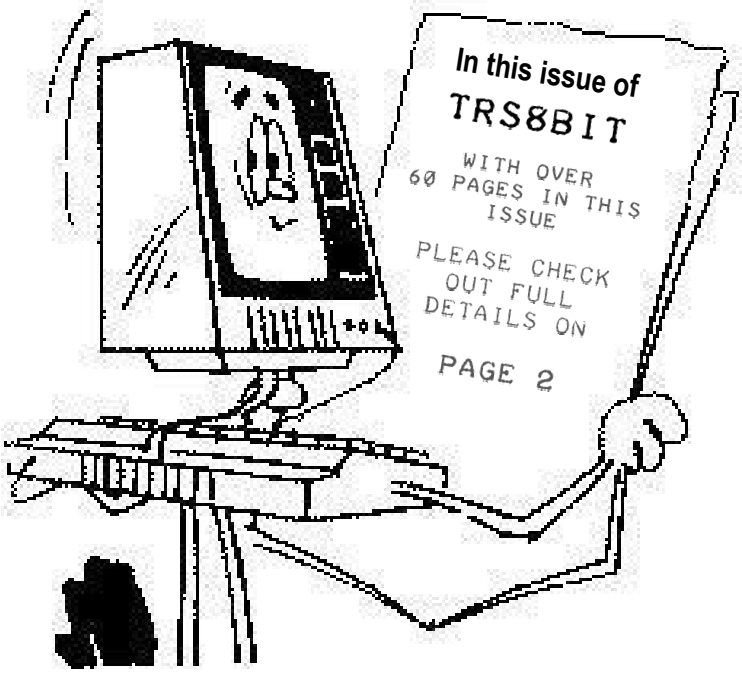
http://www.forum.kryoflux.com/viewtopic.php?t=1126

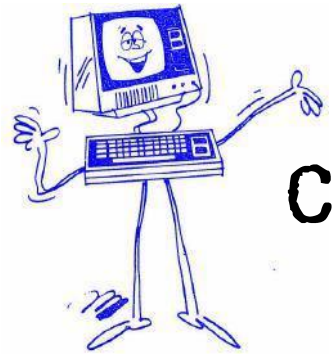
EBAY. VARIOUS ITEMS HAVE CAUGHT MY EYE OVER THE PAST FEW MONTHS

DO YOU REMEMBER THE TANDY 'SCREEN PRINTER'? WELL, THIS HAD A STARTING PRICE OF OVER £1000! I DON'T KNOW IF IT SOLD, BUT IT WAS A BIT BEYOND MY PURSE LIMIT DEE SAID!

ALSO, THERE'S VARIOUS 8K MEMORY UPGRADES FOR THE MODEL 100. IT'S ADVERTISE AS 'NEW'. WHETHER IT'S NEW, OLD STOCK, OR A FULL

(CONTINUED ON PAGE 56)





CONTENTS

IN MAV'S WORKSHOP:

"TRANSFORMERS" EXPLAINED

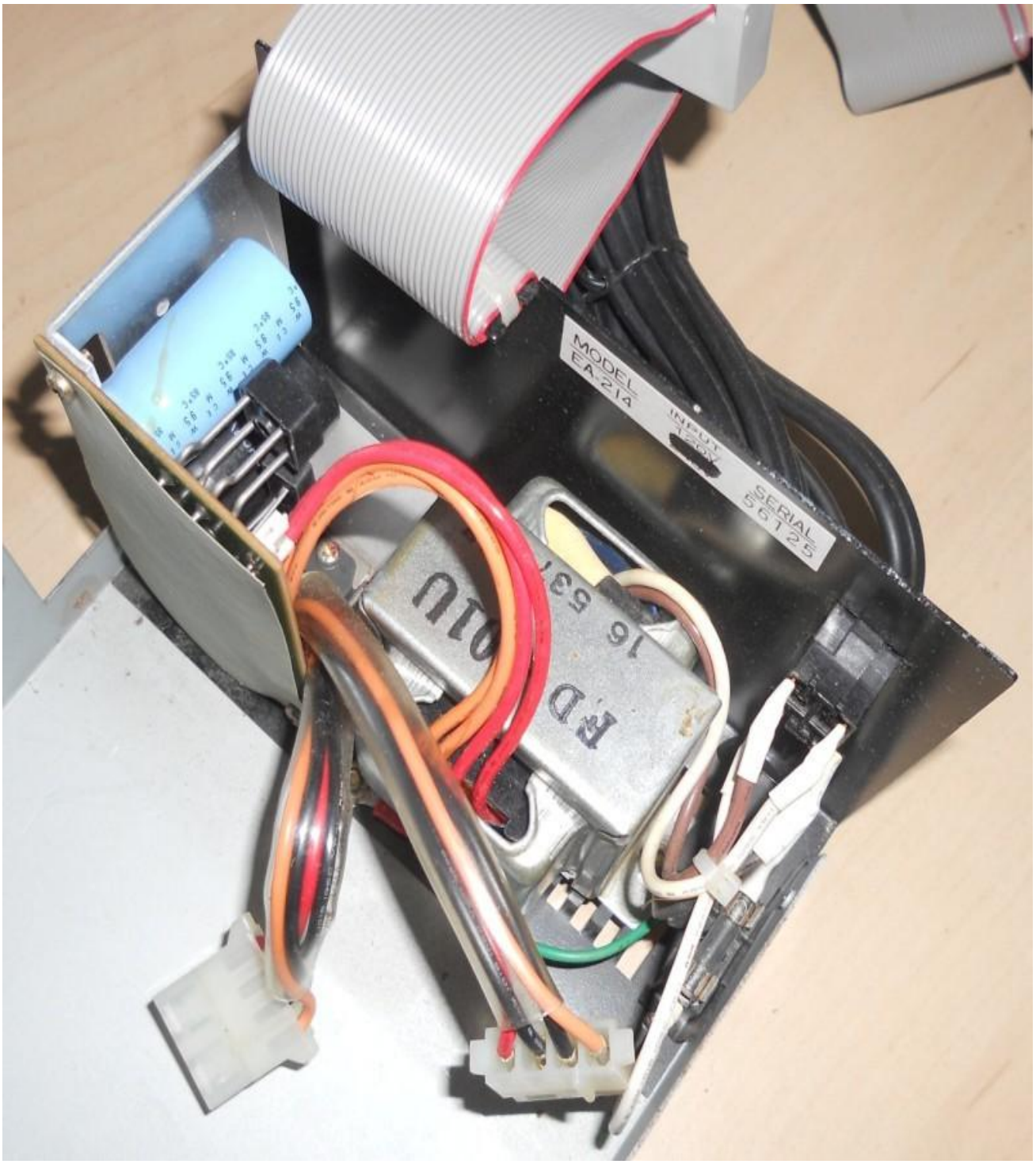
How to change 120V TRS-80 External Disk Drives to 240V

by Ian Mavric

Being an American computer, the TRS-80 is still one of the best value vintage computers which can be purchased in the USA, and if you are patient, shipping to other places around the world is reasonable. This is a real bonus if you live in a country where TRS-80s aren't very plentiful (the UK and New Zealand are good examples of this) , or if they are around they are usually in poor shape (Australia).
Trouble is that as delivered from the USA they will be 120V and this presents its own set of problems in 240V countries.
Fortunately for people who don't mind clutter, a 240V-120V step-down transformer can be bought on eBay for $35 or less. However my place is messy enough so my preferred way is to replace the 120V transformer inside the case itself.

The Tandy FD-501 Coco Disk Drive Power Supply:

I chose the FD-501 as it's representative of all later Coco disk drives: FD-500, 501 and 502 are more or less the same. Mains AC is fed into a transformer which has a single primary winding (120V) and two secondary windings (8.4V and 14.75V). Notice that they are independent of each other, not multi-tapped from a single secondary winding. I'll refer to this again later.

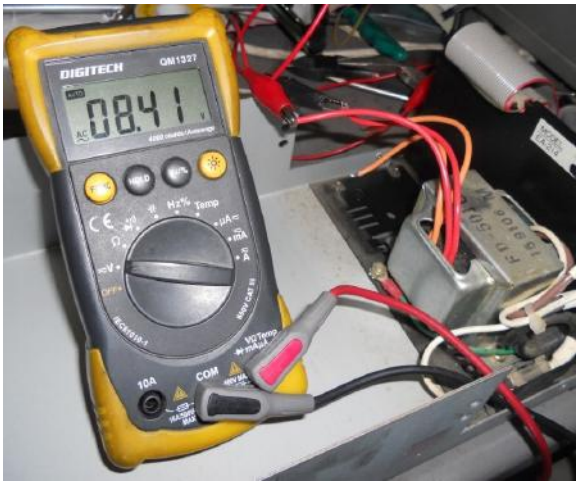


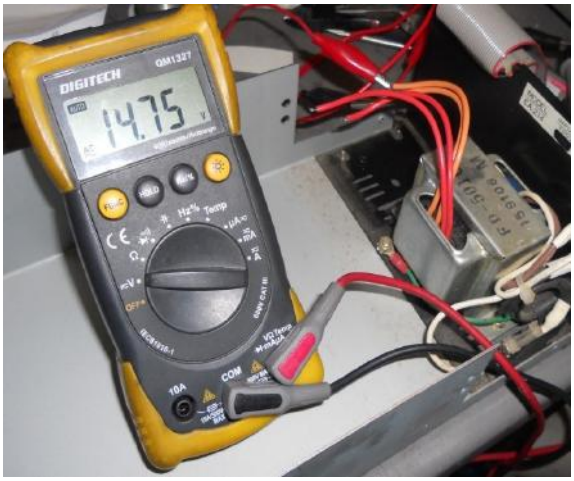
Picture: shows transformer and is joined to regulator via red and orange wires

Next these two low voltage AC supplies go into a small circuit board with a few components. This is the voltage regulator board and it's purpose is to take the two raw AC voltages, turn them into DC, smooth them out and regulate them down to the required voltages to feed the disk drive: 5VDC and 12VDC. This is accomplished by firstly a bridge rectifier to turn the AC to DC, a large capacitor to smooth out the resulting DC's ripply current delivery, and then finally feeding to a voltage regulator IC which maintains a constant output voltage. The theory behind the voltage regulator IC is that regardless of how much the mains AC dips or spikes, the output voltage fed to the disk drive is constant. On the voltage regulator board there are two such circuits, one for 5V and one for 12V, each with it's own bridge, capacitor and voltage regulator IC. They are kept separate from each other though they are on the same board.

Transformer Substitution Principle:

Ideally what we need then is to replace the existing transformer with one which has a 240V input and two secondary windings with outputs around 8.4V and 14.75V like the last one. In practice it's very hard to find a drop-in replacement. It's tempting to buy a multi-tap transformer and tap the 7.5V and 12V points, but this won't work as it ties together the two independent circuits on the voltage regulator. It's not designed that way and will short and damage the whole voltage regulator board. The only way to achieve a satisfactory result is to use two small transformers, one feeding the 5V side of the voltage regulator board and one feeding the 12V side. Because of the physical space restriction inside the FD-501 case, the two transformers need to be small, but not too small, so we need to calculate how much current is needed.



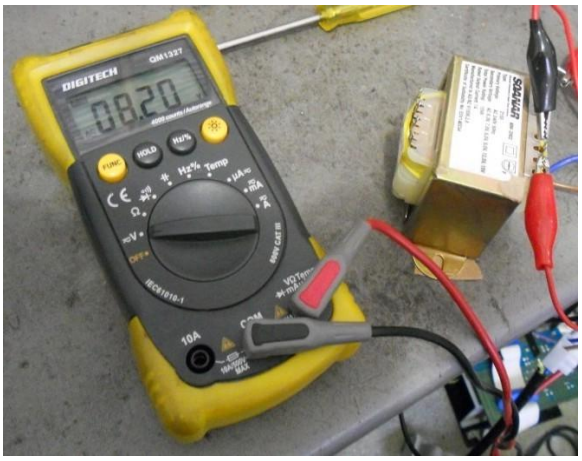


Pictures: shows the AC outputs of each of the two secondary windings

Disk Drive Power Requirements:

The disk drive in the FD-501 unit as a TEC FB-501 which is surprisingly difficult to find a technical manual for. Its performance characteristics are very similar to the TEAC FD-55 series of disk drives, and I do have access to the technical manual for that disk drive. If nothing else it gives me an indication of what a typical 5.25in half-height disk drive needs to run: 12VDC @0.25A, and 5VDC @0.4A Given that we might want to run two disk drives we need to have around 0.5A on the 12VDC and close to 1A on the 5VDC.

Now that we know our power requirements we need to find transformers which meet the requirements. Depending on your retailer of electronic parts, they may have a small or large selection of transformers to choose from. My local parts stockist is Jaycar Electronics which carries a small selection of transformers, and two of them fit the bill. On the 12V side I used a MM-2013 which delivers 500mA at 12.6V and on the 5V side I used a MM-2002 which delivers 1A at 7.5V Interestingly the MM-2013 is the physically smaller of the two, the MM-2002 is much larger and although it produces less voltage in our application the disk drive relies on it for more amps.



Pictures: show the outputs of the two transformers I will use in the substitution

Disk Drive Power Requirements - Part II

It should be noted that disk drives don't use power at the above amperages constantly, especially on the 12V side. A typical disk drive like the TEC or TEAC will sit in its halted state drawing about 250mA on the 5V (to power it's logic board) and virtually nothing on the 12V (12V being used mainly for the disk drive motors - stepper and rotation motors). When a disk drive is called and the diskette needs to be quickly spun up to speed from standstill, the 12V pulls maximum current, but once the diskette is up to speed, and this only takes a fraction of a second, 12V current draw is minimal.

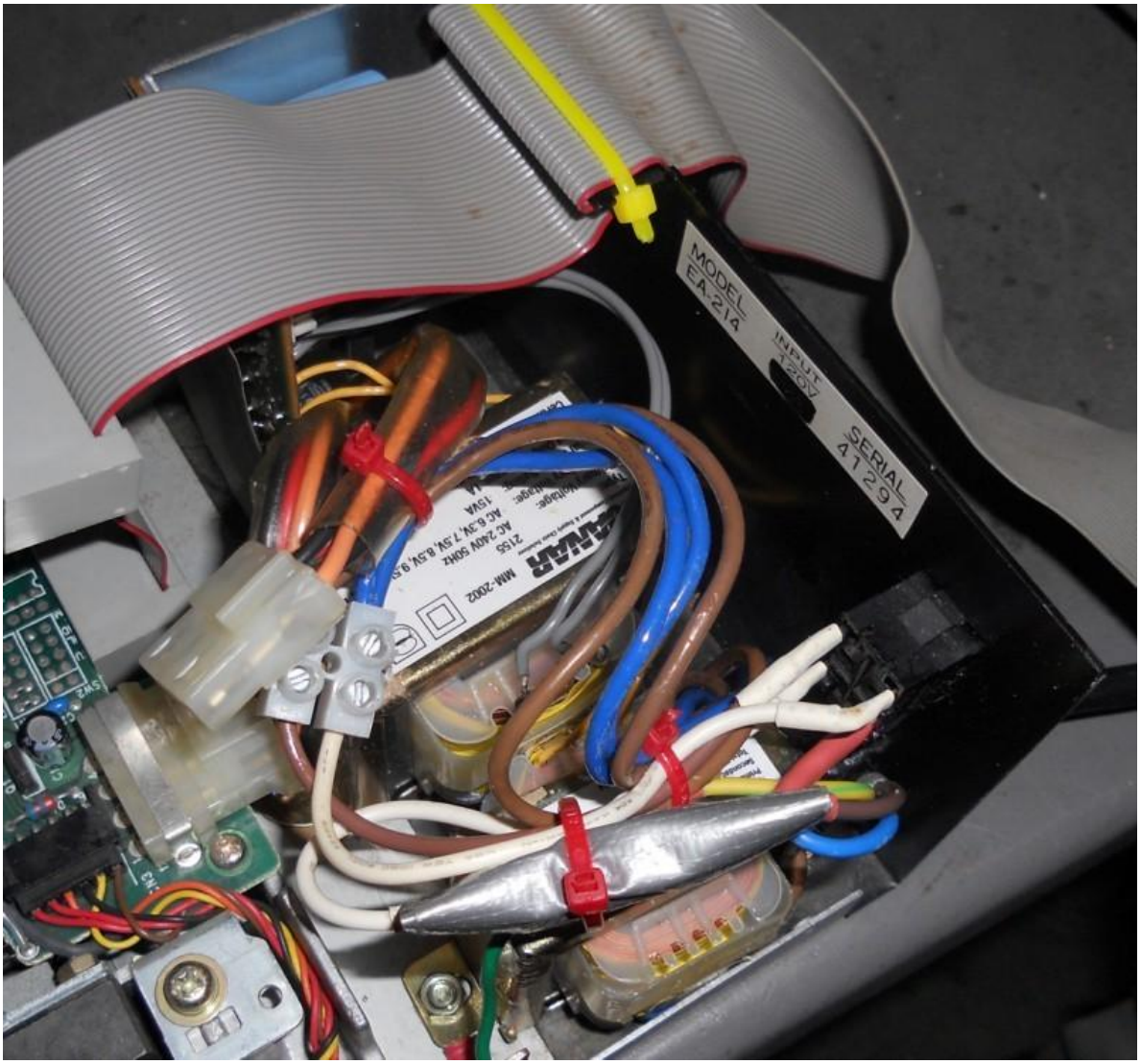
Final thoughts on selecting replacement transformers:

The design of the voltage regulator board and the regulators used (the final stage before being fed into the disk drive) means that AC voltage from a fairly wide range can be fed and excess power will just be regulated out in the form of heat. It's why the voltage regulators are mounted to a large aluminium heat-sink which sits near the cooling vents on the disk drive case. You can feed each circuit slightly higher voltage than the previous transformer but conversely it will also take a slightly lower voltage. In my example above I am feeding 8.2V where previously the transformer fed 8.4V, and feeding 13.7V where previously the transformer fed 14.75V

It's OK to do this as long as you don't go too low in voltage (disk drive won't work properly) or too high (voltage regulator PCB will overheat and die). In practice around one volt lower than the previous transformer works well and is what I aim for.

Mounting the Transformers and Finishing Off:

I tried a few different arrangements inside the case and found it works best in my instance if the voltage regulator PCB stays in its original place, and if using two different sized transformers, mount the smaller one near the size, basically under the power switch, and the large one in the middle. You are powering two transformers so the mains input needs to go into a splitter to feed both transformers, but keep a fuse in the circuit and make sure it's before the split. Other than that you just connect the output of each transformer to the appropriate points on the voltage regulator board. Cable ties neaten things up, as does clipping wires to an appropriate short length to minimise clutter inside the case.



Picture: shows two transformers mounted and wired up

Note To Would-Be Transformer Upgraders:

BE CAREFUL - if you are unsure of anything don't try it or get someone in the know to do it for you.

Power supplies can and do contain lethal voltages and I don't want anyone, and can't be held responsible, for anyone who hurts themself.
TREAT MAINS POWER WITH CARE,
BECAUSE IT DOESN'T CARE ABOUT YOU!

Ian Mavric
Ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

INSTANT SORT/SEARCH DATABASE - UNBELIEVABLY FAST

Everything in electronics takes a finite time, consequently nothing can be instantaneous. However a database that will search 500 records and sort the names into alphabetical order in 1½ seconds, that will go on to do the same thing with 1,000 names in only 2½ seconds, is fast. If you add to that the ability to search 500 or 1,000 records for a specific range of names or ages or sexes or whatever, in such a small amount of time that it is not worth timing it, then the program deserves to be described as instantaneous. Especially as these times are attained on a standard Level II TRS-80. These results are achieved, obviously, by some very clever machine language coding. This however is not enough. After all GSF from Racet will sort 1,000 arrays in about 11 seconds and that is indeed a clever program. No, in order to achieve the results required from this program it is necessary to change one's entire overview of database. There are many databases available for the TRS-80 now. All of them have been designed to store as much data as possible, as easily as possible. Not as an afterthought, but nor as a prime design requirement, they have also incorporated as fast a sort as was practicable. This program was designed from the outset to achieve unbelievably fast sort and search times. Indeed we do not recommend this database for application in which fast searching or sorting is not a prime requirement. And what are the applications? It's a hackneyed phrase to say that they are limited only by the user's imagination, but that's about it. Let's take an example. Suppose you are running a marriage or data bureau. An ordinary database will file all the names and addresses away together with the necessary information as to sex, age and so on and with some you would be able to sort the list, so that only people with similar characteristics were eventually obtained. With this database you could, for instance, file the name, sex, age, category of hobby, category of chief interest, vital statistics and other data so that at the touch of a button you could instantaneously display on the screen all women of a certain age with certain vital statistics, living in a certain area. You could also display men with similar (excluding the vital statistics!) data that fall into similar categories. And all of this almost instantaneously. Not everybody runs a marriage bureau, but other applications are not hard to think of. Estate agents can file details of property away so that they can instantaneously obtain data on houses in a certain area or of a certain size. Doctors can reach information as to patients with similar diseases, ages or whatever immediately. In the home, a record library can be stored and every record by a certain composer written in a certain year can be accessed without delay. The list of applications is endless. For any use where it is important to extract information within a certain range or it is important to sort information, this database will find a use. The prime commands and features of this program are as follows:

Datafile creation

1. Create a file
2. Add a record
3. Delete a record
4. Display a record
5. Tape a file
6. Amend a record
7. Display the file data
8. Load a tape

Sort/Search

1. Sort up or down
2. Page forward or backward
3. Select a range for search
4. Select or exclude a category
5. Select or exclude on initial letter
6. Resort records in a sort
7. New sort all records
8. Extended sort
9. Arithmetic
10. Display file data
11. Load a tape
12. Printout sorted data

The data is displayed in columnar form and the data may be alphabetical, alphanumeric, integer or decimal. The number of columns is from 2 to 10 and the records may contain a maximum 44 - 60 characters depending upon the number of columns used. Columns may be of any width within the screen capacity but integer or decimal columns more than five and six characters wide respectively will not have the option of searching within a range. The program consists of two parts. The first is used for entering the data and the second for the sort or search. The second part overlays the first when it is loaded so only 4K of memory is used by the entire program. The remainder of your memory space is available for data. The amount of data that can be contained will of course depend upon the amount of memory available, but as a rough guide a 16K user will be able to manipulate at one time 250 records of 39 characters each or 514 records of 17 characters each. As a further rough guide on sorting speed, the time to sort 1,000 records on fields of random strings of random length, or of random number between 1 and 99,999, averages under 2½ seconds. Numeric columns either integer or decimal may be arithmetically manipulated almost instantaneously. A total may be cast or an average taken for any numeric column up to five digits. This is so fast that when adding 1,000 numbers totalling over 50 million, only a slight hesitation can be noticed before the total is given. In summary, therefore, this program is ideal for any application concerning the manipulation of information whether it be business, personal or hobby which can be comfortably displayed as one record per line upon the screen and in respect of which it is required that super fast searches or sorts be carried out. The program is supplied on cassette. At this time it is not compatible with disk systems. A disk version is in the course of preparation. The cassette includes a set of data randomly generated which can be fed into part 2 of the program to demonstrate the fantastically fast sort and search features. Also available for the Video Genie.

SYDTRUG Meeting News for Saturday 14th November 2015

By Errol Rosser, SYDTRUG Librarian

(Photos courtesy of Ivan Kennedy)

Committee meeting

Present :-

President, Secretary, Hardware Coordinator, Membership Secretary, Public Officer, Treasurer, Librarian, Web Editor and Newsletter Editor

Left to right Errol Alex, Peter, and Debbie

Meeting Details :-

Started at 13:10, previous minutes were accepted, correspondence noted and various committee reports were given. $0.00 income and $0.00 expenses for the previous month. The treasurer submitted the group's balance sheet for 2014-2015. Income for the year was more than $200 LESS than the costs (hall hire, Fair Trading fees and insurance)

Discussion on web site updates – additions in the last month :- 200+ more exchange newsletters,1 GB+ of Model II manuals and 8" disk images, emulators for Coco, M100, Models I, III & 4, CPM and more TRS-80, MS-DOS & Windows software & manuals, new general Discussion Forum. The on-line library now has 35,162 files (76.7 GB) as 1566 hardware user & service manuals (3.1 GB), 5033 software manuals and programs (8.2 GB), 1874 magazines (46.9 GB), 476 books (6.8 GB) and 6,270 exch. newsletters from 137 clubs (12.5 GB)

New Business :-

Discussion on the upcoming AGM, distribution of the AGM notice, proxy form, balance sheet and notice of the proposed motion to suspend meetings at Regents Park. Some of the alternatives to the suspension of meetings were discussed as well as the content of the notice of motion. The primary concern is the continuing decline in financial members and the increasing costs of hall hire and public liability insurance resulting in the group not having the funds to pay the public liability insurance in February 2017

General Meeting

Three non-committee members attended today - Jenny (arrived with Alex), Colleen May, and Ivan Kennedy. Ex-member Garry Howarth visited with his System 80

Peter was busy on his laptop, Jenny was working on her iPad while Alex helped Colleen with her new printer

Left to Right – Colleen, Alex and Jenny

Garry was testing SD Card floppy disk emulators on his System 80 with a VGA LCD instead of the usual CRT monitor.

Left to Right – Garry, his System 80 and me

I set up my Model I and demonstrated the MISE (Model I System Expander) from Peter Bartlett (also using an LCD screen instead of a monitor)
Ivan had his Model I with a FreHD hard disk emulator

Left to Right –Ivan's Model I with FreHD, and my Model I with MISE

Ivan borrowed a spare MISE unit from me for evaluation at home.

Meeting was closed, very reluctantly, at 17:05

While I was getting the exchange newsletters last month, I saw the photo (below) of a computer group meeting. This is typical of the majority of the other computer groups.
Note there is no where to put your own computer. It is just a lecture/presentation style of "meeting", NOT a hands on, bring your own PC to demonstrate or get help like our meetings have been since the very first back in 1978

Reprinted from "SYDTRUG News C/- Peter Wignell PO Box 95 Narwee NSW 2209 Australia"

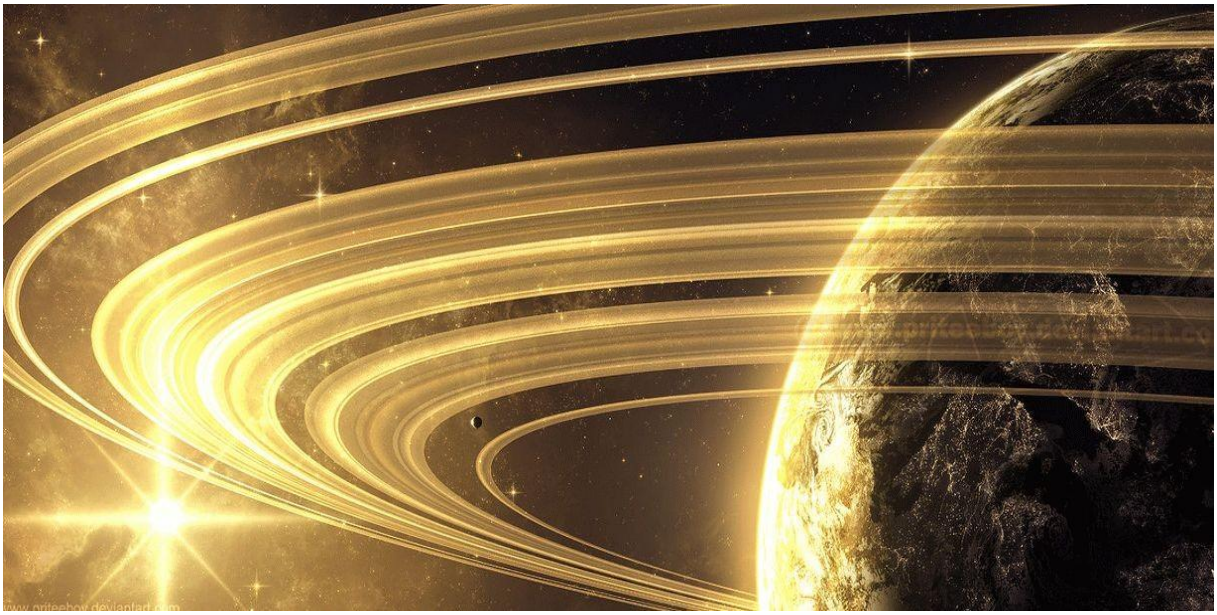
LOOKING FOR FAST, INEXPENSIVE, UNLIMITED MASS STORAGE FOR YOUR TRS-80 MODEL I/III/4/4P/4D?

The amazing

"FreHD"

-Emulates a TRS-80 hard drive, but faster than any hard drive!
-Works with your favourite DOS (LS-DOS, LDOS, CP/M, Newdos/80 2.5)
-Uses SD card for storage medium
-Bonus free Real Time Clock function!
-Designed in Belgium and proudly built and shipped from Australia
-Kit form or fully assembled

Order yours today:
http://ianmav.customer.netspace.net.au/trs80/emulator



THE NAMING OF PLANETS

ET FONEHUME

DUSTY, DURING THE CHRISTMAS HOLIDAYS, I TOOK MY GRAND CHILDREN TO THE CINEMA TO SEE THE LATEST EDITION OF 'STARWARS'. WHILST THE COMPUTER GENERATED GRAPHICS WERE QUITE EXCEPTIONAL AND WE ALL THOROUGHLY ENJOYED OURSELVES, I COULDN'T HELP BUT THINK IT WAS VERY SIMILAR IN CONTENTS TO THE ORIGINAL EPISODE FROM THE LATE 1970'S.

AND THEN I GOT TO THINKING OF ALL THE 'STARWARS' AND 'SCI-FI' GAMES THAT WERE WRITTEN FOR THE MODEL 1. DO YOU REMEMBER 'XWING'? THIS LED, IN TURN, TO RECALLING PROGRAMS TO PRODUCE 'ORIGINAL' PLANET NAMES FOR ALL THOSE PROGRAMS WE WERE BEAVERING AWAY AT! THERE WERE QUITE A NUMBER WRITTEN AT THE TIME, ALL NOW LONG GONE, SO I THOUGHT, HOW ABOUT A ONE-LINER TO RECALL THOSE EXCITING TIMES.

```
10 DATA AR,RA,CO,ET,TE,LO,OL,ES,ED,XA,AX,TO,OT,OM,OB,OV,VO,OZ,ZO
,IX,XAL,PHO,REW,RU,UR,SAC,ETH,COM,TEP,CER,KOL,SKO,XIN,SH:DIMA$(3
4):FORX=1TO34:READX$:A$(X)=X$:NEXT:FORX=0TO1STEP0:N$="":S=RND(3)
+1:FORN=1TOS:N$=N$+A$(RND(34)):NEXT:PRINTN$:INPUTZ:NEXT
```

LOETOMTEP TOOLTO SACED CERSHCOOV TOOLSKO OZSH TEPKOLOBOZ
RUVO COSAC LOSH SHAX SHZO COOVCO TOTEPCO ESVOXA ESAXTO
XALAR ARETH URARXASAC TEPHOSHAX ETHCERXA ZOTO OMOZ ETHETS
H URAXOTED ESIX TEPIXCO KOLOZXALES ZOCOMLO ETHOZ AXXASAC
ETCOOZ URCEROTOT OVTEPTO RAOLXAOL VOOL REWRATO PHOPHOAX
LOUROZCOM ETZO TEOZTO OMKOLXINOT VOXAESXIN AXXATE TOAR ES
RUUR TOOVOZ TETEP TOIXETED COURARRU RUZOOBOM COMTE LOSKOA

EMAILS TO THE EDITOR

I RECEIVED THIS EMAIL FROM RICHARD LOXLEY, DURING JANUARY, WHICH I HOPE YOU FIND OF INTEREST. RICHARD'S SECOND ONE-LINER, WHICH I NAMED 'MAZE DODGE' (AFTER HIS ORIGINAL PROGRAM MISSILE DODGE), IS FEATURED ON THE NEXT PAGES COMPLETE WITH A DESCRIPTION AS TO JUST HOW IT WORKS. ONCE AGAIN, IT'S A 10 PUT OF 10 ONE-LINER!

AS FOR THE OTHER QUERY WITH HIS VIDEO GENIE, I SUGGESTED CONTACTING BAS AT GAMMA COMPUTERS. I WONDER IF HOOKING UP A FREHD WOULD WORK?

Hi Dusty,

I was just doing a Google search regarding a program I wrote, and found your TRS8BIT PDF (vol 7 issue 4).

In it you feature my Missile Dodge program but said you couldn't trace the author. Well I thought you might like to know that it's me :-)

I published it on YouTube, but it seems a couple of sites ripped the content and republished it as if it was their own, hence the confusion.

The original video and description is here:

https://www.youtube.com/watch?v=lWyZO-yPnvM

I also did a follow up here:

https://www.youtube.com/watch?v=siu_f2Jcllc

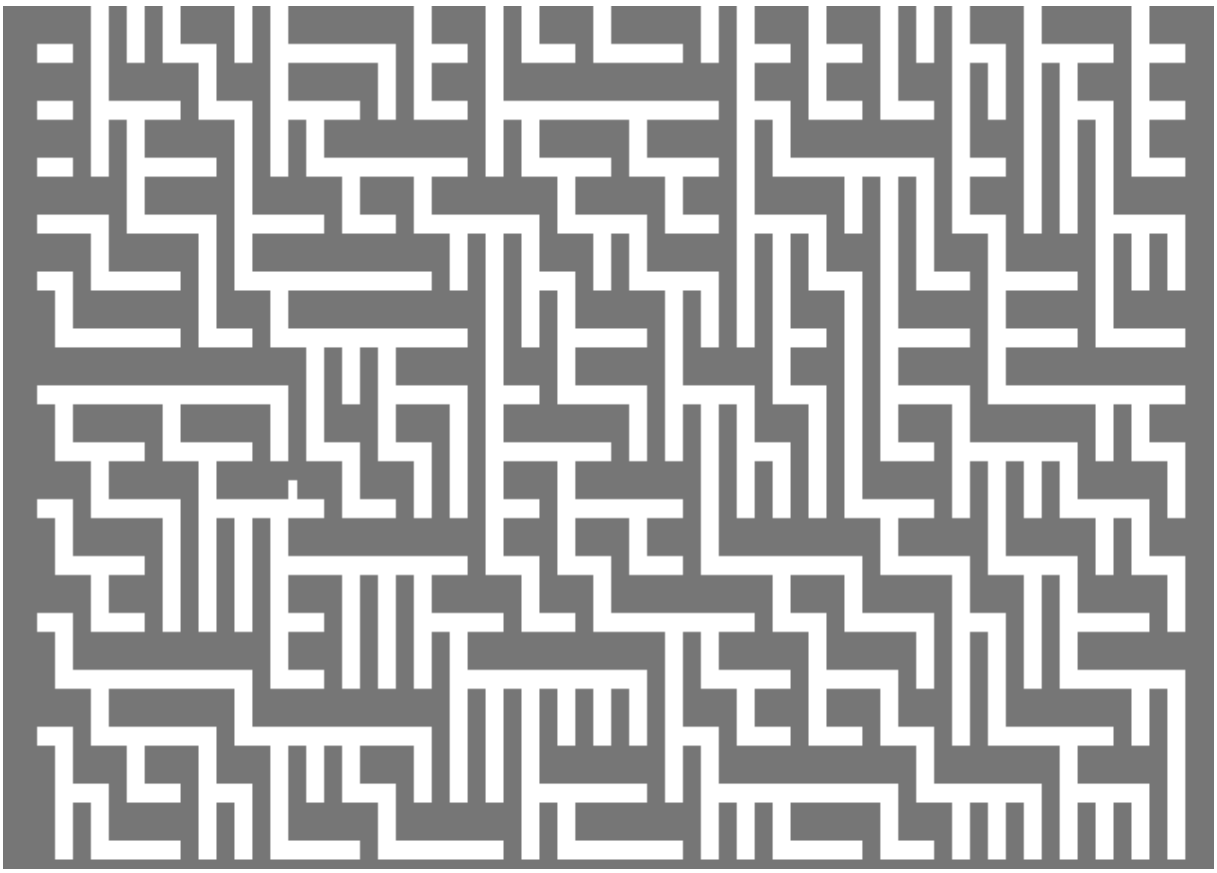
I was just looking around as I'm playing around with TRS-80 emulators at the moment, as a warm up to hopefully getting an old Video Genie up and running soon (my old one no longer works, but I've got a lead to get another that hopefully works, or will at least help me to fix my existing one!)

Then I'd like to see if I can get something to use memory cards instead of cassettes, but most of the information out there seems to be around TRS-80s rather than Video Genies/System 80s, which use a different expansion interface. Still looking ... do you have any leads that could help?

Regards,

Richard

```
0 IFC=0THENCLS:FORC=1TO480:R=RND(2):PRINTCHR$(320-144*R);CHR$(16
1+15*R);:NEXT:GOTOELSEK=PEEK(14400):P=X+(K=32ANDX>0)-(K=64):Q=Y+
(K=8ANDY>0)-(K=16):IFPOINT(P,Q)THENGOTOELSERESET(X,Y):X=P:Y=Q:SE
T(X,Y):IFX>126ORY>44THENPRINT"WELL DONE"ELSEGOTO
```



Following on from Missile Dodge (see my other videos), I decided to write a new one-line game from scratch.

One thing that always bothered me about my original one line games was the lack of playing area - there wasn't enough space in one line to draw anything useful. I really wanted to make a Pacman type game, but could-n't find a way.

I wondered if I could draw a Pacman-style maze using a pseudo random number algorithm.

I haven't found that yet, but I did discover the Binary Maze algorithm (see http://en.wikipedia.org/wiki/Maze_generation
And http://www.astrolog.org/labyrnth/algr...).

So here's a game which draw a maze, and then lets you try to escape out the bottom or right edges (the binary maze is always guaranteed to have a route to an exit on the bottom or right edges).

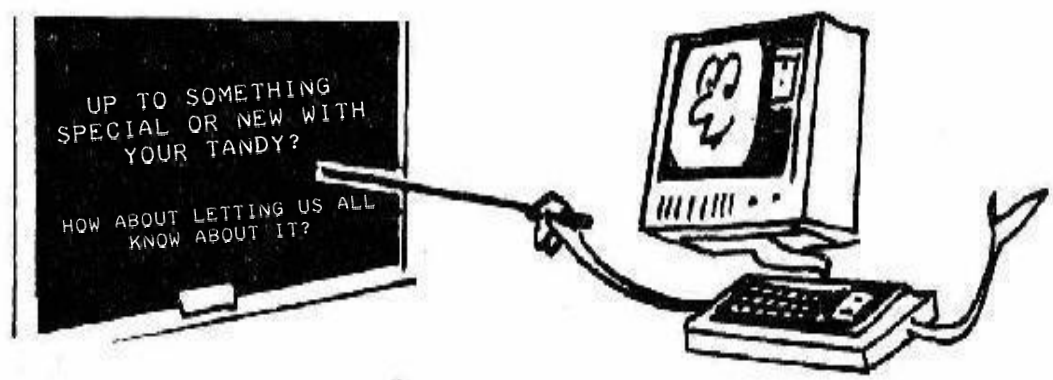
Use the arrow keys to move. NB. If the emulator you use has an option to map the Windows keyboard to the TRS-80 keyboard you may need to turn it off to get the arrow keys to work properly.

```
0 IFC=0THENCLS:FORC=1TO480:R=RND(2):PRINTCHR$(320-
144*R);CHR$(161+15*R);:NEXT:GOTOELSEK=PEEK(14400):P=X+
(K=32ANDX>0-)-(K=64):Q=Y+(K=8ANDY>0)-(K=16):IFPOINT(P,Q)
THENGOTOELSERESET(X,Y):X=P:Y=Q:SET
(X,Y):IFX>126ORY>44THENPRINT"WELL DONE"ELSEGOTO
```

A couple of explanatory notes:

(1) The shenanigans with CHR$() on the convoluted expression is to select one of two graphics characters without using an IF THEN clause. For details of the graphics characters see http://www.trs-80.com/wordpress/zaps-... The first is selecting between a space (ASCII 32) and a horizontal line (176). The second is selecting between a full block (191) and the horizontal line (176).

(2) PEEK(14400) is looking at the keyboard memory map, and picks up the arrow keys. See my Missile Dodge video for details and links.



HI DUSTY

ITS BEEN A LONG TIME. I STILL HAVEN'T GOT MY HI-RES MODEL 4 REVIEW DONE. I WILL. JUST TIME I GUESS.

BUT - HERE'S A FEW LINKS THAT I HOPE MIGHT BE OF INTEREST, I'VE SEEN THAT YOU PRINT TRS ADVERTS FROM TIME TO TIME. ARE THESE NEW TO YOU?

ADS:

HTTPS://NOSHER.NET/ARCHIVES/COMPUTERS?
TYPE=SOURCE&VALUE=TANDY/RADIO%20SHACK

THE NEXT INFO IS, I HAVE MANAGED TO FIND A FEW OF THE SAME ADS AND REVIEWS FROM PCW MAGAZINE. WOULD YOU LIKE THEM OR HAVE YOU GOT THEM ALREADY?

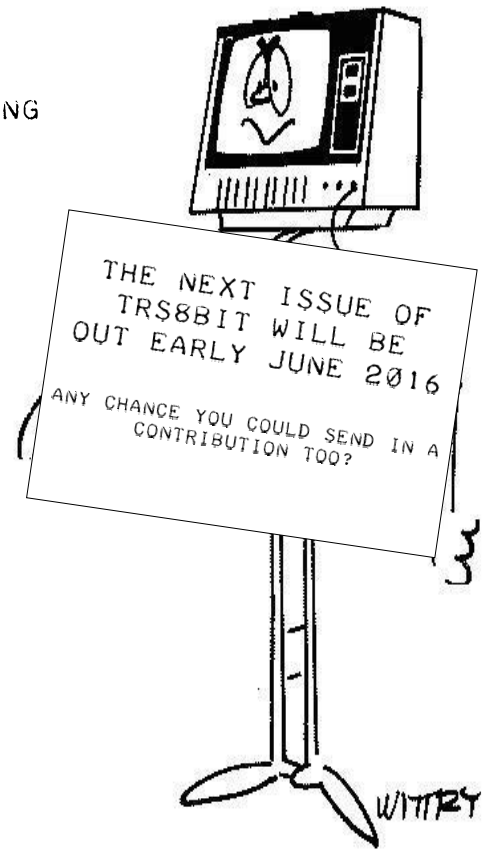
AND THE LAST INFO IS,
I HAVE FOUND A SUPPLY OF METAL GEARS TO FIX THE WORN PLASTIC COGS ON MY CGP-115 PRINTER. ITS FROM A MODEL HELICOPTER SHOP!
SHALL I DIG THE LINK OUT FOR THAT TOO?

I'M SEEING BAS TOMORROW, HE'S GOING TO FIX MY MODEL 4 PSU (AGAIN) AND GET SOME HIRES SOFTWARE ONTO MY SD CARD AND SORT OUT WHERE I WAS GOING WRONG.

OK, SPEAK SOON, MAYBE EVEN WITH MY REVIEW(S)

IAIN HANCOCK

THANKS IAIN FOR THE EMAIL. LOOKING FORWARD TO HEARING FROM YOU WITH YOU REVIEW, WHEN YOU GET A CHANCE, AND DETAILS RE THE METAL GEARS/COGS.



AS THERE HAS BEEN A NUMBER OF VIDEO GENIES SOLD ON EBAY OVER THE LAST FEW MONTHS, I THOUGH THAT THE FOLLOWING 2 ARTICLES, WHICH WERE ORIGINALLY PUBLISH IN NATGUG NEWS IN AUGUST 1983, MIGHT BE OF INTEREST. I CAN'T RECALL EVER SEEING ANY OF THE ROM KITS INSTALLED AND I CAN'T IMAGINE THERE ARE ANY KICKING ABOUT THESE DAYS. UNLESS, OF COURSE, YOU KNOW OTHERWISE!
I ASSUME THE ROMS USE THE 'SPARE' 2K AREA OF RAM AVAILABLE ON THE M1 AND THE VG. IF THIS IS THE CASE, THEY TAKE UP THE SAME AREA OF RAM THAT THE ACULAB FLOPPY TAPE OPERATING SYSTEM UTILISES.

REVIEW of GNOMIC FEATURE ROM "B" For the Video Genie

The Feature ROM'B' from General Northern Microcomputers contains several utilities that are normally loaded from tape as required. With all the bother that the procedure involves, such as CSAVEing your program first, setting MEM SIZE. CLOADing a renumber program or compress utility and then reloading your program etc.. etc.

The ROM sits in an area unused by the Tandy ROM, so takes up no user RAM (again, unlike most tape utilities). It contains the following aids

0 Return to Basic.
1 Keyboard Invert.
2 BASIC Renumber .
3 Single BASIC Keywords.
4 Flashing block Cursor.
5 BASIC program Merge.
6 BASIC program Compress.

These options are displayed by SHIFT/BREAK, once a short initialisation routine has been typed, in direct Mode. The number of the option required is then entered.

Renumber chances all line numbers, GOSUBs, GOTOs, ON GOTOs etc.. starting at 10 with increments of 10. Single keywords are perhaps a little strange.
(Pi=3.141526535898) would be attractive addition, though no doubt increase the cost, if the user could define these keywords when ordering. It's nice to see that those followed by brackets, such as STRING$, CHR$ etc. have the first bracket symbol included. These keywords are called by SHIFT/A-Z.
Flashing cursor changes the standard character to a Graphics block. CHR$(176).
Merge allows a BASIC program in Memory to be protected while a second program is loaded. Renumber is called automatically at this point! It would have been better to omit this call because, if merging a suite of sub-routines with defined line numbers, these will be changed.

Compress removes all redundant spaces and REM's.

Installing the ROM is a little more complicated than first envisaged.

Although the neat PCB it fitted with an edge connector to match the one on the rear of the TRS-80 keyboard, it leaves it 'flapping in the breeze' and consequently, easily damaged. So a case (one of Tandy's plastic boxes) will be required.
If you already use the expansion connector for a printer interface unit, there is a further expense in obtaining a splitter cable (available from GNOMIC). The power supply for the ROM is of the mains adaptor type combined with a 13 amp plug.

GNOMIC show a Feature ROM "A" in their catalogue and this raises another point. Although the ROMs are fitted into sockets on the PCB, this is an ordinary IC type and it would not be easy (or recommended) to keep unplugging and plugging ROMs. Neither does there seem to be much space on the PCB to allow fitting of a 'zero insertion force' socket. Such a socket could be fitted to the case, with a connector to the PCB ROM socket, but this is all extra cost again.

The instructions are rather poor copies and while covering the basic operating details, could be expanded. The instructions also fail to mention the Screen Print facility, (although in the adverts) which is called by holding down keys JKL. This can be done at any tine, even when a program is running. There does seen to be a slight bug here: if the last line displayed is

PRESS ENTER TO CONTINUE

when JKL are pressed, the last character (E) is erased and not printed. It's easy to overcome this. Just leave a space in the program line after the last character. If you have a full screen of graphics though, hum...

On the whole, once installed, the Feature ROM is well worth the cost and the tine saved over using tape utilities when developing long programs is a big bonus.

Pete Bamfield

MORE ON GENERAL NORTHERN MICROCOMPUTERS UPGRADES

Since buying my Genie System two and a half years ago I have added several

pieces of hardware as supplied by

GNOMIC,
8,Whitworth Rd.,
S.W. Industrial Est.
Peterlee, Co.
Durham.

The first addition was their lower case kit. Once installed this gives l/c (of course) with true descenders plus up arrows, down arrows, left and right arrows, some interesting graphics symbols and a POUND sign.

The hardware as it stands requires a software driver which is supplied. (See later for hardware driver).
The fitting of the kit requires the removal of the keyboard. The instructions clearly take you through the stages of removing the character generator chip; inserting the new generator chip; replacing the video memory IC; making three wire links and cutting one track. The next instruction tells you to have a cup of tea or coffee before checking everything and trying it out.

The second important addition was the ROM. This was in fact fitted at the same time as the l/c kit.
The ROM replaces the first of the three Genie ROMS and is easily fitted once the keyboard is removed. Taking the precautions given in the instructions to avoid static the original ROM is unplugged and the ROM is inserted in the same socket.

The ROM adds several useful features which are transparent to the user. Particularly useful is the option to have your own message appears on power up instead of just READY?
There is a 1/c driver, an option to have normal typing in UPPER or lower case; a keyboard debounce routine; and there is a printer re-route (for Genies using TRS-80 expansion interface). If you have a TRS-80 then I am told that the improved tape loading offered by the ROM is very good. A further feature is a routine to initialise Feature ROMS.

Another of the additions was Feature Rom A.
This is an EPROM which can be plugged onto the bus at the rear of the Genie System. The more daring can fit it internally as a more permanent fixture.

There are actually two Feature ROMs available, A and B, but I can only speak for A.
It is necessary to buy a made up PCB which plugs into the bus at the rear of the Genie, the Feature ROMs are then plugged into this PCB (One at a time).
The Feature ROMs can be initialized in either of two ways. One way is to use a short software routine, the other is to fit The ROM. The latter method is by far the best since initialization is automatic and transparent.
Once fitted and initialized the Feature Rom is called by pressing shift and 3> together, this puts a menu on the screen.

The options for Feature Rom A are:-

To abort the function.
Shift lock. This toggles between UPPER and lower case and back.
Auto repeat. This gives a repeat on all keys after a short delay.
Single key commands. Press shift plus a letter.
Cursor flash. This can be toggled on or off.
Machine code Monitor and Editor. This allows the examination and alterations of any location. (Like T-BUG). There is J for jump command and B to return to BASIC
6 System Load and Save. This option has a further menu.
L = Load 500 Baud M/C Program. No names needed and after loading, the NAME, START, ENTRY and END addresses are given.
S = Save a M/C Program. Just copy NAME, START, ENTRY and END addresses from screen and the work is done for you.
R = Run the resident M/C Program.
B = To exit function.

Option 6 is useful for making M/C backups and for checking to see if you have any resident M/C Utilities.

I have used a Genie I which has similar built in functions but find Rom A easier to use. A further bonus is that the plug in PCB can be used for other EPROMS e.g. Lowes HIRES Driver.

One of the really beneficial changes was to upgrade the onboard RAM to 48K. This was probably the most difficult but only because the Genie finished up in more pieces during the process.

The keyboard and CPU board have to be removed before cutting two links. The CPU board is then replaced before removing the existing eight RAM chips.
The replacement 48K is then inserted in the original RAM sockets followed by the making of several soldered connections. Now for another cup of coffee! PRINT MEM and up comes 48380

It is fair to say that not everything I fitted worked first time but a cal to GNOMIC on (0783) 860314 at almost any time brought uncomplaining assistance.
Colin and Tom at GNOMIC demonstrate what the words BACKUP and PUBLIC RELATIONS really mean.

All the hardware described has been designed to fit all Genies and in most cases TRS-80's. A phone call or s.a.e to GNOMIC will soon answer your questions about compatibility with your particular version of the machines.

Prices:- VAT to be added after cost and p and p.
p and p is 0.80p on small items like those described.
The ROM = £14. 95
Rom A or Rom B alone = £18.95
Rom A or Rom B with plug in PCB = £29.95 for Genie.
£34.50 for TRS-80
Lower case kit = £19.00. 48k Expansion = £43.50

If you would rather not fit your own upgrades there is a charge for fitting and testing by GNOMIC.

All the additions, are running on my Genie System with smalLDOS and I find that all is satisfactory.

Apparently the Feature ROMs are not compatible with the Aculab Floppy Tape.

Lynn Smith

In Mav's Workshop

RADIO SHACK EXPANSION INTERFACES - Part 1 of 2

by Ian Mavric

History

Even back in 1977 if you were designing a computer that you wanted to be taken seriously, it needed to be able to utilise that wonder-device of the time, the floppy disk drive. Micro computers prior to 1977 like the Apple I and KIM-1 were not designed with disk operation in mind, but their successors, the Apple][and CBM Pet were. Steve Leininger, the main designer of the Model 1, decided correctly that for the computer to competitive, a pathway to add disk drives to the TRS-80 was needed, and so an bus was included on the Model 1. This bus would interface to a unit which would expand the system. Not wanting decisions about the name to delay the device, they named it what we call it today, the Expansion Interface, or EI.

While quite a considerable time was spent designing and debugging the TRS-80 Model 1 itself, when it was released and the interest, pre-order and sales far exceeded anyone's expectations, talk of how to expand the system was a common topic of the time, and Steve was given a much shorter timeframe to design the EI. Microsoft had completed its Level II Basic language by the end of 1977, a language which was necessary because of it's floppy boot routine, which made getting the EI on the market even more urgent. In the early months of 1978 Tandy released Level II Basic, the EI, the "Mini Disk" disk drive, and TRSDOS 2.1, and that's when the fun started.

Underwhelming operation and unreliability during the first year, coupled with the bugs in TRSDOS 2.1 meant the first six or so months of Expansion Interface production and usage gave the EI a bad reputation. To Radio Shack's credit, they didn't stop trying to make it reliable, and it did take some time to properly understand why the EI gave such varying results from installation to installation. DB Kitsz writes a very good explanation of the problems with the EI so rather than re-invent the wheel, take a moment to go read "How the System Expands" in The Custom TRS-80, Chapter 5, from pages 121 to 124. Basically the beta-testers of the device were simply not prepared for the extreme environmental conditions TRS-80 systems would eventually find themselves in, and while it worked great in the test lab, it was less than reliable in the field.

Fixes were implemented in the form of the Buffer Box and then the DIN cable modification, but these only partly made up for the inherent design problems in the EI. In January 1980 Radio Shack finally released a re-designed EI and shortly afterwards TRSDOS 2.3, and the problems with the EI finally went mostly away.

Perspective

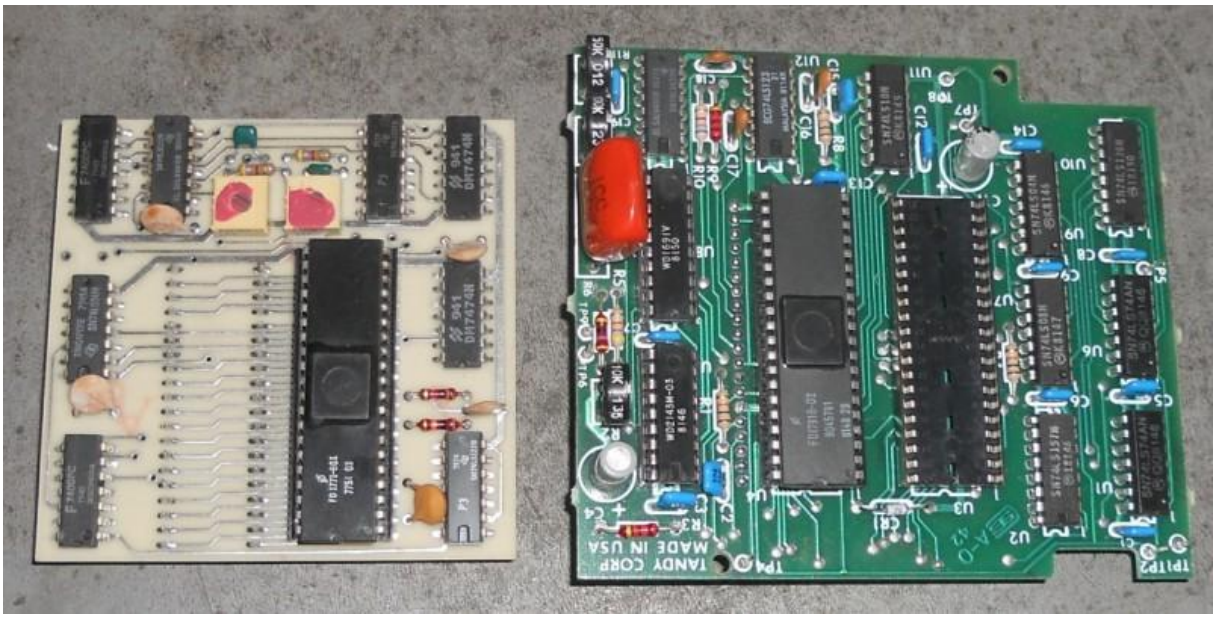
Depending on where you read to come up to speed with TRS-80s you can either think that all EIs are a chunk-o-junk that were never reliable and only the last re-designed one worked satisfactorily, but really a product from a competitor, like the LNW System Expansion was required to bring true reliability to the system. I beg to differ. Once you understand the problems and solutions you can understand how each design can be made to work reliably.

RAM, for example, has a big bearing on the performance of the early EI. Those ceramic and gold 4116 chips from early 1978 may look impressive but they are s-l-o-w (350-450ns) but they were the best available at the time. They are too slow to reliably keep up with the high-speed signals going to and from the Model I keyboard unit. It may surprise you that using faster 200ns 4116 chips from the early 80s in the EI makes it much more reliable. Co-incidence that by the time the redesigned EI came long this was the Ram which came with it?

Oxidising Edge Connectors. In a typical Tandy economy move which carried all the way through the Model 4, Tandy 2000, and Coco line, rather than sensibly gold plate the edge connectors for reliability, they solder coated them. Solder, in contact with gold or silver in the cable connectors, oxidises over time making the signals weak and prone to misinterpretation. This affects both designs of EI if left unchecked. (It also affects LNW EIs where you could buy a cheap kit version without gold connectors if you were tight.) Clean the connectors with a pink-pearl eraser every 6 months and even the earliest EIs can work well. It's often a lottery... there are those out there who never clean the connectors and the system works well, and those who find they need to clean them every 6 weeks. This randomness has to be considered a quirk of your system and preventatively attended to accordingly.

The two points listed above should set of light bulbs in people's heads that they are related, and that if you could somehow take the memory out of the EI then most of its problems go away. Installing 48K (using either 24 4116 chips or 8 4164 chips) into the Model I keyboard and having no RAM in the EI eliminates nearly all problems with memory crashes and spontaneous reboots.

Data Separation and early TRSDOS bugs. Those who have only ever used TRSDOS 2.3 know it as a boringly stable DOS with limited functionality. It does nothing perfectly but everything OK. Versions prior to 2.3 (ie. 2.1 and 2.2) which were in wide circulation at the time the EI problems were at their peak had many bugs. These bugs would sometimes cause the same spontaneous reboots that looked remarkably like the memory problems mentioned above.
Recognising this problem Apparat wrote a totally new operating system called NEWDOS 2.1 (forerunner to Newdos/80 2.x) and people were amazed at its stability even on habitually problematic systems. Tandy quietly sneaked TRSDOS 2.3 into the marketplace and as people moved to it reboots due to poor programming started to abate.
Data separation is a function of the FDC (floppy disk controller) IC and in the EI the WD1771 uses its own built-in lacklustre data separation routine. Western Digital, the maker of the 1771, even warns in it's datasheet that it shouldn't be trusted and to do the job properly, an external hardware data separator should be included in the design. Radio Shack ignores this because it worked OK in the lab. Out in the field it works most of the time too.... unless you've been unlucky enough to buy an EI which has crapulent data separation.
To fix this the Percom company made the "Data Separator" which plugs into the 1771 socket and fixes the problem. Later on when double density adapters became popular they all had the necessary data separation circuit installed. So the solution in this case is to install a doubler even if you don't use the double density function just having it there can improve single density read/writes. Finally, another problem from the time is that diskettes from the 1970s from some of the cheaper manufacturers were just not that good. Couple that with all the problems above and you are starting to get a clearer understanding of the early EIs bad reputation.



Picture: Data Separator (left) and Double Density adapter (right)

(Anyone still using diskettes on a Model I should stick to reputable brands like Verbatim, Kodak, Fujifilm, Xidex, and Gestetner 360K DSDD disks from the mid-80s will get good results. I even avoid Radio Shack brand disks from the 1980s as over their lifetime I have found they don't hold up as well as the brands listed above.)

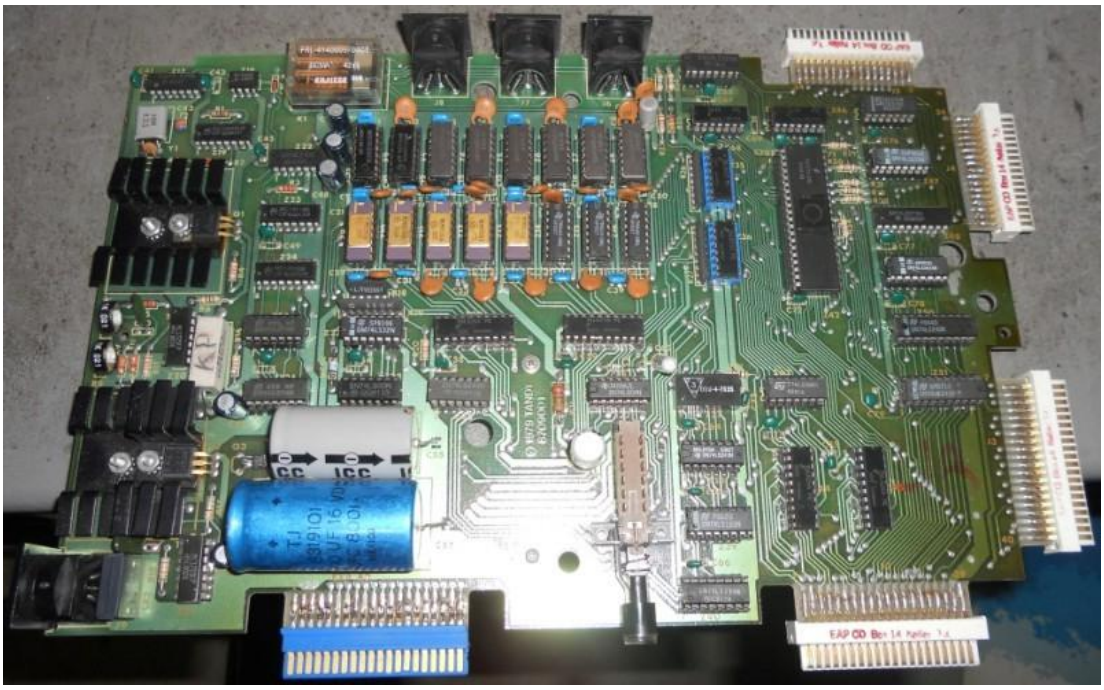
Examining the Re-designed Expansion Interface (1980-82)

NOTE: There are two basic designs of the EI, understanding them can be confusing and nothing should be taken for granted. If you don't understand something, ask on relevant internet forums or ask TRS-80 experts, as you can run into considerable trouble and not even know it.

We call them early and late EIs to differentiate between them, but the fact that the PCB can be interchanged in the case of either EI means you may have one you *think* is an early and is a late, or vice-versa, so serial and catalog numbers are handy to know but can be misleading. Far better to look at the PCB itself.

This talk concerns the re-designed EI, of which they are the most plentiful. This is due to reliability concerns starting to abate and reduced prices as the Ram and FDC (the most expensive parts in the EI) cost RS less so they could sell the EI for less, hence more people took up the option to buy it.

Although they contain similar number of chips (early: 51, re-designed: 50) the re-designed EI has a revised layout so solve a problem "microphonics" which was apparent in the early EI.
Included is four 74LS244 buffers which do the same job as the Buffer Box on the early EI, which is why the Buffer Box is not needed and won't work on the re-designed EI.
Additionally, a DDU - Digital Delay Unit - in U37 takes the single RAS line and generates the CAS and MUX signals inside the EI. It works because instead of being dependent on 3 signals accurately travelling between the Model I keyboard and the EI, only one (RAS) has to.



Picture: Re-designed EI shows parts layout. This one has Goldplug-80s (good) and mis-matched RAM (bad)

Making re-designed EIs More Reliable

1. Clean edge connectors really help, but also the cable which runs between the Model I keyboard and the EI needs to be in top condition. Since the original cables are 35 years old, if they have been treated with care they can continue to run reliably but if it's suspect (like works well on some angles and not at all on others) it's best to replace it with a new one. I sell them on my web site. Also don't forget to clean the edge connector on the Model I itself, a process which required disassembly of the M1 case but it's important you clean it's edge connector also.

2. Make sure memory is up to speed. Most of these EIs came with decent quality fast (250-300ns) Ram but examine it closely anyway and replace it with memory which has a good quality reputation. I like NEC D416C-2 which are rated at 200ns. Motorola MCM4116BP20 are also good devices rated at 200ns. At least make sure all the Ram chips in both banks are rated at the same speed. Buffering of this memory is important so make sure Z29, Z31, and Z44 and Z45 pairs (all 74LS244) are all working and if they are suspect, best to replace them with new ones.
3. Disk drive problems. The line drivers are responsible for signals stepping the disk drive heads from track to track. They are type 7416 (not LS16 or HC16) high-current devices and are know to wear out then die. Z34 and Z41 are the culprits so install good quality sockets and those chips. If a disk drive head sticks or the terminating resistor fails these two chips cop it so it's good to keep some spares on hand.

As mentioned before the data separation problem can sometimes affect these re-designed EIs as RS decided it was not a priority to add a data-separation circuit, so if even replacing the 7416s and the WD1771 with another one the disk operation is questionable at best, add a Percom data separator if you can find one, or if not add a double-density adapter.

4. Heat. In Australia the TRS-80 Power Supply units which came with our Model Is and EIs are oversized and don't fit inside the EI. I remember one summer it was 42C in my room and even with a fan on me and my Model I the system was crashing even though the PSUs were outside the EI. Similar conditions in the US which have two heat-generating PSUs installed inside the EI I would surmise cause problems at much lower ambient temperatures. Even DB Kitsz in The Custom TRS-80 mentions in hot climates to get the PSUs out of the EI and discard the plastic covers which, although they make the system look neat, trap heat in the EI that eventually cases instability problems. If you prefer the neatness of having the PSUs inside the EI an alternative is to use the Toroid PSUs I make and sell, they generate much less heat and I have had one living inside my EI for three years now and have not had any heat-related system failures.

5. Treat your system with respect. As one of the earliest disk systems the Model I has a fragile nature about it which must be respected. Don't move your keyboard around while the system is running. It's not designed that way and the movement and strain on the interconnect cable can cause a crash in even the most otherwise reliable system. Likewise don't go turning any peripherals on or off with the system running, it was most likely crash. Want to print something but didn't turn the printer on? Sorry bud but you need to save your work, power down the system, power up your printer, and the system again, reboot, re-load then print. If you are feeling adventurous try turning the printer on or off while the machine is running, 1 out of 3 times it will crash. The same goes for the Model III and 4 to a lesser extent but TRS-80s are just not similar enough to modern PCs as to be directly comparable in operating procedures.

Conclusion for Part 1 - the Re-Designed Expansion Interface

How do I know I'm buying a re-designed EI? It can be hard to tell from photos on eBay or Craigslist, but a dead giveaway is if it comes with a Buffer Box and has a sticker on it which says "Use Only With Buffered Cable". If the plastic cover from the middle front of the EI is missing, look for a large capacitor (usually it's blue), that indicates a re-designed EI. If you can see inside it via the grilles underneath the EI look for Ram chips running left to right. Early EIs have them running back to front.

NEXT TIME: I discuss the early EI and the reliability modifications RS applied

Ian Mavric - ianm@trs-80.com

E.T. FONEHUME

HI DUSTY

THE OTHER DAY I NOTICED A REQUEST FOR HELP ON THE TRS-80 FORUM, IN FINDING AN OLD BASIC PROGRAM WHICH SIMULATED A FLYING BIRD, USING JUST A STANDARD TRS-80.

THE INEVITABLE PROBLEM ENCOUNTERED WHEN USING BASIC, IN ANY FORM OF ANIMATION, IS USUALLY SPEED, OR RATHER THE LACK OF IT! HOWEVER, ON THIS OCCASION, I HAD TO BUILD IN A DELAYING ROUTINE. I KNOW, I COULDN'T BELIEVE IT MYSELF. WITHOUT THE DELAY, THE 'BIRD' JUST DIDN'T LOOK AS THOUGH IT WAS PUTTING IN ENOUGH EFFORT WHEN TRYING TO FLY!

THE PROGRAM IS VERY SIMPLE AND I'VE EXPLAINED EACH STEP ALONG WITH THE PROGRAM LISTING ON THE NEXT PAGE.

AS PER USUAL, I COULDN'T RESIST TURNING IT TO A ONE-LINER, AS LISTED BELOW! I HOPE IT MAKES YOU SMILE!

```
1 CLS:U$=CHR$(132):D$=CHR$(153):M$=CHR$(149):P=RND(14)*64:FORX=P
TOP+60:PRINT@X,U$:FORZ=1TO10:NEXTZ:PRINT@X,M$:FORZ=1TO15:NEXTZ:P
RINT@X,D$:FORZ=1TO10:NEXTZ:PRINT@X," ":NEXTX:GOTO1
```

```
10 CLS:U$=CHR$(132):D$=CHR$(153):M$=CHR$(149):P=RND(14)*64
20 FORX=PTOP+60
30 PRINT@X,U$:GOSUB90
40 PRINT@X,M$:GOSUB90
50 PRINT@X,D$:GOSUB90
60 NEXTX
70 RUN
80 REM DELAY ROUTINE
90 FORZ=1TO15:NEXT
100 PRINT@X," "
110 RETURN
```

THE ABOVE BASIC PROGRAM ATTEMPTS, USING MODEL 1, BLOCK GRAPHIC CHARACTERS, TO EMULATE A 'BIRD' FLYING ACROSS A STANDARD TRS-80 SCREEN. IT IS DESIGNED TO RUN CONTINUOUSLY, WITH THE 'BIRD' FLYING ACROSS ON RANDOMLY PICKED LINES, SO YOU WILL NEED TO PRESS THE ESC/BREAK KEY TO STOP THE PROGRAM.

LINE 10 SETS UP THE INITIAL VARIABLES TO BE USED. THE 3 GRAPHIC CHARACTERS (NOS. 132,153,149) ARE ASSIGNED TO THEIR RESPECTIVE VARIABLES. THERE IS AN 'UP', 'MIDDLE', AND 'DOWN' STROKE TO EMULATE 'FLYING'.
THE (PSEUDO) RANDOM STARTING LINE IS SET WITH VARIABLE 'P' BECOMING THE FIRST 'PRINT@' NUMBER FOR THAT LINE.
LINES 20 AND 60 DELINEATE THE MAIN PROGRAM LOOP.
LINES 30,40 & 50 PERFORM THE PRINTING OF THE 'BIRD' AND USE THE GOSUB ROUTINE PURELY AS A DELAY. (LINES 80 TO 110)
LINE 70 RUNS THE WHOLE PROGRAM AGAIN.

EXATRON STRINGY FLOPPY T.M.

SPEED
LOW COST
RELIABILITY

Exatron is a California based corporation that has been in business since 1974. As well as the Stringy Floppy, Exatron designs, manufactures and sells state-of-the-art electro-mechanical equipment for a variety of commercial and industrial applications. Exatron is an established supplier of automatic test equipment to manufacturers, and large OEM users, of integrated circuits worldwide.

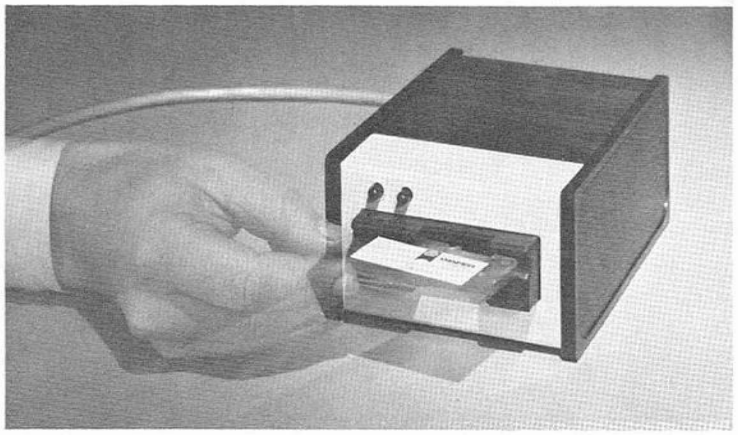
The software in every ESF adds a parity bit to every byte saved on tape, and a checksum to the end of every file. These are checked both after recording data and upon replay, any detected error is indicated by a message on the video display. This system of automatic error checking gives confidence in any data saved, also each wafer is rated for at least 2,000 complete passes past the record/replay head.

- ►Assembled and tested
- ►All operating software in ROM
- ►Fully automatic operation
- ►Professional quality
- ►No Expansion Interface required
- ►Large Owners Association
- ►High speed operation
- ►Extremely reliable
- ►No technical knowledge needed

WHAT IS IT?

The Exatron Stringy Floppy (ESF) is an extremely fast, reliable, economical alternative to cassette or floppy disk storage of computer programs or data.

Totally self-contained, the ESF has no buttons, switches, knobs or levers to adjust or forget. All of ESF's operations are under the computer's control.



TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation.

HOW DOES IT WORK?

The ESF uses a miniature tape cartridge (called a 'wafer') as the data storage medium, about the size of a business card and 3/16th of an inch thick. The tape used inside the wafer is a special Mylar based Chrome Dioxide type, specially developed for digital applications. Wafers are available in several lengths, 5 feet being the smallest and capable of holding up to 4 thousand bytes of information – the 75-foot wafer is the largest available and can hold up to 64 thousand bytes of data.

The wafers contain a single reel of the special tape connected as a continuous loop, the ends being spliced together with a piece of reflective tape. In operation the ESF drive unit pulls the tape from the center of the reel inside the wafer, causing the entire reel to rotate. Thus, the tape automatically winds itself around the outside of the reel at the same rate as which it is pulled from the center. This process is similar to that found in an 8-track cartridge.

The ESF transport mechanism is very simple, consisting of a precision die-cast aluminum block – with a capstan, drive motor and magnetic record/replay head mounted on it. The wafer loads into a slot in the casting (it will only fit the correct way) and the tape is driven at a single point by the capstan, past the record/replay head.

HOW DO YOU USE IT?

Once connected to your computer the ESF operating system needs to be activated—simple. Just type 'SYSTEM'(enter), and in response to the ? prompt type '/12345' (enter). Your TRS-80 will instantly display the ESF sign on message 'EXATRON STRINGY FLOPPY VERSION 4.1', and from this point onwards you will have the extra commands '@LOAD', '@SAVE' and '@NEW' recognized by your TRS-80.

The ESF's operating system is built into the electronics of the unit, in much the same way that BASIC is built into the computer, so it is always available – the SYSTEM command is to let your computer know that the ESF has been connected. If you normally reserve some memory for subroutines then the ESF software will relocate itself under your selected top of memory. The ESF uses only 4 bytes of your available RAM, these bytes are used to 'point' to the 2048 bytes of software in the ESF unit itself.

WHAT'S THE CATCH?

Well, the only catch that most people find is that they have to actually pay Exatron for their unit! Even this is no big deal.

Starter Kits are available with the Exatron Stringy Floppy, a supply of wafers, a bus extender and a selection of useful programs – for $299.50.

Through regular advertisements in both Kilobaud Microcomputing and 80 Microcomputing, owners are kept informed of the latest developments in wafer-based software. Plus hundreds of user 'workshops' are starting up over the country, so you can always be sure of being near to another ESF owner.

Exatron also gives a 30-day full money-back guarantee, with a 1 year parts and labor warranty on the unit.

If you have any questions about the ESF then give Exatron a call on the Hot Line (outside CA) **800-538-8559.**

East Coast customers can call **800-343-4424** (inside MA **617-899-3862**)

Open House Workshops take place from 9 am till 1 pm every Saturday at Exatron's factory in Santa Clara, and on the East Coast the last Saturday in each month at Micro Communications, 80 Bacon Street, Waltham MA 02154. All are welcome.

exatron

3555 Ryder Street
Santa Clara, CA 95051
408-737-7111

✓3

A 512KB prototype card compatible SUPERMEM

Pascal Holdry

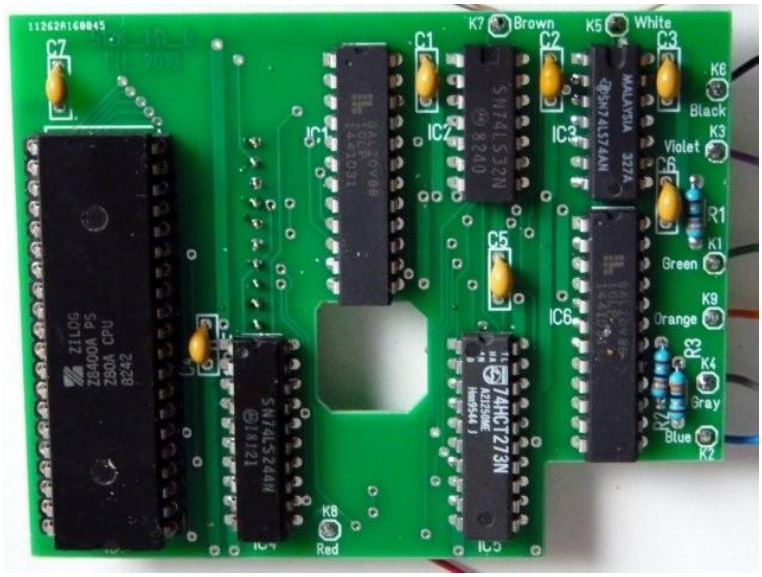
Goal:

The opportunity to work with 512KB RAM on the TRS-80 Model I computer
To retrieve existing software
Keep track of our digital heritage
Provide a diagram for troubleshooting and understanding the operation of the card.
Use only type GAL20V8 unlike Supermen card or two types of Pals are used.

The programming of the GAL was directed by Régis Munoz.

A hole in the PCB was made for the passage of a large colonette the housing bottom (unlike the SUPERMEN card that requires cutting this great column).
Attention all components except the Z80 should be mounted without brackets

Map view prototype



Card attached to the motherboard

Changes in the bottom of the case

Must be shortened by a few millimetres a short column with a forest

Demo.

Demo 01

Demo 02

The connections of the card

The wiring is identical to the wiring instructions of the SUPERMEN card due to compatibility.

but the prototype of a memory bank uninterruptible PCB is under consideration.

PART I. The 16K memory chips now in your computer use -5, +5, and +12 volts. 64K and 256K memory use only +5 volts therefore we must eliminate the -5 & 12 volts and re-route the +5 to the proper pin. Once you make this modification it will not be easy to return to 16K chips.

1. Remove all 16K memory chips from the sockets. These memory chips will not be used and you may want to save them for one of your other projects. Z13 Thru Z20

Remove the following components from the CPU board...these may be de-soldered or clipped.

C3, C4, C16, C17, C18, C19, C29, C31, C33
CR2
R19

NOTE: Skip the next paragraph if you elect the ALTERNATE METHOD.

Isolate pin 9's of all the RAM sockets by cutting the trace that goes to pin 9 for each socket. The trace for Z20 is on the component side of the board. Now that all pin 9's are isolated we need to connect them together. Connect each pin 9 together and also connect them to Z51 pin 12 on the back side (solder side) of the CPU board.

ALTERNATE METHOD

Before installing the 256K memory chips, bend pin 9 out and back over the top. Do this for each memory chip. Insert the memory chips in the sockets. Using a length of small wire, connect all pin 9's together and also connect them to pin 12 of Z51. Z51 pin 12 must be common to all memory chips installed. We recommend you use the first method so that you don't destroy and RAM chips. The pin 9's will break off if you are not careful. If you use the alternate method it will be hard to upgrade to 512k.

Isolate Z19 pin 8 by cutting the large trace that runs to C15. Cut the trace close to pin 8. Verify that all pin 8's of the RAM sockets are connected to each other but isolated from the 12 volt buss. Once you are satisfied that they are, run a jumper from pin 8 of Z19 to the trace connecting to Z36 pin 14 .

******** USE EXTREME CARE IN HANDLING THE 256K MEMORY CHIPS ********

Install 1 set (8 chips) of 256K memory in sockets Z13 thru Z20. Be sure pin 1 of the memory chips is in the upper left corner of each socket.

You are now ready to check out the memory circuits before installing the memory board. Do not connect the expansion interface at this time. Plug in the power and video cables and turn on the computer. Press <ENTER> for MEMORY SIZE?. Type PRINT MEM and you should have a 16K system. If you get MEMORY SIZE? and the computer locks up the system is still working ok. All we are checking here to to see if the computer will power and access the memory. Some 256K memory chips will not work at all without the 256 cycle refresh provided by our board. Some will work for a few minutes. If the computer won't power up or you get garbage, double check all your work. The computer must be working properly before beginning part II. Once you are sure that the computer is working you may proceed to the next step.

PART II. Remove the DIP shunt installed in Z71. Bend out pin 5 and re-insert the shunt back in the socket. Remove the Z80 (IC Z40). Install the memory board in the socket for the Z80. Press the header into the Z80 socket firmly. Insert the Z80 into the socket on the memory board (Z40). Pin 1 should be pointing in the opposite direcction as the other chips on the memory board. The wires from the memory board should be coming out towards the center of the CPU board.

You must now solder the wires to selected points in the next step. We have left the wires longer than needed but you must cut them to fit as short as possible. This is critical for the violet wire. If the color code is different on your board, go by the position of the wire, not the color. After you determine where each wire goes, we suggest you put a drop of solder at that point and then solder the wire to it. Use as little solder as possible. The more the better is not the case here.

Now solder on the following wires from the memory board to the CPU board.

Memory board pad 1 -GREEN--> Z51 pin 13.
Memory board pad 2 -BLUE---> Pin 1 of RAM sockets.
Best place to connect is at the left side of C19 or rather where C19 used to be.
Memory board pad 3 -VIOLET-> Z67 pin 13.
Memory board pad 4 -GRAY---> PIN 15 2nd set memory
Memory board pad 5 -WHITE--> Z72 pin 10.
Memory board pad 6 -BLACK--> Z72 pin 3.
Memory board pad 7 -BROWN--> Z74 pin 10.
Memory board pad 8 -RED----> Z21 pin 15.
Memory board pad 9 -ORANGE-> Z51 pin 14.

The scheme, programming LAGs and Gerbers of this map are available for free in the Downloads menu

ph@prof-80.fr

THERE'S A LINK TO PASCAL'S WEBSITE ON TRS-80.ORG.UK

Cleaning a TRS-80 model III keyboard

Robert C Baruch

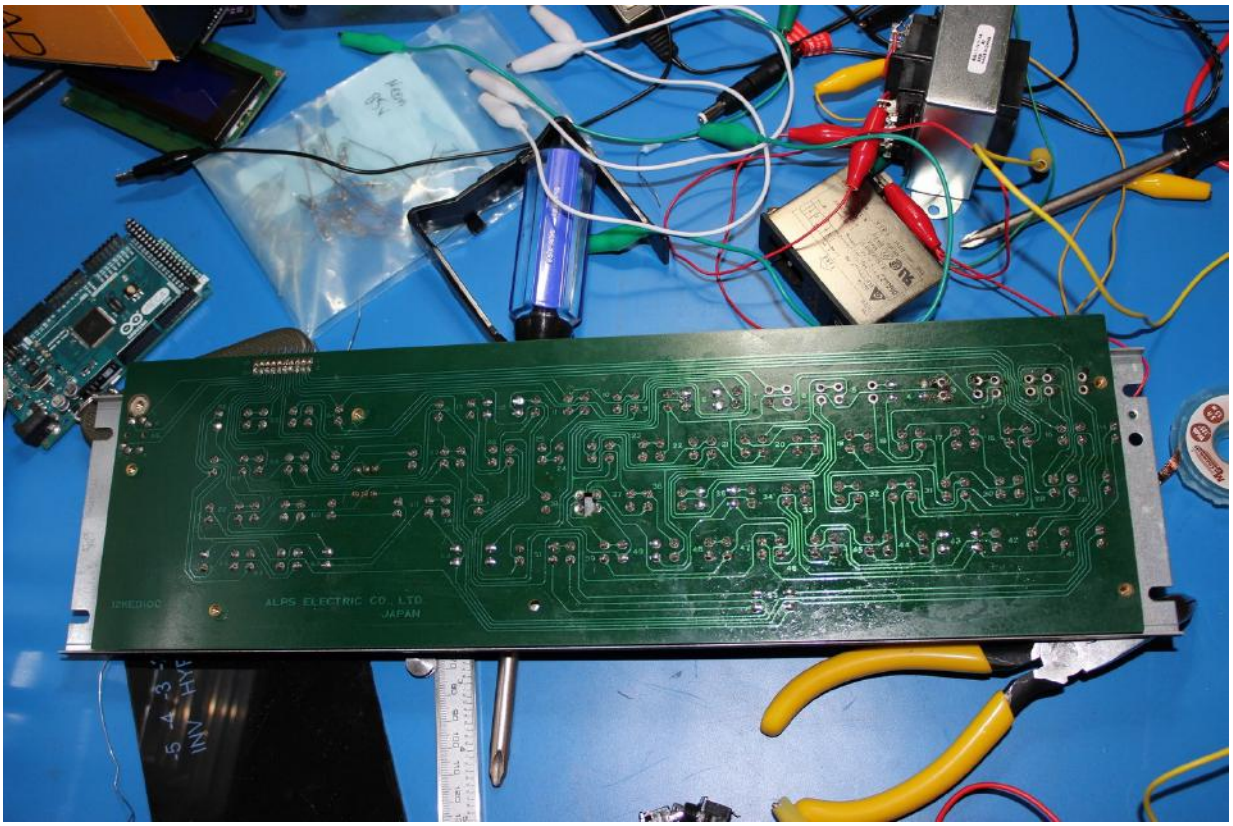
(www.halfbakedmaker.org)

I have a TRS-80 model III.

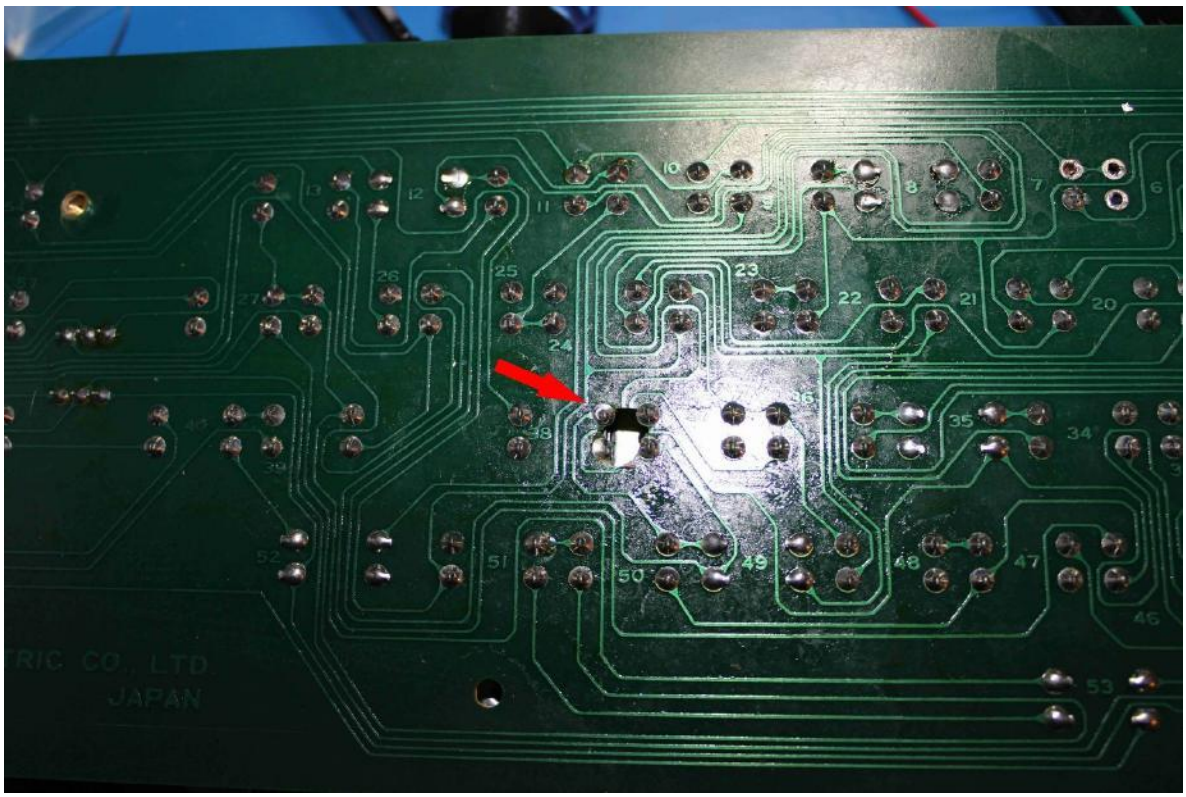
It has a keyboard. It doesn't work so well. Some keys don't work, others are flaky.

So I opened it up and popped the keycaps off.

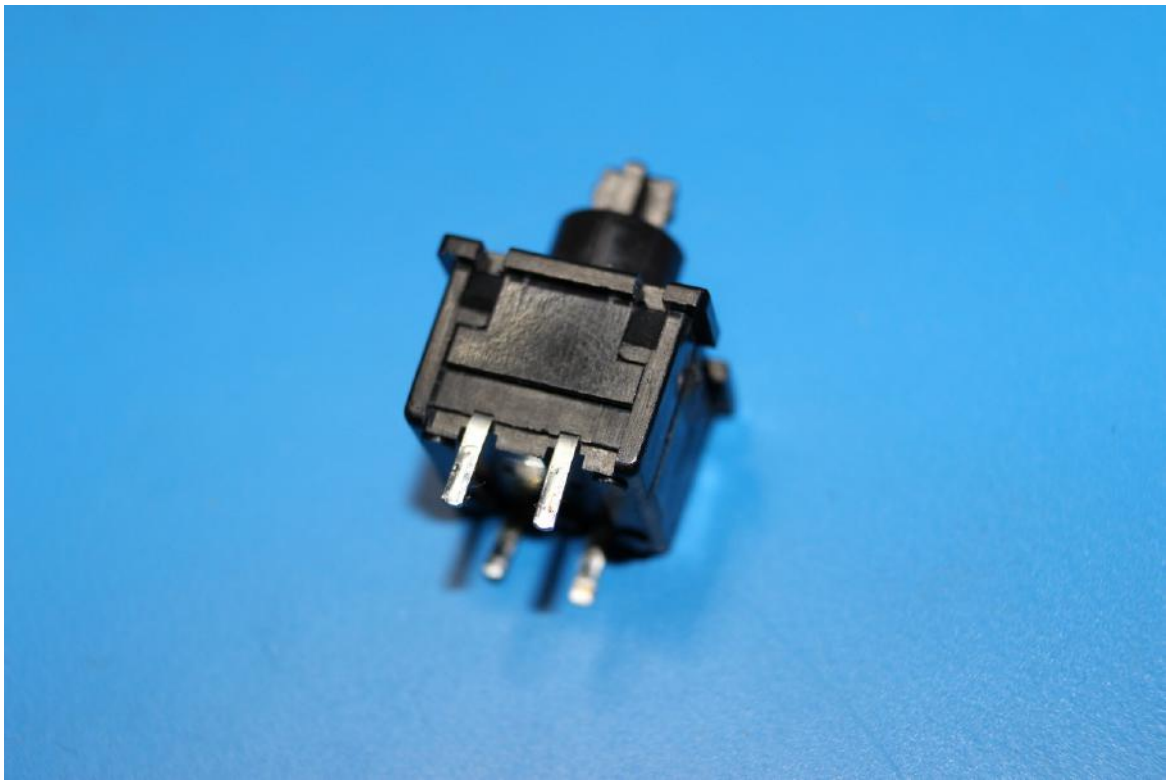
I removed the keyboard assembly. It's an Alps 12KE010C.



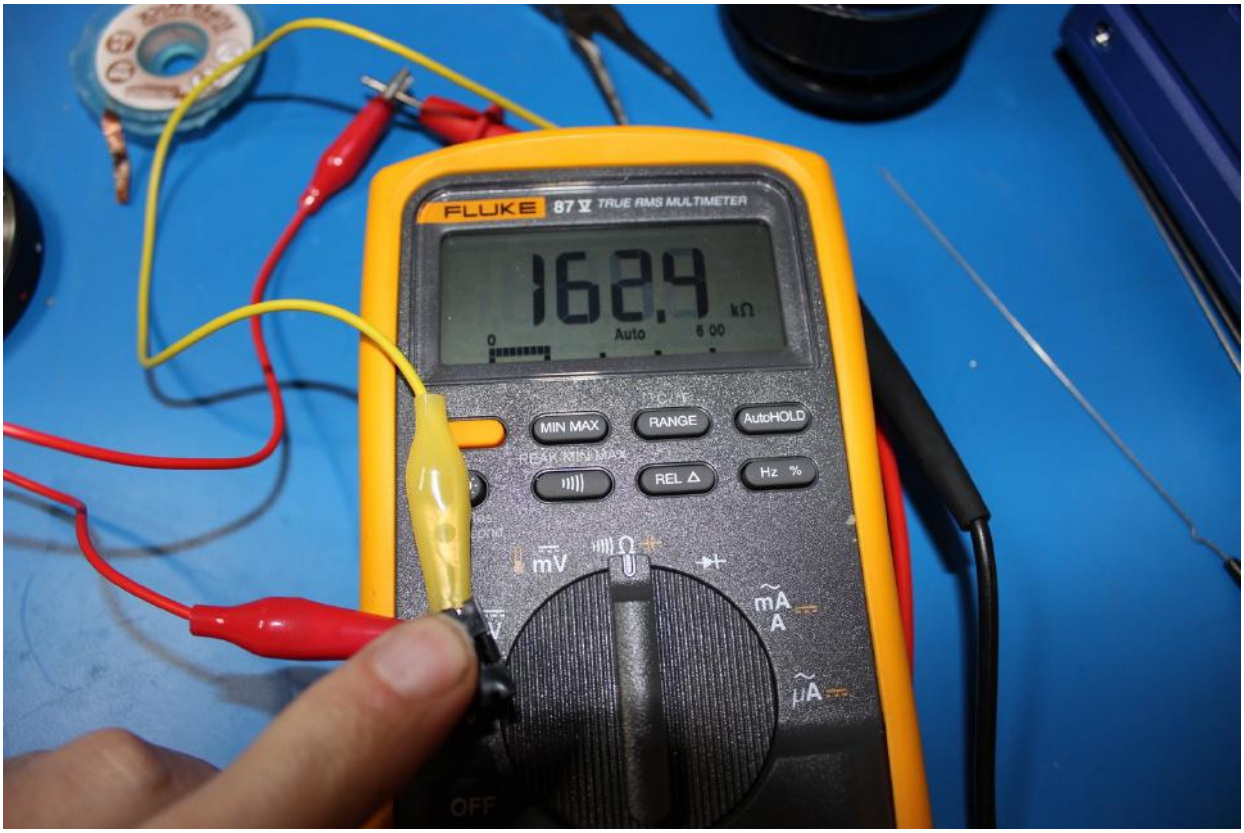
Well, that explains some of the keys not working at all. The only thing protecting the keyboard PCB from the plastic post in the base of the computer is... nothing. Oh Radio Shack. Were you already beginning the long slow slide into oblivion?



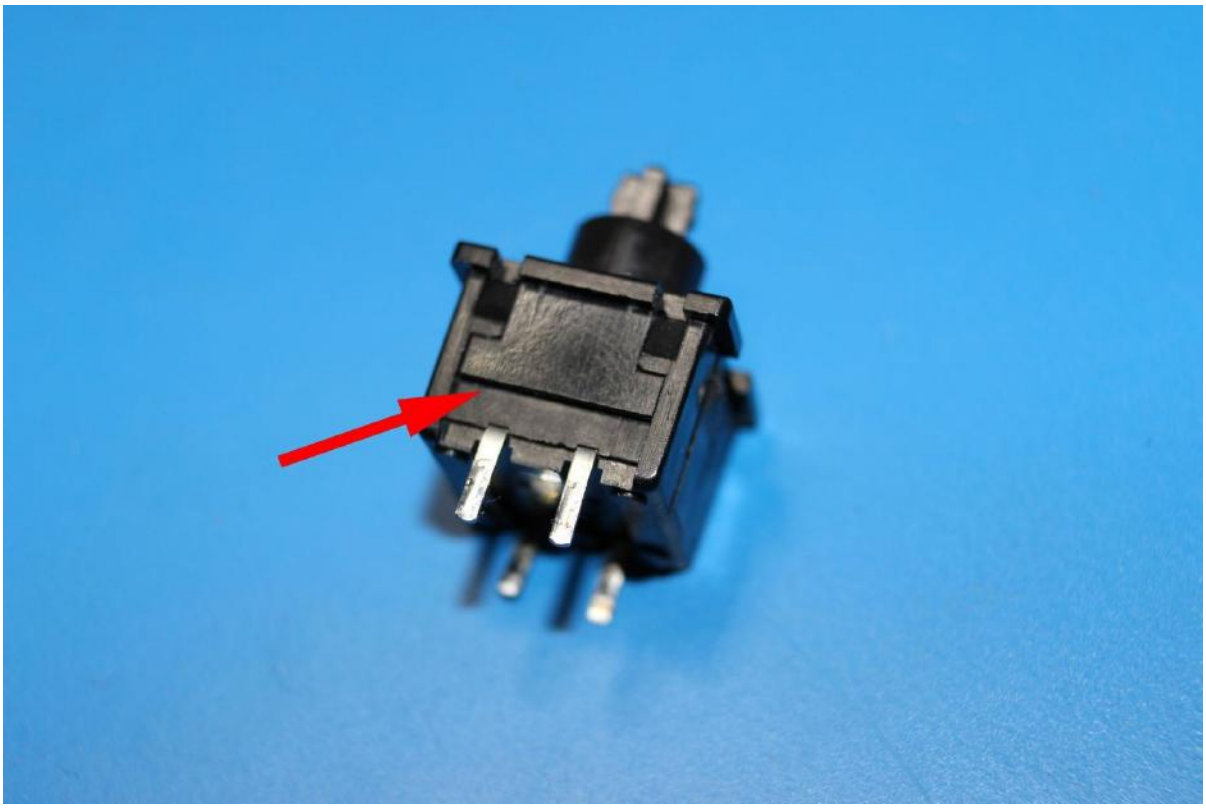
Still doesn't explain flaky keys. So I desoldered a key, since that's the only way to get them out.



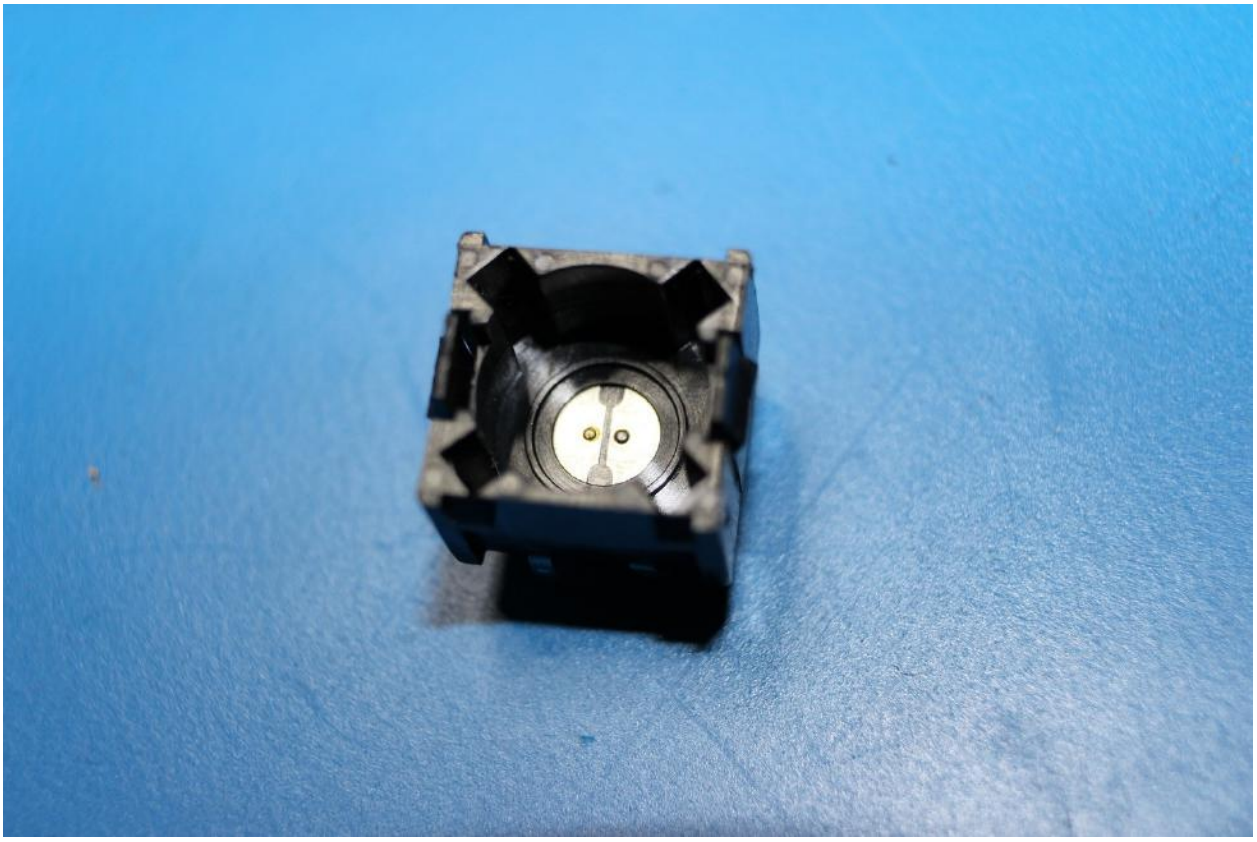
I measured the resistance when pressing the key.



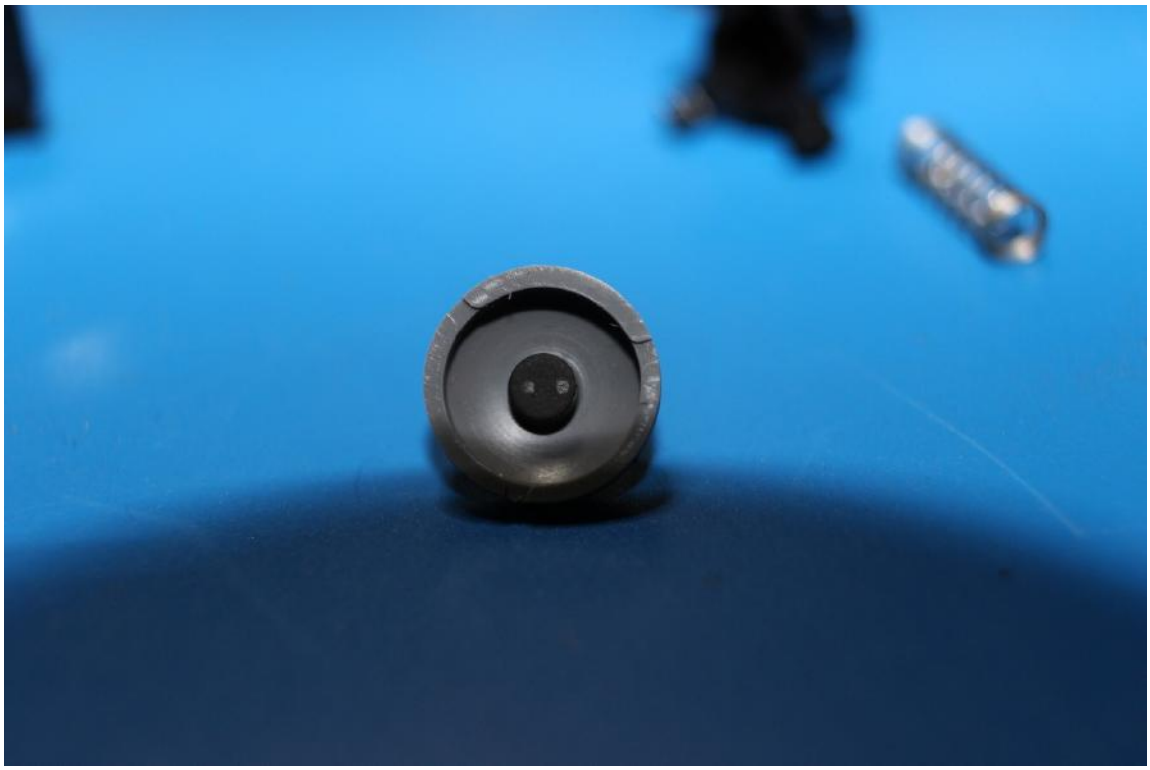
160kohms?! That's just wrong! Let's open this thing up. It's easy: you just pry on these tabs with your fingernails and they just pop apart.



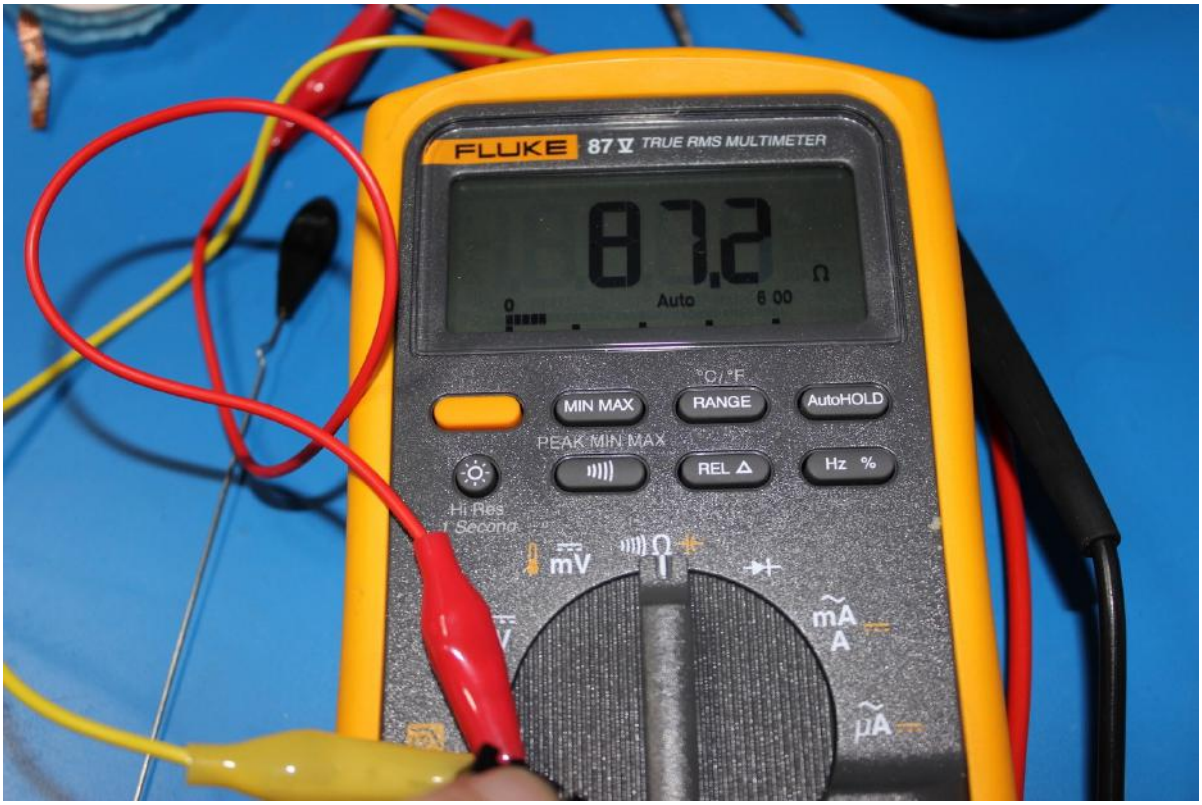
The inside of the bottom contains the contacts.



Those little dimples are supposed to press into the rubber conductive material inside the dome. And press they did. I suspect these keys weren't meant to stand up to much use. 640k key presses ought to be enough for anyone.



Lacking true contact cleaner, I just scratched up the contacts and dimples with a screw-driver to maybe get rid of oxidation. I also rotated the dome 90 degrees so the dimples can press into fresh rubber. I put the key together and...



87 ohms! That's more like it.

63 more keys to go :(

TRS-80 Emulators . com

TRS32: A Model I/III/4/4P Emulator For Windows

written by Matthew Reed

Unregistered Shareware Version:

- Works under all current versions of Windows
- Full Windows application — no low-level hardware conflicts!
- Model I, Model III, Model 4, and Model 4P emulation
- Four floppy disk drives (with optional realistic disk drive sound)
- Cassette tape drive with graphical on-screen controls
- Exatron Stringy Floppy emulation
- Printer support
- Serial port for RS-232 communications
- Joystick support (using a Windows joystick — TRISSTICK and Alpha Products joysticks are emulated)

Registered Version:

- All features included in the shareware version
- Built-in emulation of an Epson FX-80 dot matrix printer (including graphics and control codes)
- High resolution graphics (Radio Shack and Micro-Labs)
- Up to 1 megabyte of additional memory in Model 4 and 4P modes
- Hard disk support
- Orchestra 85/90 music generation

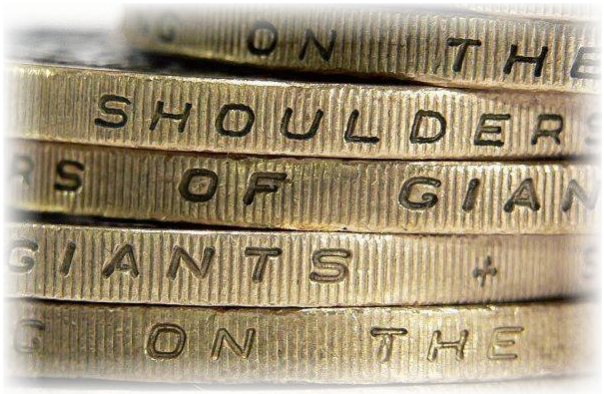
Interested?

- Read the TRS32 emulator documentation
- Download the shareware version
- Register online

Actual Dotwriter output — emulated FX-80
click for larger sample

MATTHEW'S EMULATOR IS QUITE ASTOUNDING.
IT'S MY FIRST CHOICE WHEN DEVELOPING IDEAS FOR TRS8BIT.
HE OFFERS FULL SUPPORT AND IT IS STILL UNDERGOING ENHANCEMENT.
IF YOU HAVEN'T REGISTERED YOUR SHAREWARE VERSION YET, PLEASE DO SO AND ENCOURAGE
AND SUPPORT MATTHEW'S ENTHUSIASM TO CONTINUE WITH HIS WORK.

ED.



USING NEWTON'S QUOTE "STANDING ON THE SHOULDERS OF GIANTS" AS INSPIRATION, I OFFER ANOTHER IN THE SERIES OF SOME OF THE KNOWN, AND LESSER-KNOWN NAMES WHO INSPIRED MUCH OF WHAT WE, TODAY, TAKE FOR GRANTED.

Herman Hollerith

1860
Born in Buffalo, New York State

1879
Graduates from Columbia University and becomes an assistant at the US National Census Office

1882
Enrols at Massachusetts Institute of Technology to do further research

1883
Works in Washington for the Patent Office

1884
Applies for his first patents for representing information on continuous punched paper tape

1887
His system is adopted for processing mortality statistics in Baltimore, Maryland

1889
System installed in Surgeon General's office for organising army medical statistics. Files a patent for the concept of the individual punched card

1890
Wins competition for supplying equipment to process the 1890 census. Gains a PhD from Columbia University for work on information processing

1900
Introduces a new generation of improved equipment

1901
New equipment used in the census of agriculture

1905
First patents start to expire and competition begins against his monopoly

1911
He forms a holding company, the Tabulating Recording Company

1914
The famous business administrator Thomas J Watson takes over the running of the company

1924
Company changes its name to International Business Machines (IBM)

1929
Dies in Washington DC

COURTESY OF IBM UK LTD

The inventor who put America's population figures on punched cards and went on to found the world's largest computer company

Hollerith was born in America in 1860. After graduating from Columbia University he became an assistant at the US National Census Office, helping to compile statistics from the census of 1880. The work was all done by hand and was painstakingly slow — so slow, in fact, that when the time came for the next census, 10 years later, the Office was still tabulating the results. Hollerith knew his strength lay in invention; in order to train himself as an inventor and develop his creative skills he left the National Census Office and joined the Patent Office in Washington.

Hollerith's first idea was to code information onto paper tape. The paper tape was marked with ink into 'fields'. Each field represented different categories — say male or female, or black or white. The presence of a hole in the male/female field meant that the subject was male, while its absence denoted a female, and so on. These holes could later be 'read' by machine. His first patents came out in 1884, and over the next few years he improved his system. He began by processing information on health statistics for the fast developing American cities and the army administration.

Five years later in 1889 he improved on the idea of punched paper tape by using separate cards for each individual. The cards were the size of a dollar bill — partly, it is said, because the only equipment that could be adapted had been built for handling money. The holes were originally round and made with a bus conductor's punch, but later special punches were built to cut a 6mm (¼in) square hole. In this way a great deal of information could be included on a single card.

The advantage of individual cards over continuous tape is that information can be sorted as well as totalled. For example, you might want to find out the number of white women aged 80 living in New York City. All the cards would be sorted through and any with holes punched in these three fields could be mechanically separated from the rest. These early machines could produce only a running total but later on, in order to improve the efficiency of his machines, Hollerith introduced addition and other simple arithmetic operations, proving that machines could replace the human hand.

Commercial success came in 1889 when the Bureau of Censuses announced a competition to supply a system of equipment to process the coming year's census. Systems were tested by retabulating data from the previous census. Hollerith's equipment won. His machines were by then all protected by patent and he took advantage of his monopoly by charging the Government 65 cents to process every thousand cards. Though there was an individual card for every person in America, Hollerith took only two years to finish. He announced that the population stood at 56 million, and invoiced the Government accordingly.

By the time of the 1900 census he had developed far more efficient machinery but he refused to lower his charge. When his patents ran out the Government looked to other companies but Hollerith overcame the competition by forming his own company, which later became International Business Machines. Today IBM is the world's largest computer manufacturer with an annual turnover of 20 billion dollars.

Card Games
Hollerith's original method of representing information is still in use today, a century later, though the format of the cards has changed. Modern punched cards have 12 rows of 80 columns. Tabulating machines used the decimal system, and so each card could store 80 numbers. Alphabetic characters were created by 'multi-punching' — making more than one hole in a column. Computers can also accept cards that are punched in the binary system

TRS-Box

Peter Cetinski

TRS-Box is now in beta!

TRS-Box is a Dropbox client for the TRS-80 Microcomputer. It allows you to navigate and copy files from and to your Dropbox from your TRS-80. It is an easy and powerful way to manage the files on your TRS-80.

TRS-Box requires a TRS-80 Model I with MISE or a TRS-80 Model III or 4 with M3SE setup with a broad-band Internet connection and a Dropbox account.
To use TRS-Box you will first log into your Dropbox account and give permission for TRS-Box to access it's folder in your Dropbox. Authentication takes place on the secure Dropbox site so TRS-Box never has access to your Dropbox credentials. TRS-Box will only have access to its own folder found at Apps/TRS-Box. This is where you will place your TRS-80 files.

Go to http://www.trs.pski.net/trsbox/ login and just follow the directions.

The TRS-Box Commands

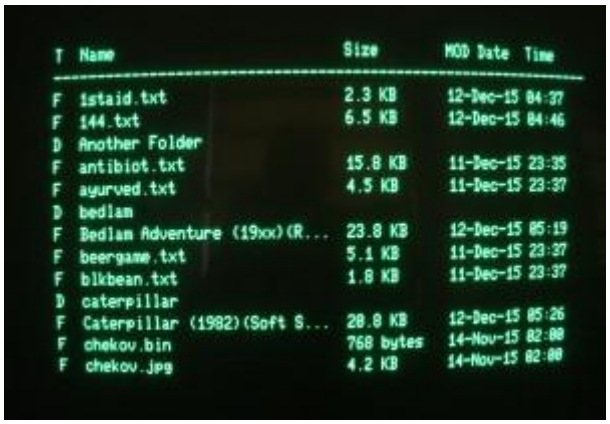
You can download the TRS-Box commands here trs-box-0.1.zip. Once you unzip them, you can copy the command programs to your TRS-80 using the MISE/M3SE FTPD service. Be sure to transfer using Binary mode.

dbsetup – Configures TRS-Box with the access token provided after logging in to Dropbox on your modern computer.
e.g. `dbsetup 1111111111`

dbpwd – Shows the current working folder in Dropbox

dbcd – Changes the current working Dropbox folder. You specify the folder to switch as a parameter. Standard directory techniques apply such as ".." to move up a folder. Use quotes when folders have a space.
`e.g. dbcd "Another Folder"`

dbdir – Shows the contents of the current working folder in Dropbox



Viewing the contents of a Dropbox folder on a TRS-80 using the TRS-Box dbdir command

dbget – Retrieves a file from Dropbox. You specify the file name of a file in the current Dropbox folder and a TRS-80 filespec for where you want the file to be copied.
e.g. `dbget readme.txt readme/txt:2`

dbput – Copies a local TRS-80 file to Dropbox. You specify the TRS-80 filespec of a file on the TRS-80 and a filename for where you want the file to be copied in the current Dropbox folder.
e.g. `dbput readme/txt:2 readme.txt`

Future: dbsync – Synchronizes the current working Dropbox folder with a TRS-80 logical drive. You provide the TRS-80 logical drive you want to sync with as a parameter. Only files in the Dropbox folder that are valid TRS-80 filespecs will be considered.
e.g. `dbsync 3`

Future: dbexec – Runs a CMD file in the current working Dropbox folder on the TRS-80. Only files in the Dropbox folder that are valid TRS-80 filespecs will be considered. You specify the filename as a parameter.
e.g. `dbexec visicalc.cmd`

Notes

This is beta software and is more than likely to occasionally crash and freeze your TRS-80 from time to time. Don't worry… simple system reset will get you going again.

If you run into problems, try getting a new token and running dbsetup again. Let me know if you continue to have issues.

During beta, please don't copy over important files without suitable backups in place.

Data from TRS-Box tools to the proxy server is not currently encrypted so please do not transfer files with sensitive data. I'm looking into implementing lightweight encryption in a future release.

Works with Dropbox

I am a software engineer of modern systems who played with Atari's and Commodores as a boy and have a new found interest in retro computing and software development on the classic Tandy machines.

THIS ARTICLE FIRST APPEARED IN THE FEBRUARY 2016 EDITION OF SYDTRUG NEWS.

Back from the dead - The resurrection of a System 80

by Garry Howarth (Gazza)
(ex SYDTRUG member)

Recently I repaired my System 80 and also revitalised a piece of firmware I had designed and built. This was ROMDOS/80, a hacked (improved!) version of NEWDOS/80 in an EPROM.

Having done a restoration of an old MG many years ago, I know what to expect when working on vintage equipment. Frustration, a feeling of hopelessness interlaced with tears, tantrums and much consumption of alcohol. Still it's been an interesting exercise, getting this old computer up and running after it sat in the junk pile for over 15 years. Dead keyboard, corroded and broken connectors and sockets, A PSU that went up in a cloud of smoke, dead Percom, dead insects and the list goes on.

The keyboard responded to a bit (lots) of contact cleaner, the same with the various sockets and connectors. The broken ribbon cables were repaired and the old beast given a general clean-up. Then came the big moment, switched her on and was greeted with something I haven't seen in over 15 years – Ready

Great, she's a goer, the image is a bit shaky but she works. A big pat on the back for a job well done.

But what is that smell and why is the room filling up with smoke?

sigh!

Replaced the smoking ruins of what used to be the power supply with one I found in the junk pile and tried again. A quick check with the CRO and all looks good. A bit of noise but zero ripple and a rock steady image on the monitor.

The expansion unit was given the same treatment as the computer and a quick once over with the CRO.

Connected the drives and tried again.

The motors start and the drive lights come on but no head activity. I removed the cover from the drives to find one of the edge connectors half off.

Both the 1771 and 1791 controllers are toast.
sigh - is this ever going to end?

Back into the junk pile and I scored a spare 1771.

Try again - and a sound I haven't heard in over 15 years - the clunk, clunk,clunk of the heads moving.

A 1791 controller was found on ebay and we now have a fully functioning Percom Doubler.
All 4 drives were given a complete overhaul and are now running perfectly

After a lot of hard work it's finished and working as good as new - or close to it.

I'll tell you about ROMDOS/80 next month

The complete setup
(Full size photo in our on-line Photo Gallery

My System 80 (Blue Label - number pad replacing cassette) with it's accessories:-

128k of memory – bank switched, like a 4P

CPU turbo running at a fantastic 3.6Mhz

2 * 80 trk DSDD Teac 5 ¼" drives

1 * 40 trk DSDD Tandon 5 ¼" drive

1 * 40 trk DSDD MPI 5 ¼" drive

A switch to make any drive become drive '0'

A printer buffer built from a kit designed by Don McKenzie from Melbourne.

A Votrax SC-01A Speech Synthesizer

An old Epson LX-850 9 pin dot-matrix printer

Joystick port, amplifier and speaker

Nokia 1200/2400 Modem and a fire extinguisher (in case of more smoke).

TRSWiki

TRSWiki now runs on the Model III and Model 4!

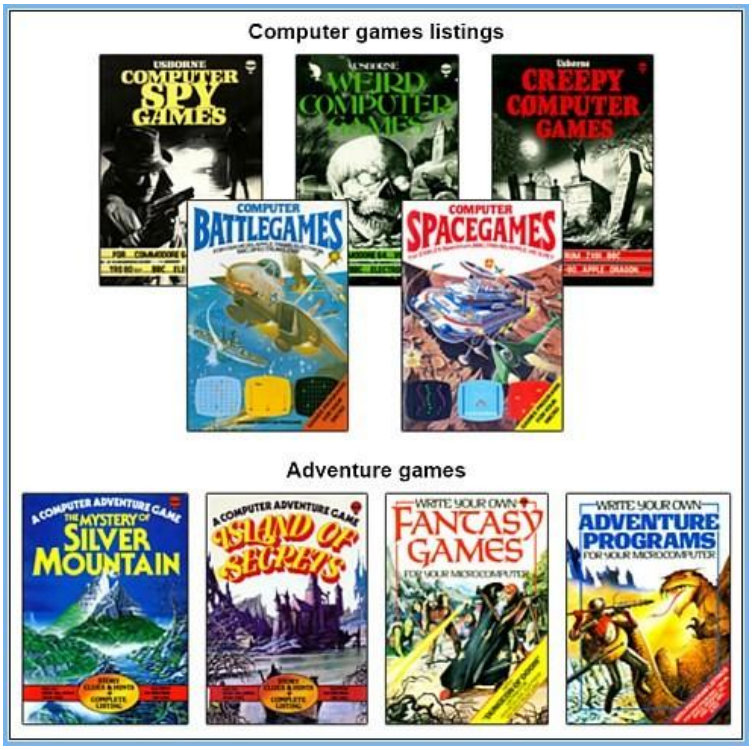
The Joker as rendered by TRSWiki on a Model 4 with the M3SE and Hi-Res graphics card

Mad Max movie poster as rendered by TRSWiki on a Model 4 with the M3SE and Hi-Res graphics card

Portrait of the Lost in Space family as rendered by TRSWiki on a Model 4 with the M3SE and Hi-Res graphics card.

See TRSWiki in action

TRSWiki is a Wikipedia client for the TRS-80 Model I, Model III and Model 4 computers. TRSWiki currently requires the MISE – Model I System Expander or M3SE configured with an Internet connection.



NOW AVAILABLE FOR DOWNLOADING AT

http://www.usborne.com/catalogue/feature-page/computer-and-coding-books.aspx#copyright

IN MAV'S WORKSHOP:

CP/M on the Model I

- A discussion on Mappa-1

by Ian Mavric

Development on CP/M adapters for the TRS-80 was effectively killed on 1st August, 1983 when Tandy released the Model 4, which had all the hardware necessary to effectively run CP/M.
Montezuma was quickly in with their version of CP/M 2.2 and Tandy followed around 6 months later with its version of CP/M 3.0 Plus. Before the Model 4 there was a thriving industry in making CP/M work on the Model I and III, and Omikron was at the cutting edge of it.

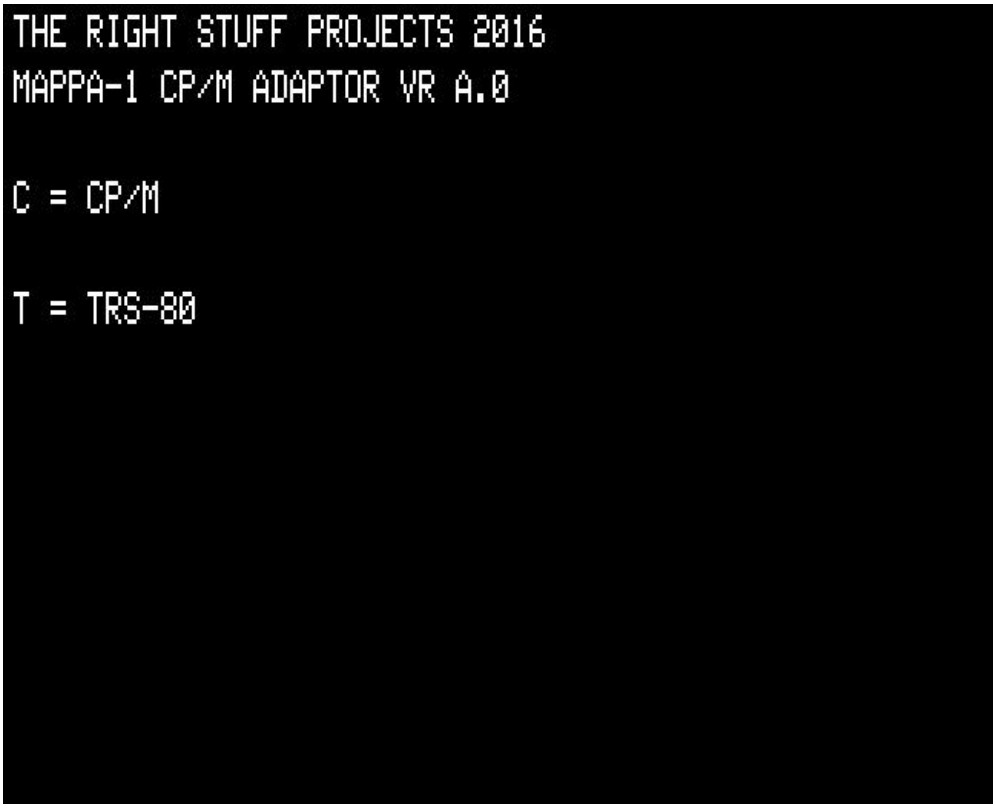
Why CP/M?

Control Program for Microcomputers, or CP/M, was developed by Gary Kildall of Digital Research around 1973 as an operating system for the 16K Intel Intellec-8 development system with a Shugart disk drive.
It was based on the Intel 8080 processor and so was a natural for the Z80 systems which would come along in the late 1970s. CP/M's requirements are very basic: an ASCII terminal, 16K memory (or more - preferably 64K), an 8080, 8085 or Z80 CPU, and a disk drive an a bootstrap routine.
With such vague requirements, virtually all systems based around the Altair's S-100 bus ran CP/M. Anyone designing a Z80 system in the late 1970s would have had CP/M in the back of their minds as an operating system, and this might have been the case at Tandy as well, since the original design of the Expansion Interface spec'd that it would be S-100 compatible.
It ended up not being S-100 compatible and around this time Tandy chose to develop their own DOS in-house, which became TRSDOS 2.3

Compared to later operating systems like LS-DOS, Newdos/80 and MultiDOS, CP/M is a quaint little operating system with an enormous software base. When it comes to functionality it's a plain, reliable, low memory-overhead operating system .

MAPPA-1 Brings CP/M Easily To The Model I

Based on the old Omikron design, Mappa-1 installs easily into your Model I and is completely reversible as no trace cuts or soldering is required. To install it you simply open your Model I, prize out the Z80 CPU and install it in the Mappa-1, and then insert the Mappa-1 into the motherboard Z80 socket. Once installed, Mappa-1's Eprom boots the system to a choice of either TRS-80 or CP/M mode:

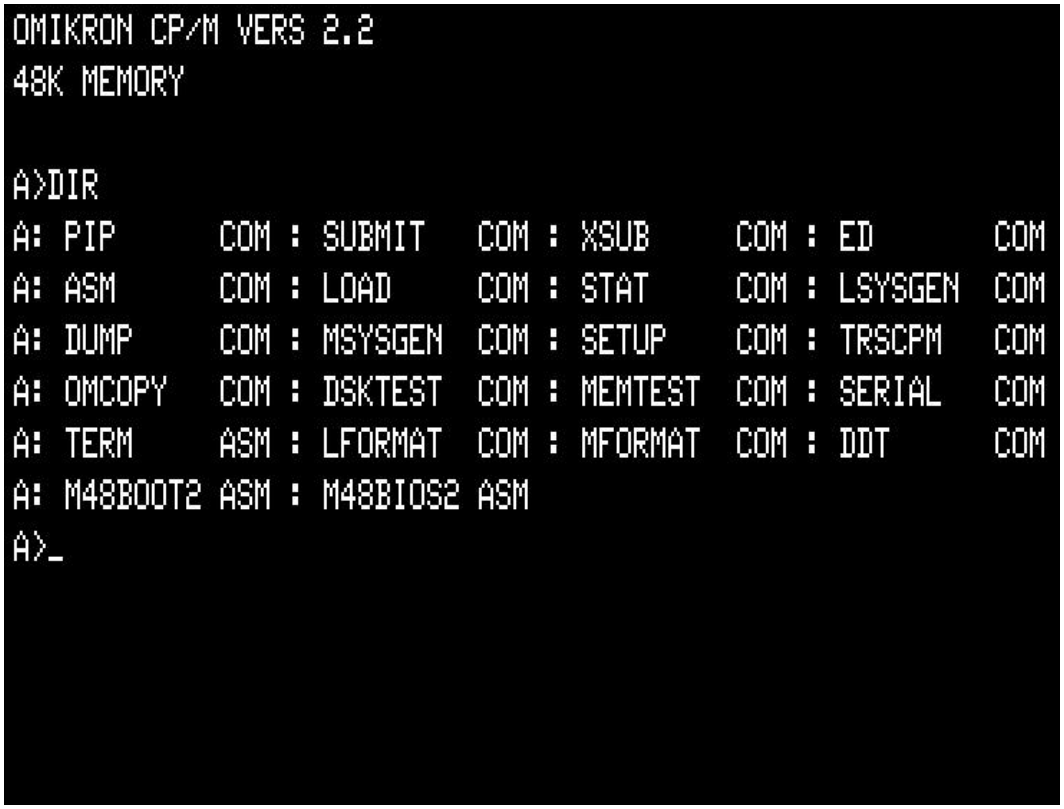


In CP/M mode the memory map is re-arranged, or "re-mapped" (hence the name) which starts main memory at 0000H to C000H. All the memory mapped ports of the Model I are addressed at new addresses now. This includes the keyboard, video display, cassette port and floppy drive ports. The new addresses are calculated by adding C000H to the old addresses. So the video memory is now found at FC00H to FFFFH).

OK Let's Boot CP/M Already!

Booting and running CP/M couldn't be simpler, just have your boot floppy disk ready inserted in Drive 0 and answer C to the operating system request above. Pressing C remaps the memory, loads the CP/M BIOS and goes to load in and execute the contents of sector one of track zero on the boot floppy disk. Before you know it you are greeted with:

Doing a DIR gives us a directory of the disk in the usual manner, and I'll talk about a few of the files on the disk:



CP/M 2.2 comprises of 6 commands and 5 utilities. The utilities you can see above are standard CP/M offerings: PIP, which copies files, STAT, which displays information about files and free disk space, DDT, a Debug monitor, ED, a text editor, ASM, which is an 8080 assembler.

Specialist utilities for the Omikron implementation on the Model I: MFORMAT, formats a 5 1/4in disk, LFORMAT, formats an 8in disk, OMCOPY backs up the whole disk, MSYSGEN/LSYSGEN copies CP/M image to 5 1/4in or 8in disk respectively, DUMP, displays a file in hex, LOAD, converts a Hex file to Com, and DSKTEST and MEMTEST are diagnostics for the system.

What Will Run on Mappa-1 CP/M?

The heyday of the CP/M mapper for the Model I was 1980-82. If you Google around you can find lots of repositories of old CP/M software. http://www.classiccmp.org/cpmarchives/ and http://www.retroarchive.org/cpm/index.html are two which spring to mind. You will find lots of software which doesn't work or only partially work - mainly due to the 64 column screen or 48K limitation - since the standard CP/M 2.2 system of the early 80s has an 80 column screen and 64K memory. But you will also find a fair amount of software which is interesting and does work.

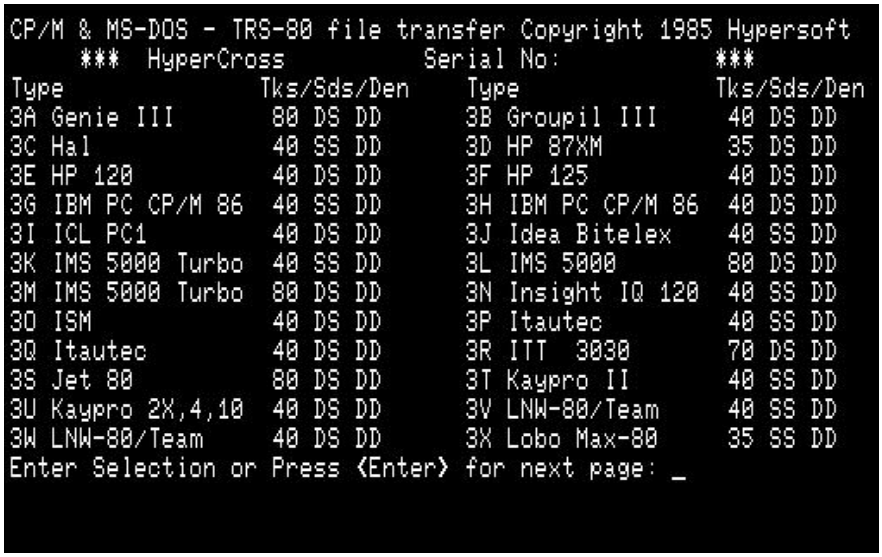
I look through archives and try to find software, and when I do I upload it to the Mappa-1 web page so people can download it and use it on their systems. My time however is limited, what with everything else I do for the TRS-80 community, so you can help me out and as you find interesting software which works, email me the details so I can make it available to others for their usage.

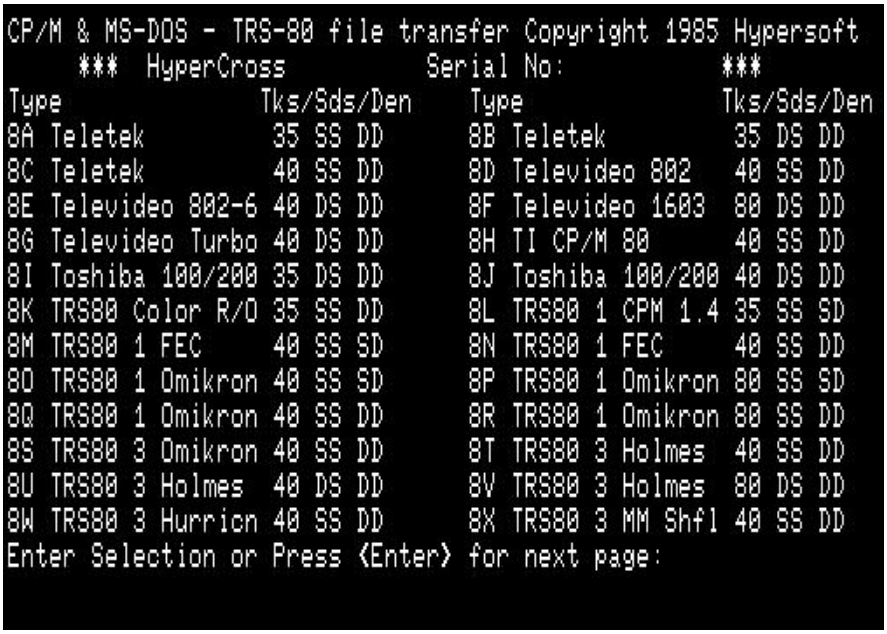
HyperCross - handy utility to move files from and to Omikron CP/M disks

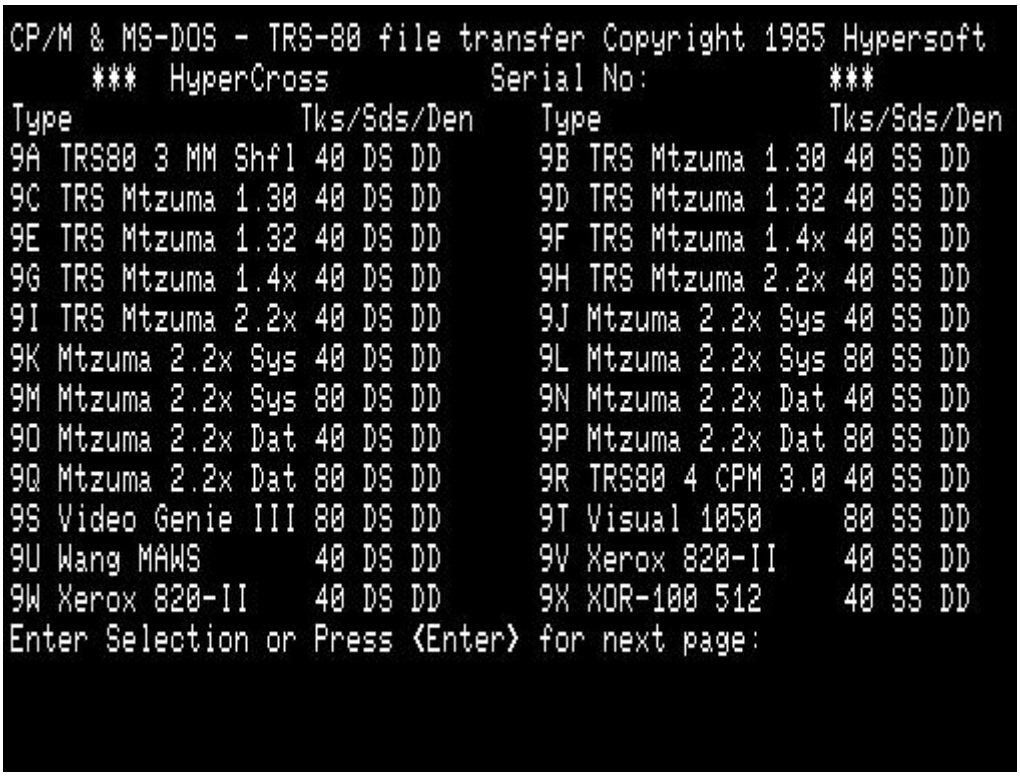
One thing you will find is that most CP/M software out there is not in a format readily readable by Omikron CP/M 2.2 because it still uses a 35 track single density format similar, but not the same, as other Model I operating systems. In fact CP/M was infamous for it's myriad of different disk formats, and for a while it seemed that every new computer released had a new and incompatible disk format. Future operating systems like MS-DOS had the look and feel of CP/M but stuck with a standardised format, making life much simpler, and it's one of the reasons PC compatibles running MS-DOS flourished.

What we need is a program which knows how to read a ton of different formats and copy them over to disks of incompatible formats. HyperCross was just such a program by Michael Gingell and was sold during the 1980s before being turned over to public domain in the early 90s. It runs on the Model 4 and anyone who has a Mappa-1 will want this program. It's simple menu-driven design lets you chose the format of your alien disk, you can then examine the contents and tag them to be moved.

It's not perfect and some disks require two steps. Say you have a CP/M disk from a Kaypro II, you would select 3T and move your files to a TRS-80 disk, then you would select 8L, Omikron CP/M 1.4 format (which is what Mappa-1 uses) and move the files to that disk. It's then ready to try out on your Mappa-1.







As you can see from the above screen shots, just about every CP/M format is supported, and also other useful formats like MS-DOS, RS-DOS (for the Coco), and CP/M-86. There are 10.5 pages of file types supported. HyperCross can be downloaded from my web site, it's on the Mappa-1 page.

Concluding:

This history of CP/M is really the history of microcomputers and disk drives. Prior to the MS-DOS era the productive world of small computers was a story of CP/M as the dominant operating system with the most widely ranging library of specialised software. With Mappa-1 you can re-live the earliest days of computer usage for research and nostalgia.

Ian Mavric

ianm@trs-80.com

New TRS-80™ Model 4

16K Cassette-Based Model 4 **£749** Inc. VAT
Cat. No. 26-1067

64K 2-Disk Model 4 **£1499** Inc. VAT
Cat. No. 26-1069

TRS-80 Model 4 Personal Desktop Computer. Yes, it looks similar to a Model III, but compare the price and the standard features of this all-new Model 4 to any other computer in its class.

Amazing Versatility. Model 4 can run all Tandy® Model III programs without change. It will soon be able to run CP/M-based software. This means that Model III disk owners can upgrade to a Model 4 without making their present software obsolete, and have at their disposal thousands of different applications.

Friendlier ... and Faster. Because of its 64K memory and 80 x 24 display, you'll find Model 4 perfect for large business applications. Clock speed is twice that of Model III, plus there's "MemDisk", which lets you use extra memory as a superfast disk drive. The result is keystroke-quick responses instead of time-consuming disk access.

And That's Not All. The software print spooler lets you print out a job at the same time you're using Model 4 for other tasks. With the Job Control Language, you can set up a whole series of operations for your Model 4 to perform without supervision. The new Microsoft BASIC includes CHAIN and COMMON for extra-large programs.

Full - and Helpful - Documentation. You get a comprehensive owner's manual, reference card, our famous tutorial manual for BASIC programming and a tutorial manual for BASIC programming and a tutorial introduction explaining how to get the most out of your Model 4.

Super Features. Including a full 80-column by 24-line display for word processing and large spreadsheets, a built-in high resolution green screen monitor, numeric keypad, parallel interface and dual 5¼" disk drives, a new keyboard with control, caps, and three function keys, and compatability with Model III TRSDOS, LDOS and CP/M plus programs that puts thousands of applications at your disposal.

Model 4 Upgrades

64K RAM Kit For Disk Model 4. Expands internal memory from 64K to 128K. Installation required (not included).
26-1122 ..... **£99.95**

RS232C Interface Board. Installation required, (not included).
26-1148 ..... **£79.95**

Microprocessor: Advanced Z-80A 8-bit. **Clock Speed:** 4 MHz. **Memory:** 14K ROM, 64K RAM-expandable to 128K **Keyboard:** 70-key typewriter style with control key, 3 programmable function keys (F1, F2, F3), caps key and 12-key datapad. **Video Display:** 80 characters by 24 lines (Model 4 mode), 64 x 16 (Model III mode), or double-wide 40 or 32 characters per line. Upper and lower case and reverse video. Memory mapped with high-resolution 12" green monitor, includes 96 text, 64 graphics and 96 "special" characters. **Sound:** Obtainable from BASIC with sound command including tone and duration. Toggled "keyclick" filter. **Disk Drives:** One or two built-in double density 184K 5¼" drives. **Operating System Compatability:** TRSDOS 6.0, TRSDOS 1.3, LDOS, CP/M Plus (available soon). **Programming Language:** New Advanced Microsoft BASIC with CHAIN, COMMON and more. **Input/Output Interfaces:** Cassette operation at 500/1500 baud selectable under ROM BASIC. Parallel printer interface with TRSDOS print spooler. Model III/4 Input/Output bus. **Dimensions:** 12½ x 18⅞ x 21½". **Power:** Integrated power supply, 240v AC, 50 Hz.

Model III Owners!

Model 4 Upgrade Kit. Allows Model III disk system owners to convert their computer to a Model 4. Installation required, (not included).
26-1123 ..... **£599.95**

LDOS is a trademark of Logical Systems, Inc., CP/M is a trademark of Digital Research, Microsoft is a registered trademark of Microsoft, Inc.

Tandy®

The Biggest Name in Little Computers

Send For Your FREE TRS-80 RSC-9 Computer Catalogue

Computer Marketing, Tandy Corporation (Branch UK), Tameway Tower, Bridge Street, Walsall, West Midlands, WS1 1LA.

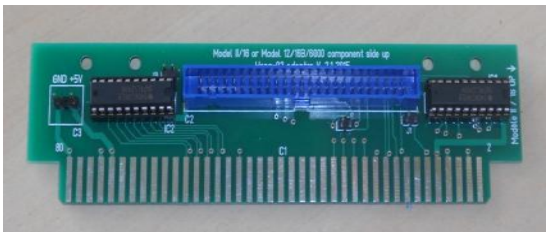
Name _____

Address _____

PCN-1

(CONTINUED FROM PAGE 1)
REVAMP, I'M NOT TOO SURE. NICE TO SEE IT AVAILABLE THOUGH.

HANS RIETVELD IS BUSY WITH HIS 'MODEL 2 GENERATION' ENHANCEMENTS. MAV HAS VARIOUS OF HIS ADD-ON CARDS NOW AVAILABLE. THERE'S FULL DETAILS OF THE VARIOUS PRODUCT AND INFORMATION ON MAV'S WEBSITE.



BOTH HANS AND PASCAL (AND I'M SURE MANY OTHER TOO) PUT IN 100'S OF HOURS OF EFFORT, DEVELOPING AND IMPROVING TANDY HARDWARE AND SOFTWARE. THIS IS, IN MY OPINION, WHY THE TANDY SCENE IS STILL SO EXCITING. QUITE UNBELIEVABLE AFTER 35 YEARS WHEN YOU COME TO THINK ABOUT IT!

IRA HAS POSTED INFORMATION REGARDING USBORNE, THE PUBLISHER. THEY HAVE MADE AVAILABLE MANY OF THEIR 1980'S BASIC BOOKS AVAILABLE FOR FREE DOWNLOAD, IN .PDF FORMAT. THIS INCLUDES MANY GAMES BOOKS WITH TRS-80 PROGRAMS YOU CAN TYPE IN.

KEYING IN LONG BASIC GAMES IS CERTAINLY A WAY TO RELIVE YOUR JR. HIGH SCHOOL YEARS. WELL DONE USBORNE. HOWEVER, IN AN IDEAL WORLD, IT WOULD HAVE BEEN A SUPER IDEA TO RELEASE THEM INTO THE PUBLIC DOMAIN, AND PUBLISH THE SOURCE CODE IN A DOWNLOADABLE DISKETTE FORMAT.
IRA SUGGESTED THAT IF ANYONE WANTS TO OCR THE PROGRAM LISTINGS AND SHARE THEM BACK TO USBORNE AS A THANK-YOU?
THERE ARE ACTUALLY MANY MORE USBORNE BASIC BOOKS AVAILABLE THAN I WAS PREVIOUSLY AWARE OF. MOST, IF NOT ALL, APPEAR TO CONTAIN LISTINGS FOR THE MODEL 1, 3 OR IV BASIC.

IRA HAS ALSO LISTED, ON EBAY, VARIOUS ITEMS HE HAS FOR SALE. FULL DETAILS ARE ON HIS WEBSITE.

AND NOW, DOWN TO THIS ISSUE. WITH NEARLY 70 PAGES, IT'S REALLY A MATTER OF WHERE TO START.
NEW HARDWARE DETAILS FROM MAV (M1 CP/M BOARD), HANS (M2 ADD-ONS), BAS WITH A COCO/DRAGON64 CONVERSION AND PASCAL (M1 512K RAM BOARD). HARDWARE REPAIRS, SOFTWARE REVIEWS AND NO LESS THEN 3 ONE-LINERS!
PLEASE DON'T FORGET, ALL THESE GUYS RELY ON YOUR SUPPORT, INTEREST AND ENTHUSIASM IN ORDER TO SUSTAIN THEIR EFFORTS.
I DO SO HOPE YOU ALL ENJOY THIS ISSUE, SURELY THERE REALLY IS SOMETHING FOR EVERY TANDY ENTHUSIAST!
ANY CHANCE **YOU** COULD SUBMIT AND ARTICLE FOR THE JUNE EDITION AND KEEP UP THE MOMENTUM?

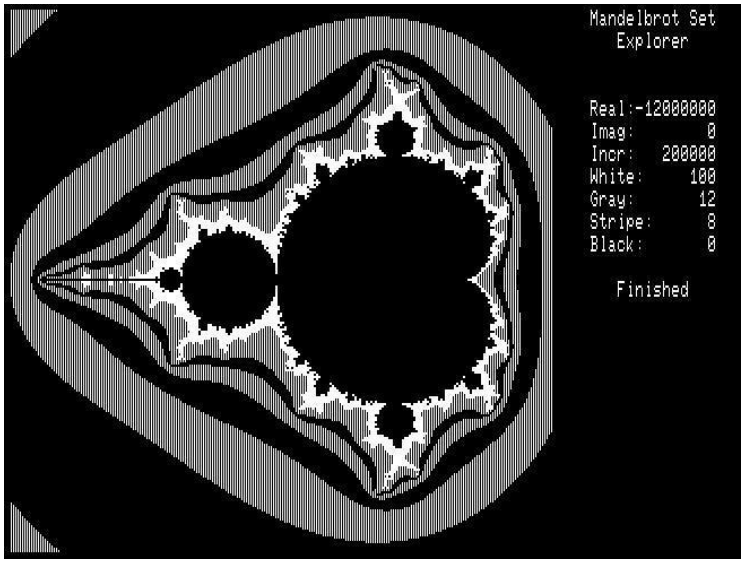
TAKE CARE EVERYONE

DUSTY

LOOKING FOR HIGH-RESOLUTION GRAPHICS FOR YOUR TRS-80 MODEL 4/4D?

Then you simply must have the

"Improved Grafyx Resolution Solution!"



-Improved version of the established Grafyx Solution GX 4.2 board
-640x240 resolution for a total of 153,600 pixels, High Bandwidth
-Easy install in minutes inside your Model 4, Gate-Array Model 4, or 4D
-4P version is just around the corner - stay tuned!
-Compatible with most M3 and M4 Hi-Res software for Radio Shack HRG
-Designed in Belgium and proudly built and shipped from Australia or UK

Order yours today exclusively from:

UK and EU: BetaGamma Computing: bgialopsos@aol.com

Rest of the world: http://ianmav.customer.netspace.net.au/hires/hires.htm

NEW REDUCED PRICE USD$165
(DUE TO AU$ EXCHANGE RATE CHANGES)
WITH ADDITIONAL DISCOUNT FOR FURTHER PURCHASES.
CHECK WITH IAN FOR FULL DETAILS

Mav Quarterly

Updates on the state of the Biz, in no particular order

by Ian Mavric

Price Changes FreHD Products: The Australian dollar continues to fall against the US dollar so I've lowered the price of the FreHD to $USD179.99 for US buyers. Also, on the FreHD Emulator page on my website I've put a Quick FreHD Pricelist so you don't need to scroll all the way through the screen looking for prices for everything. At this time I also reduced the $USD prices for all FreHD-related items consistent with the $A devaluation.

Hans-0x Products: Hans Rietveld, with the help of Pascal Holdry, came up with a series of four adapters. Two allow adding a Lo-Tech IDE hard drive to a Model II/12/16/16B or 6000 (Hans-01 and Hans-02), as well as a two adapter set (Hans-03 and Hans-04) which add three expansion slots to the Model 12 computer, which is usually missing it's card cage. These are now available from my site in either unpopulated PCBs or fully built Hans boards. For more information go to the new VCF site: http://www.vcfed.org/forum/showthread.php?49641-Model-12-BUS-BOARD

Shipping Times: Over the last 6 months I have noticed generally very long shipping times to the EU except for UK, which seems to be relatively consistent at about 10 days. Shipments sent recently to France, Italy, Greece and Germany have been taking some 5-6 weeks even though Australia Post website states they are supposed to be a 10-14 days delivery time. One shipment to Italy was also sent at the higher-cost Express International rate and still took 10-14 days instead of the quoted 3-5 days. My advice, save your money on the Express service. Also in most cases the reason for slow delivery time is items sitting in Customs in your country waiting processing. The actual transit time from Australia to your country is pretty quick.

Germany, Brazil, Mexico, Argentina: Your postal services are consistently losing my parcels and while I have in the past cheerfully sent replacements at my expense, this will now stop. All shipments to these countries will require additional $10 to pay for tracking service so you can follow your shipment and if it gets lost, you need take it up locally with the postal service in your country. I don't like to have to do this but when items get lost it is not my fault, and it ruins my good reputation. Each time I investigate lost parcels with Australia Post it is confirmed the item was shipped promptly and received at the distribution centre in the relevant country, after that I have no control.

Monday is Shipping Day: Starting from this year, I only ship out on Mondays. You may think this is a sign that business is bad but it's actually the opposite. Previously I have had to drive to the post office four or five days a week, during my lunch hour to post things, which cuts into my lunch time, and wastes petrol. It is unfortunate that TRS-80s aren't my main income but has to fit in with my normal day-job. Sending a bunch of parcels on one day instead of one or two a day over 5 days makes perfect sense.

Mappa-1: My first new product for 2016 is the Mappa-1 CP/M memory re-mapper research tool for the Model I. I've written another article elsewhere about it, but its in stock and ready to ship, and only costs $USD109.99 ($A149.99). I have noticed one quirk that it does not boot properly in a Model I modified with the FreHD Auto-Boot Eprom. The best solution is to have a 2nd Model I 16K Level II keyboard unit (most of us do anyway) and use that for your CP/M work.

Unpopulated PCBs: As usual I order too many of any given PCB in case there is a high demand for it or to solve a particular problem. This was the case when early in the Improved Grafyx M4 HRG boards I found some software didn't quite work correctly: AnsiTerm, Slot4Mod4, VidPokr4. The reason for this I suspected was due to the programs checking for which type of PCB is installed (Radio Shack or Micro-Labs) by counting memory and concluding that if it found 32K memory it was a Radio Shack board. If it only counted 20K then it decided it was a Micro-Labs board, so when it counted Improved Grafyx 32K memory and thought it was an RS board it crashed. My solution was to make a small number of 20K versions until the problem was solved. Now that the problem was analysed and solved by GP2000, I can proceed to manufacture the 32K Improved Grafyx boards and sell the spare 20K GX 4.2 and 5.1 PCBs for people who like to make their own hardware. Drop me a line if you are after one.

Coming in April - Model III HRG: People have asked me for a Model III version of the Improved Grafyx HRG with single-chip memory and I've been working on it. It should be released by the time the next TRS8Bit comes out.

The Right Stuff: One stop TRS-80 shop for restored systems, hardware upgrades, spare parts, books, magazines, software, cables and advice.

Ian Mavric

ianm@trs-80.com

Introducing - The DraCo 64 from BetaGamma Computing

Foreword

I Have always been puzzled by the fact that the communities of the TRS80 Color Computer and the Dragon 32 have always remained fairly distant, in reality as the Hardware is so similar both communities would benefit by joining together for many projects both Software and Hardware.

Recently, a clever guy going by the name of Darren Atkinson developed the CoCoSDC cartridge, a clever device that utilises SD card storage with a very good Floppy Emulator implementation allowing Color Computer disc images to be mounted and read or written too. It also includes on-board FlashROM with 6 free available 16K slots from a total of 8.
From day 1 the Firmware supported the Dragon Floppy controller too with a DIP switch enable option available, but until recently no DOS was implemented until Pere Serrat implemented DOSPLUS 5 with both Drive wire and SDC support.
The CoCoSDC was now truly multi-platform, wouldn't it be nice to have a multi-platform computer too.

Enter the DraCo 64

The Dragon 32 was manufactured in Wales during the early 80's and was another hopeful competitor in the UK's then booming home computer market.
Based around the reference Motorola design for the **MC6883 Synchronous Address Multiplexer (SAM)** chip it therefore resembled the Tandy TRS 80 Color Computer to a very high degree as that was also based on the same reference design.
The Dragon's firmware was also licensed from Microsoft, and although syntax wise it resembled Extended Color Basic it had many changes such as the Tokens used BASIC keywords, completely incompatible ROM Routines, and the keyboard was mapped and therefore read differently…
So…no plugging CoCo Cartridges or loading CoCo Cassettes into a Dragon 32 or vice versa.
Quality wise the Dragon 32 left a lot to be desired but had 3 BIG advantages over the TRS 80 Color Computer, it sported a real full size keyboard, it had a Composite Monitor output port and came with 32K RAM and Extended Color Basic as standard. It was also a lot cheaper than the TRS 80.
Today Dragon 32's are available in plentiful supplies in the UK and quite cheaply, therefore the DraCo 64 prototype began by the acquisition of a suitable but poorly Dragon 32.
The Main Goal of the DraCo 64 is to save desk space, have two systems in one and use a Single CoCoSDC in Both Modes.

The complete upgrade process requires the following individual steps.

Upgrade RAM from 32K to 64K

This involves removing the on-board 16x 4116 DRAM chips and replacing them with 8x 4164 64Kx1 DRAMS.
The 4116 DRAMS use 3 supply rails, +5V, +12V and -5V, whereas the new DRAMS

are single supply +5V, so the board must be modified to reroute the power supplies, remove redundant decoupling capacitors and wire up the A7 Address line of the 4164's. It is important to use 128cycle refresh RAMS as that is the **SAM**'s limit.
There are various Dragon 32 PCB's in existence, some easier to upgrade than others but schematics are available on line.
Finally, a modification must be made to the Address decoder to allow pure 64K RAM mode for use by OS9 / NITROS9, in line with the Dragon 64 and CoCo 64K models.
At this point we now have a Dragon 64 without the 48K BASIC option.

Upgrade the Firmware ROMS
Both the CoCo and Dragon 32 hold 2x 8K ROMs for their firmware, these need to be replaced with 2x 16K ROMs that hold the firmware for both platforms, concatenated and programmed in a manner that A13 on both ROMs will be switched HI or LO in parallel to select the correct firmware dependant on the mode the user wishes to use.
As you will see later it makes sense to have the CoCo firmware active when A13 is HI.
A minor tweak is needed to the Color Basic ROM to inform the **SAM** of the RAM layout, standard images are used for both Dragon ROMs and the Extended Color Basic ROM.
The Original ROMs are 2364 devices and these are replaced with 27128 devices, apart from going from 24 pin to 28 pin there is also some minor rewiring required. This problem can be solved by using purchased convertors, as used in the prototype, handy for finalising the ROMs, or by modifying the New ROMs once they have been programmed and verified.
At this point we now have a dual Personality Dragon 32 that can boot as an Enhanced Dragon 32 or a 64 K TRS 80 Color Computer, however typing on the keyboard in CoCo mode will produce garbage on the screen due to the different mapping.

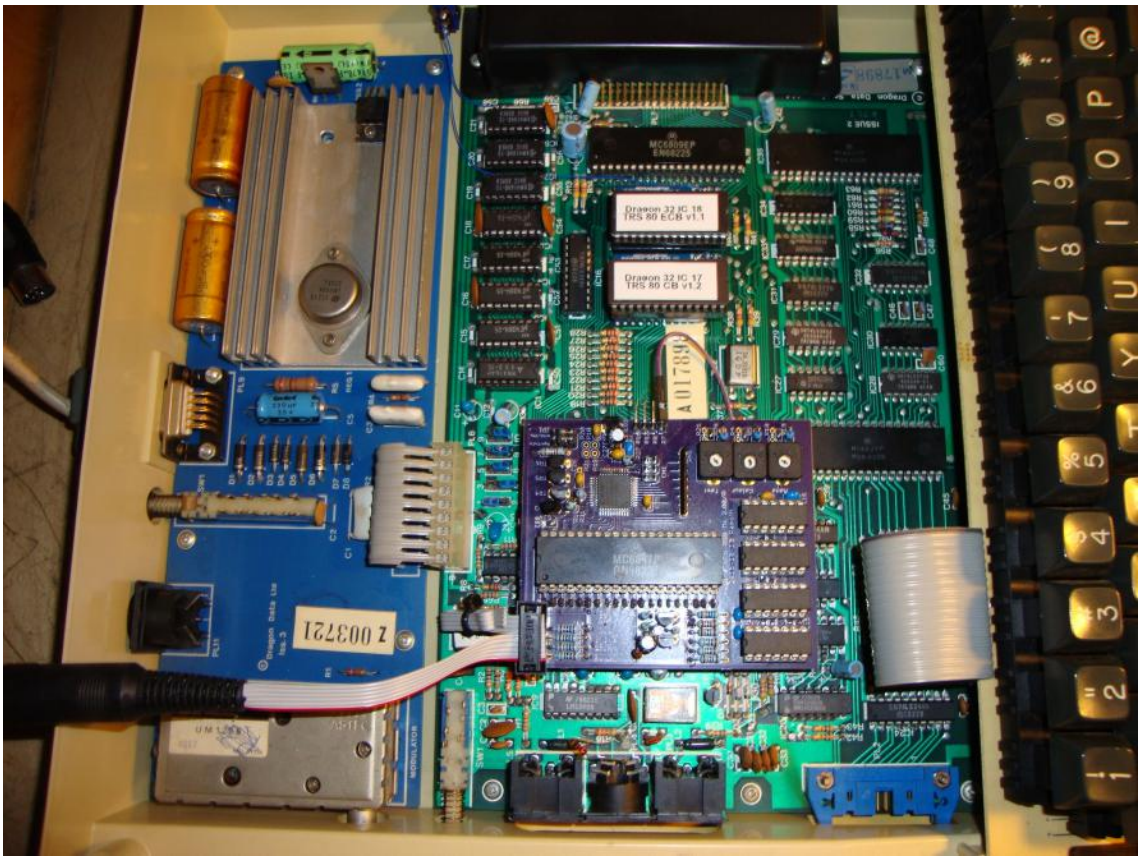
Install Automatic Keyboard Remapper
This is the fun bit.
A total of 6 lines between the keyboard and motherboard need to be rerouted for correct keyboard operation, when developing the prototype, I used a small hard wired PCB between the keyboard and motherboard, in practice though an automatic switching solution triggered by the ROM mode selection would be the ultimate solution.
A small slim PCB was hand wired, containing 2x 3 channel multiplexors, all simultaneously switched to provide both mappings dependant on the enable lines state.
HI Enable selects CoCo keyboard mapping
LO Enable selects Dragon keyboard mapping.
As with the ROM Enable A13 lines, HI also equals CoCo ROM mode so the same signal can be used to switch the Multiplexors.
Note. *The Dragon 32 keyboard PIA circuit is slightly different to the CoCo in that 2 capacitors sit between PA0 and PA1 to Ground on the PIA side, on the CoCo these capacitors are fitted after some inline inductors.*
This difference means that these 2 capacitors must be removed from the PIA lines for the keyboard to function correctly in CoCo mode, they do not affect operation in

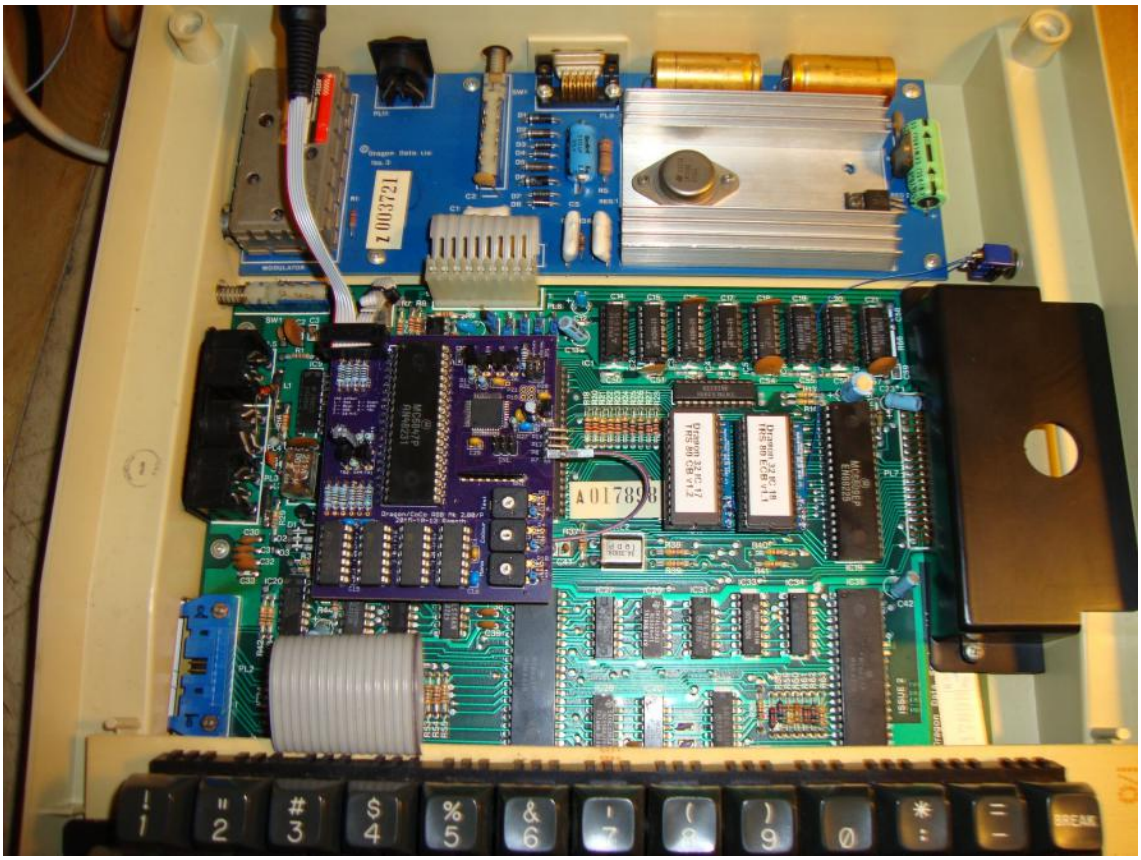
Dragon mode and Joysticks work perfectly in both modes.
Professional double sided PCB's have been manufactured for the remapper for use in future systems.
At this Point we now have a DraCo 64.

Finally

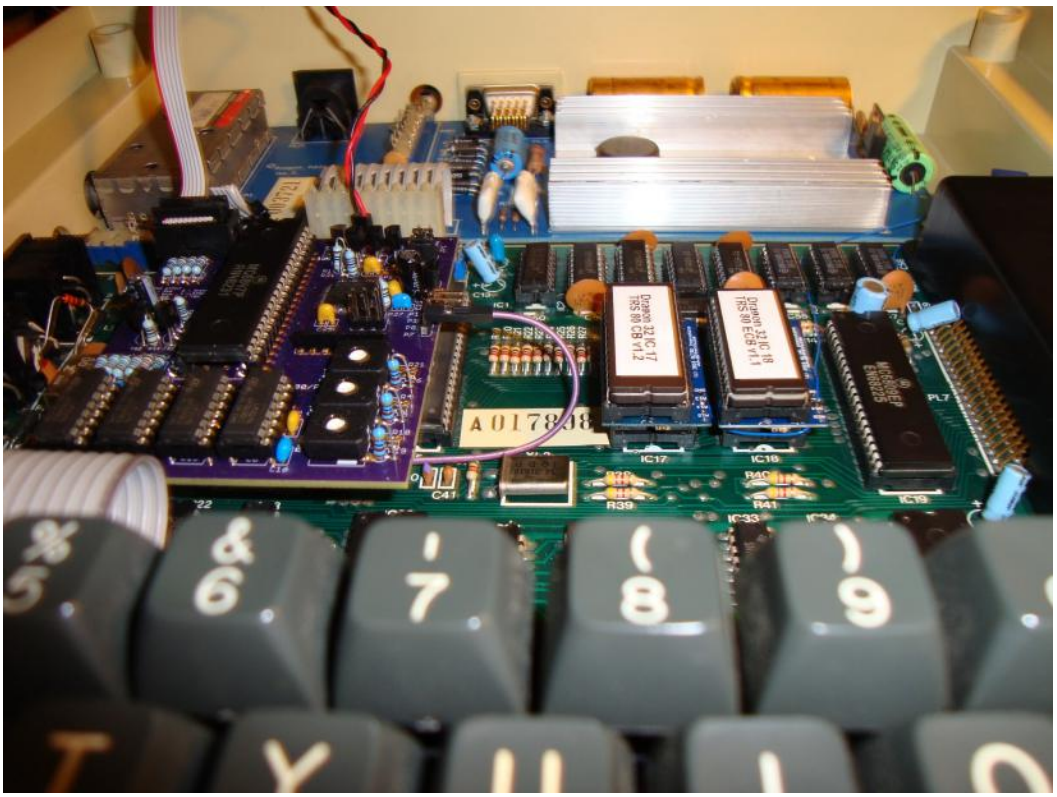
I have included some photographs with description highlighting the modifications for you to enjoy.
The eager ones amongst you will also notice a strange purple PCB in the photographs, this is an RGB Video Board being developed by Prime that maybe available in the near future, it provides a perfect clear output never before seen on either CoCo or Dragon.
If anyone is Interested in having a Dragon 32 converted or would like a complete ready system, then please get in touch.

Credit and Thanks to Phill Harvey Smith (Prime), for the auto keyboard remapping.

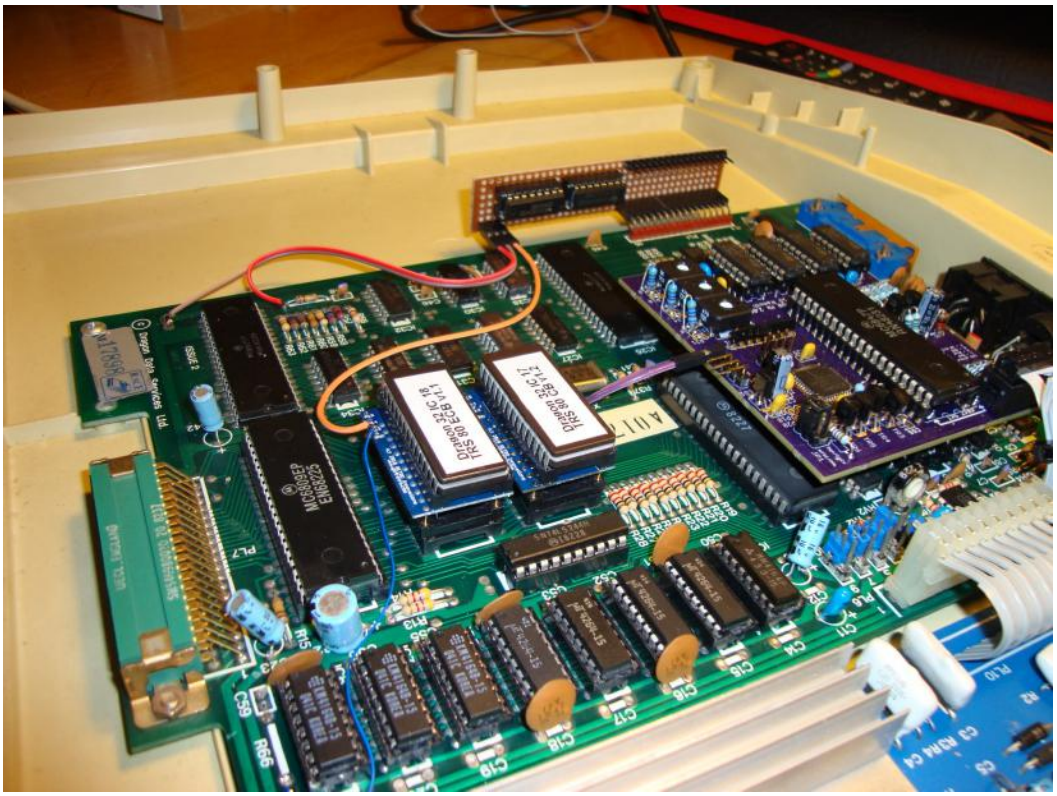




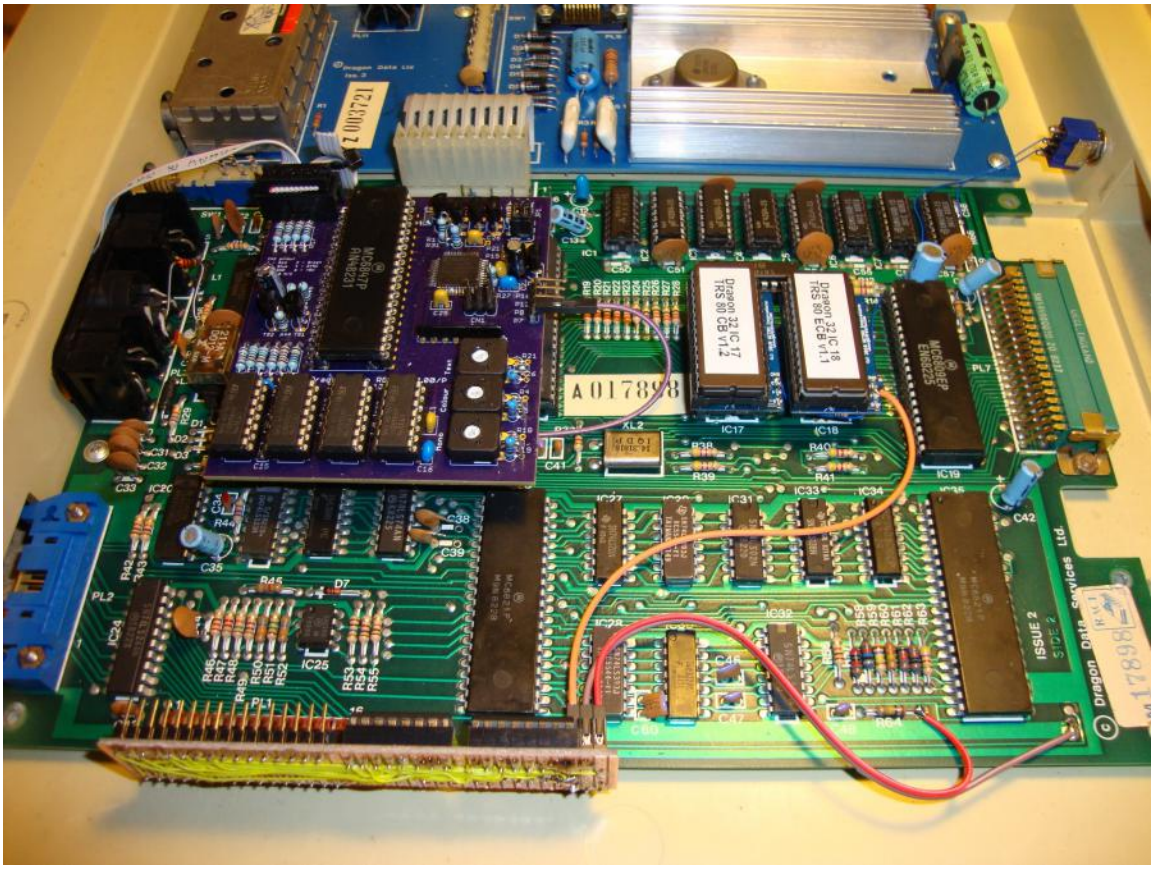
Some photographs of the upgraded Dragon Main board, New ROMs and Mode switch fitted on the Right hand side.

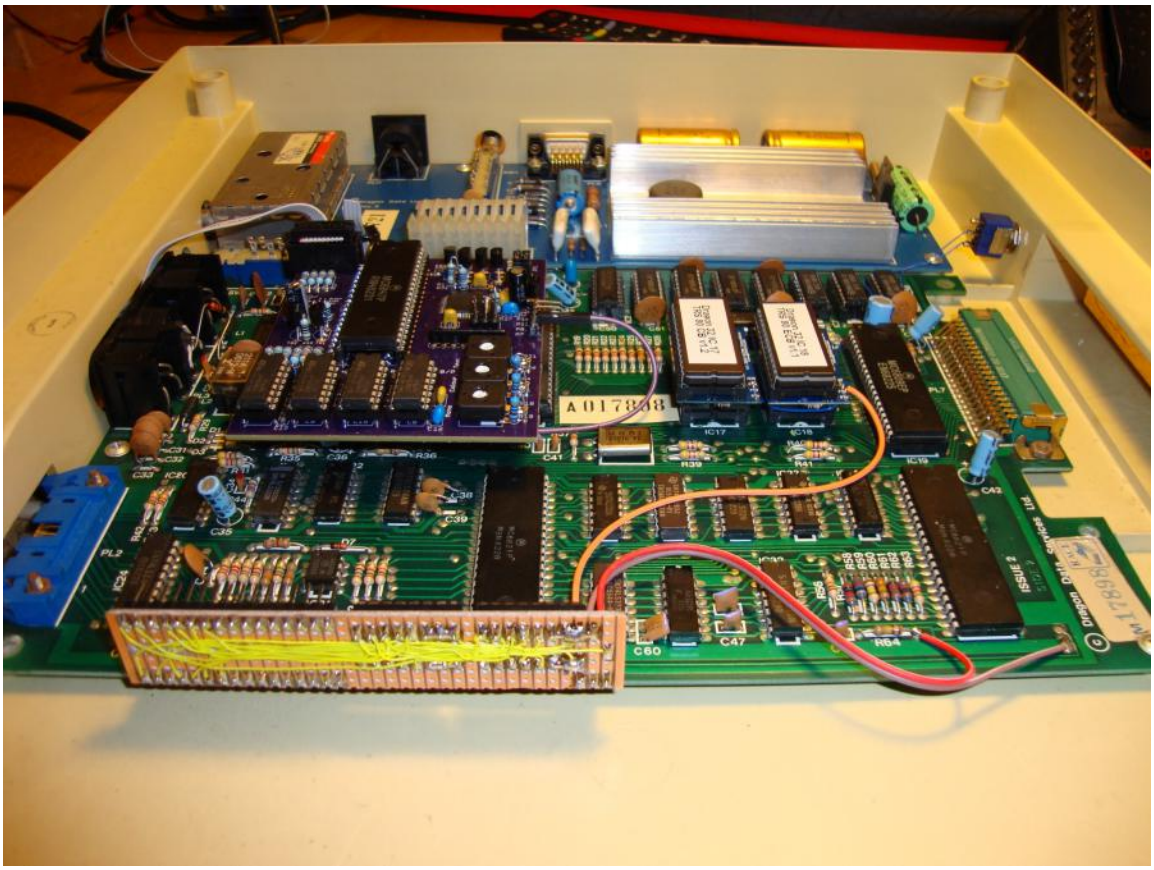


New ROMs with Convertors



New 4164 64Kx1 DRAMS, New ROMs and view of prototype auto keyboard remapper.





Some close-ups of the revised motherboard and prototype auto keyboard remapper.

Dragon 32 Mode

64K Color Computer Mode

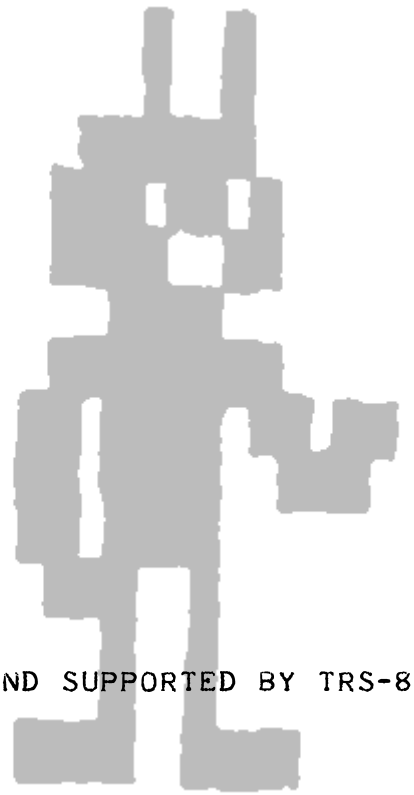
The *BEST* in TRS-80s

Call The Right Stuff

Ask for Ian
The number is +61 416 184 893

That's The Right Stuff
And he's in Melbourne

http://ianmav.customer.netspace.net.au/trs80/



PRODUCED AND SUPPORTED BY TRS-80.ORG.UK

TRS8BIT

PRODUCED BY TRS-80.ORG.UK

FLAMING JUNE IS HERE, SO WELCOME, EVERYONE, TO OUR 42ND EDITION OF TRS8BIT!

ARE YOU FOLLOWING 'TRASH TALK'? THERE'S A LINK FOR YOU TO FOLLOW ON OUR WEBSITE. ISSUE 5 HAS TRASH TALKERS, PETER CETINSKI, RANDY KINDIG, PETER BARTLETT AND IAN MAVRIC. THE MAIN FEATURE IS AN INTERVIEW WITH THE FOUNDER OF BIG FIVE SOFTWARE, BILL HOUGE. REALLY BRILLIANT STUFF GUYS! ROLL ON ISSUE 6.

THE DUTCH TRS-80 RELATED SITE, WWW.TRS-80.NL, RUN BY THE DUTCH TRS-80 USERS GROUP HAS BEEN UPDATED. (THERE'S A LINK ON OUT WEBSITE).

THE DUTCH TRS-80 USER GROUP WAS ESTABLISHED IN 1978 TO SUPPORT THE NEW HOME COMPUTER FROM TANDY, THE TRS-80. THE USER GROUP MADE A QUICK START AND GREW TO OVER 4000 MEMBERS. IT PUBLISHED A BI-MONTHLY NEWSLETTER, 'REMARKS', WHICH WAS THE MAIN SOURCE OF INFORMATION FOR MANY FIRST-TIME COMPUTER USERS. THE YEARLY USER DAY WAS THE MAIN EVENT FOR BUYING HARDWARE, SOFTWARE AND EXCHANGING KNOWLEDGE.

THE CURRENT USER GROUP HAVE SETUP A NEW WEBSITE AND THE COMPLETE ARCHIVE OF THEIR 108 NEWSLETTERS IS AVAILABLE ON LINE.

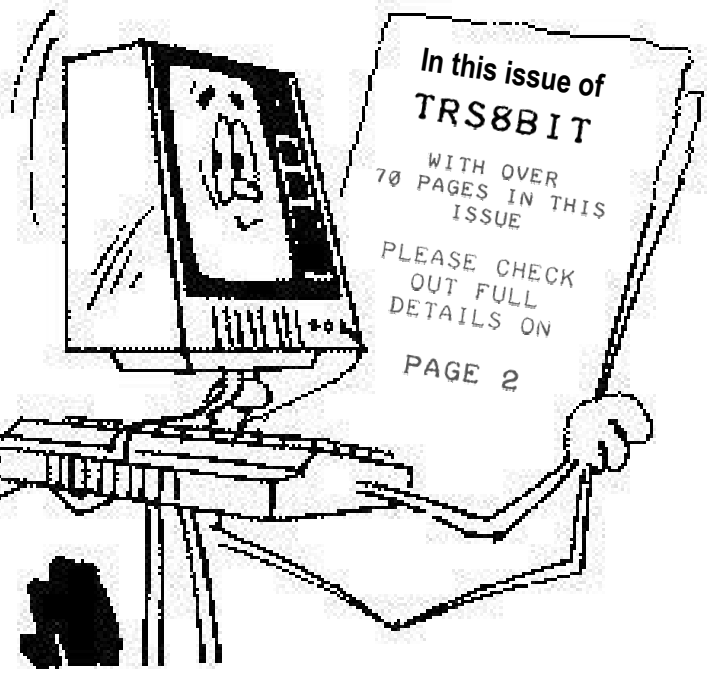
ALTHOUGH THE SITE IS IN DUTCH, DON'T FORGET, THOSE USING CHROME, GOOGLE WILL OFFER TO TRANSLATE THE PAGES AS YOU MOVE AROUND THE SITE!

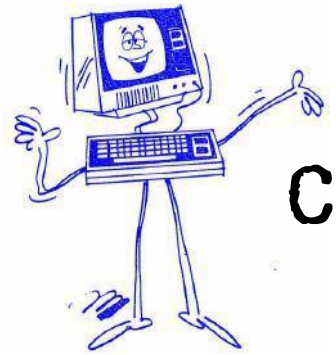
MATTHEW REED HAS RELEASED AN UPGRADE TO TRSTOOLS. THE LATEST VERSION IS V1.13 AND IS AVAILABLE FROM HIS WEBSITE. I'M STILL HOPING HE'LL UPGRADE HIS EMULATOR TO ALLOW THE INCLUSION OF, YEH, YOU GUESSED, AN ACULAB FLOPPY TAPE.

FOR THOSE OF YOU WHO HAVE NOT HAD A CHANCE TO SEE JIM MCGINLEY'S GDC PRESENTATIONS ON TRS-80 GAMES, BOTH HIS 2012 AND 2016 PRESENTATIONS ARE AVAILABLE ONLINE. I'VE PUT A LINK TO THE FIRST ONE FOR YOU TO USE ON THE WEBSITE.

PLEASE SEE PAGE 2 FOR FULL DETAILS OF THIS ISSUE. THE TRS-80 WORLD JUST KEEPS ON GROWING!

TAKE CARE -
DUSTY





CONTENTS

SPECIAL FEATURE

In Mav's Workshop:

日本製TRS-80モデルI

(The Made in Japan TRS-80 Model I)

by
Ian Mavric

Unverifiable History[1]

1979 and 1980 were years of great growth for Radio Shack's little computer for the masses, the Model I. As the computer gained popularity, sales soared and as more software came available, more "end users" bought the machine often to run a specific application, rather than to learn to program.
It's no surprise that more Model Is were sold in 1978 than 1977, and more again in 1979 than 1978. It was looking like more were going to be sold in 1980 than 1979. To augment the TRS-80 line Radio Shack designed and released the Model III in 1980, some people say as a successor for the Model I but in fact the Model I was selling so well that they planned to sell them alongside each other, and orient the sales of each machine to more appropriate buyers.
Here is the first problem, Tandy only has one factory in Texas making computers, called "TCA" and it was only geared toward making one model of computer at a time.

TCA, I presume which stands for Tandy Computer Assembly, was the factory where the Model Is were made, then the Model III, then the Model 4, and finally the Model 4 gate-array version.
Those who worked there recall the place as being small, hot, and the required assembly rate, quite demanding. In the lead-up to Model III production starting, a solution needed to be sought to continue Model I production.
As was the case with electronics at the time, outsourcing a successful product to Asia was a popular way to bring costs down. I speculate that sometime around mid 1979 Radio Shack made the decision to have Model Is produced by Japanese manufacturer TEC.

TEC stands for Tokyo Electric Co. (Ltd.), and it's not related to the Japanese electronics giant TEAC. (Some people get the companies mixed up as Tandy Corp. purchased disk drives from TEAC which they used in 240V Model I disk drives, so it was possible in countries other than the USA to end up with a TEC TRS-80 Model I, with TEAC disk drives, all purchased from Tandy Electronics.) TEC ended up having a long history with Tandy Corp., being the makers of the Coco 1 disk drive, as well as the FD-501 and -502 Coco disk drives, and the FB-503 disk drives in the TRS-80 Model 4D. They may have made items for Tandy that I have not encountered yet. TEC would eventually market their own electronics and microcomputer products under the name we know them as today, Toshiba.

TEC took the basic design of the Model I and improved it, however it still fit in the same TRS-80 keyboard enclosure. In fact the final product is indistinguishable from the US Model I, unless you turn it upside down and ready the information label, or open it up. I'll talk more about the differences between the machines later, but from now I'll refer to the US Model I as M1, the Made in Japan Model I as the J1, and the Japanese domestic market Model I as the JDM1.

I would guess that by time the J1 was completed, tested and approved for production by Tandy Corp., the M1 was starting to find itself in hot water with the FCC. For those who are unfamiliar with the story, the FCC has a meltdown when it found out that many early micros output broad spectrum RF interference due to inadequate internal RF shielding. It should be pointed out that the M1 was not the only computer affected, but Radio Shack was given a stern warning, fix your computer by December 1980 or stop selling it in January 1981. Some hard decisions needed to be made and when it was clear that the Model III was a hit, it was decided that change the idea that the M1 was to be sold alongside the Model III, and to just quit making the M1 and they would naturally sell out around the end of 1980. I speculate that around this time that TEC didn't get the memo to not make the J1. The J1 still lacks the required RF shielding and so was not going to be able to be sold in the USA. It's not indicated anywhere how many J1s were made but I would hesitate a guess that maybe an order for 25,000 was placed before Tandy realised the mistake and issued the stop production notice.

What do you do if you have a large number of computers which can't be sold in the US market? It helps that they are a popular computer, selling well in other parts of the world. Around this time I noticed Tandy Electronics in Australia phased out the original M1 configuration (Keyboard+CPU/monitor/tape deck) and released a new product into the TRS-80 line, the J1 system came with power supply, TV modulator, and a cassette cable and carried the catalog number 269-9006.

The J1 unit itself carries the catalog number 26-1006A. it sold well and appeared in the 1981 and 1982 Tandy Electronics catalogs. Correspondence with other TRS-80 enthusiasts confirmed the J1 was exported to most western countries including the UK, continental Europe, New Zealand. This is also confirmed by watching and examining eBay listings for Model Is in Australia, UK and Europe, the J1 shows up from time to time in these countries, but never in the United States. At the sales rates the M1 in the USA was achieving in 1980 I guess it would take a 18 months to two years to sell out the J1s.

A Closer Look At The J1

The upper and lower case "clamshells" and the keyboard bezel are the same as the M1. The keyboard is the highly regarded ALPS sculptured type as used in the 2nd generation of M1s. The cover for the J1 expansion bus has a sticker over it, which is black with silver writing "REFER TO THE USER'S MANUAL FOR WARRANTIES AND LIMIT OF RADIO SHACK LIABILITIES" this is in contrast to the dark burgundy sticker the M1. Underneath the computer the black label with silver writing continues, stating the unit has been Made in Japan.

Pictures: label differences on the J1

Inside the J1

The first thing you notice when you lift the top off is that TEC made the keyboard, not ALPS. Close examination of the key-switches you can see they are still ALPS key switches, soldered to a TEC base, carrying the part number HEIIE011610 and its connected to the motherboard via a removable cable! Hallelujah! That permanently soldered cable on the M1 is a thing of the past, making working on the motherboard easy when you don't have to worry about straining the interconnect cable.

Picture: J1 keyboard by TEC

Turn the J1 over and examine the motherboard and you see a few changes. The motherboard part code has changed from 1700069x to TEC HEIIE011550 and the Z80 has moved about 3cm south-west from where it was in the M1.
The power supply regulation remains the same, as do the two ROM sockets roughly in the middle of the motherboard. ROM version 1.3, aka. MEM SIZE?, is installed in these two sockets.
The motherboard has roughly the same number of chips, but the layout looks cleaner and easier to work on.

Picture: J1 motherboard by TEC (shows RAM sockets, installed by previous owner - standard J1 has soldered in RAM chips)

The Good News - The Improvements in the J1 over the M1

There is some good things to be found inside the J1; in many ways it's the best version of the TRS-80 Model I, and those who own them also speak very highly of the J1 as their preferred machine. Lets look at these improvements:

Lowercase: the J1 comes from the factory with the Motorola MCM6673 lowercase character generator installed. Like the M1 you need a driver program to enable it, but no modification is needed. Tandy Electronics didn't point out in their advertising that the computer came with lowercase, which seemed a bit strange to me. Any popular DOS like LDOS or Newdos/80 which supported lowercase displayed it right out of the box.

Video Ram: the J1s biggest reduction in chip count comes from replacing the 6 (or 7 in a lowercase M1) 2102 SRAM chips with a pair of 2114 chips.

Video Sync: ever noticed how the picture on the M1 has a slight wiggle or shimmer? It varies from computer to computer, but it's always present on M1s. In the J1 the picture is rock solid, like that of a Model III. This could be related to using higher-spec components in the video sync circuit, or better quality parts in the power supply, but the difference is profound. This improvement is noticeable even when connected to the crap-tastic RCA M1 monitor.

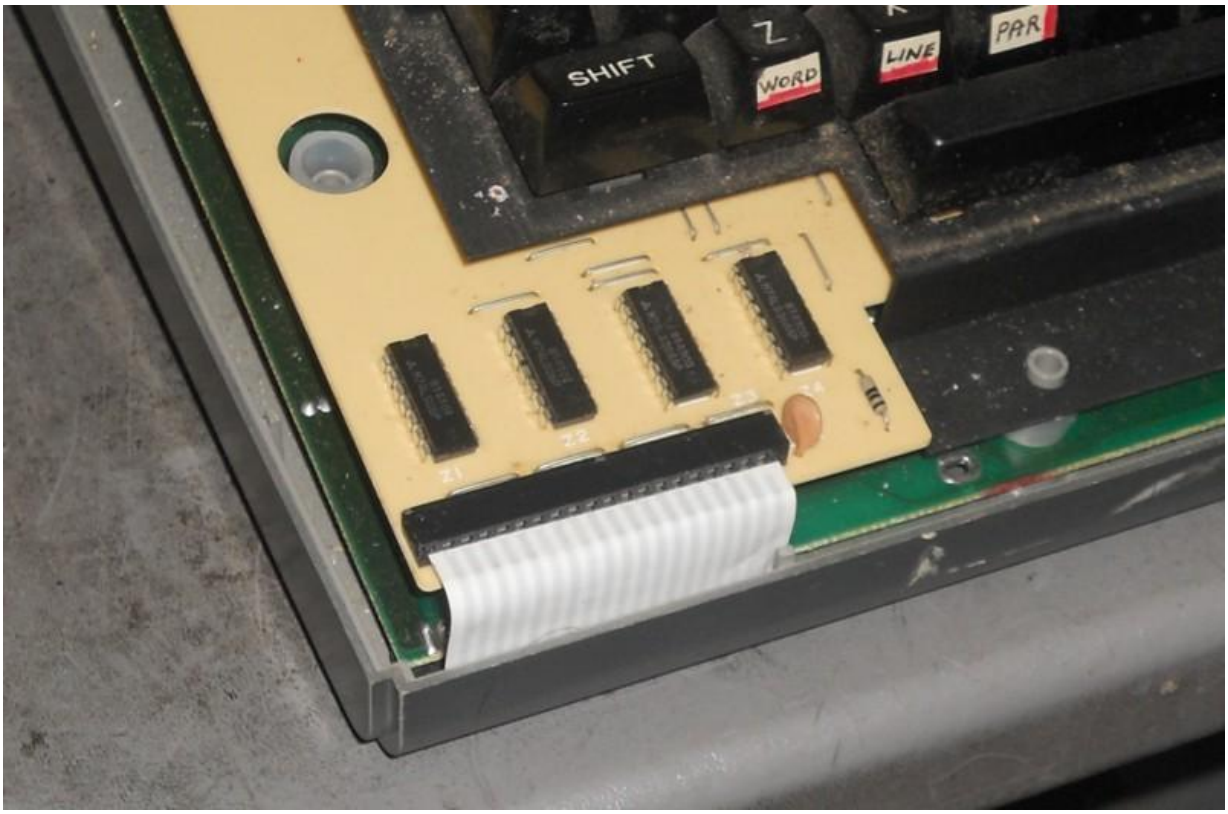
Cassette relay: the J1 has a much larger and higher rated cassette relay than on the M1. Type CLOAD on the M1 and you can hardly hear the sound of it's relay engaging. On the J1 the click is very pronounced, like that of a Model III. It can take a lot more abuse before it fails.

Cassette A-to-D converter: is a much improved design in the J1, and a lot less sensitive than the M1s, the result being that even problem tapes load easily into the J1. It's most likely based on that of the Model III which can also easily load problem 500 baud tapes.

Most if not all chips: are sourced from quality manufacturers in Japan. This probably helps to explain the extreme reliability of the J1. Nearly all the chips are either NEC, Mitsubishi, Hitachi, or TI. Even the CPU is an NEC D780C which is a locally made Z80 compatible known for its ruggedness in extreme environments (kind of the opposite to the Mostek Z80 used in most M1s).

Voltage Test points: convenient test points for Gnd, +5VDC, -5VDC, and 12VDC. A nice touch.

Keyboard cable: it unplugs, as mentioned above.

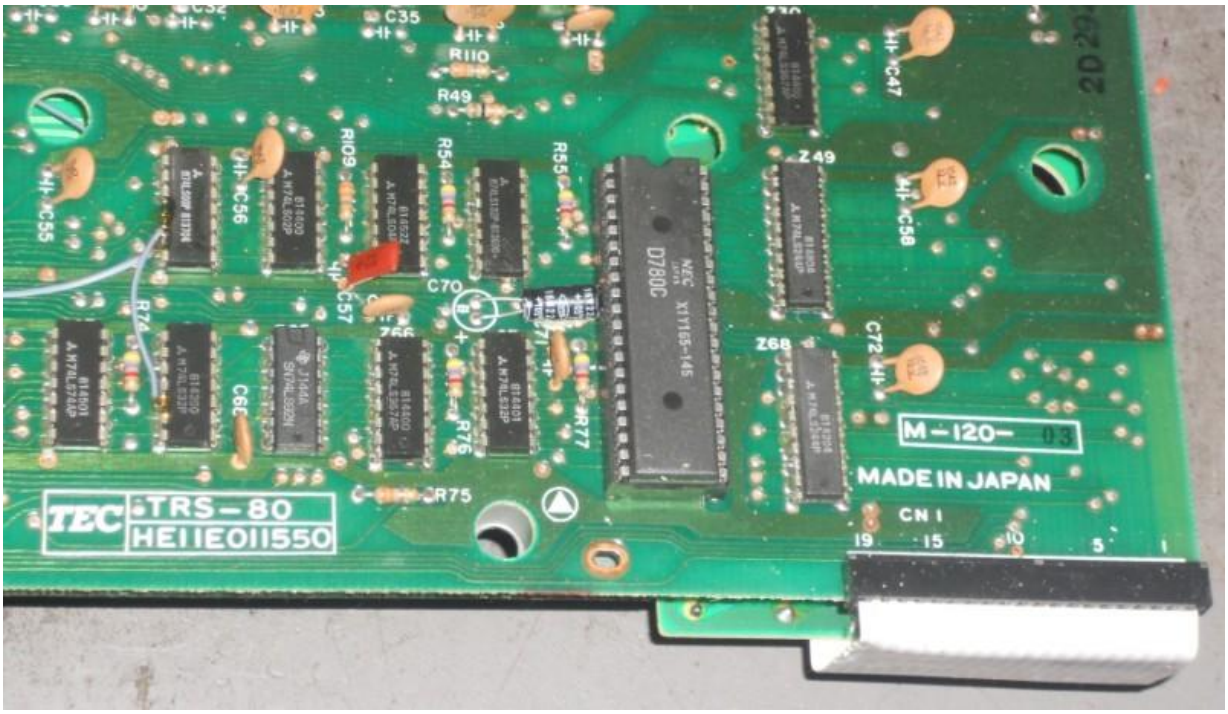


Picture: J1 keyboard - it disconnects

And the Bad News - minor criticisms of the J1 over the M1:

RAM sockets: are missing. TEC decided to just solder the eight 4116 DRAM chips to the motherboard. No 4K J1 was ever offered so they saved a few cents on the RAM sockets. Not normally a problem because they did use very reliable NEC RAM chips, but if you wanted to upgrade the system memory inside the J1 you needed to unsolder the 4116s and solder in 8 sockets before proceeding.

CPU socket moved: it's been moved about 3cm south-west of it's original position, making some aftermarket upgrades which fit into the Z80 socket (like Omikron's Mapper 1 and Anitek's 256/512K SuperMem) hard or impossible to fit.



Picture: CPU position has changed. Notice all chips are either NEC, Mitsubishi, or TI. Quality.

PCB layout: the new layout means that most modifications listed in 80micro magazine, Custom TRS-80 Book, and other sources of hardware mods need drastic revision if they are to be made to work.

TEC could have taken the opportunity to gold-plate the bus expansion connector thus helping to eliminate one of the biggest problem causing oversights of the M1, that being the solder coated bus connector. So, like the M1, you still need to periodically clean the edge connector to ensure reliable operation over long periods.

Technical / Service manual: being a new system the old easily obtainable 26-2103 Technical Reference Manual was mostly obsolete and technically minded people needed to buy a different Service Manual from Tandy Electronics that was always hard to find. In fact it was unofficially "lost" in the TRS-80 community until Brett Paulin of Spockietech brought to my attention that he had a beaten-up copy that he bought back in the early 80s. Some work was needed to clean it up for scanning but it was made available to the public thanks to Brett in 2013.

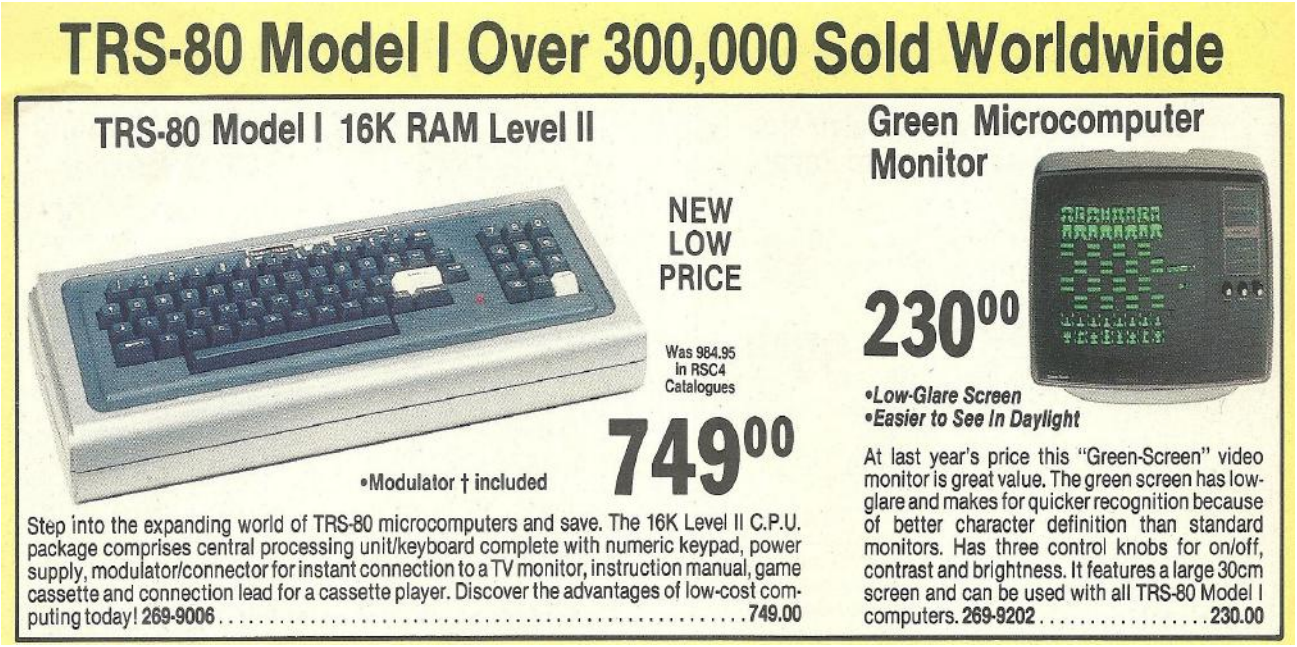
Photo: J1 Service Manual (photo courtesy of B. Paulin)

Market Segment

The J1 appeared in the 1982 Tandy Electronics Catalogue (available from September 1981) as mentioned before under catalog number 269-9006 and a price tag of $749. So where did this fit into the market segment Tandy was aiming for? The Color Computer 1 with 4K and Color Basic was $599 and the 16K Extended Basic Coco was $849. The 4K Level I Model III was also a contender in the sub $1K bracket at $999. Lets compare the offerings:

Coco comparison: Both the J1 and the Cocos could be connected to a TV set. With the J1 you got a much nicer keyboard than the Cocos chicklet keyboard, you could also fit more text on the screen (64x16 vs. 32x16) and had proper lowercase characters (Coco has no l/c but awkward reverse video representation of lowercase).
The Coco of course had colour and sound, and the J1 was monochrome and made no sound (though most people added a small audio amplifier for sound) if you went to the trouble of buying a $230 monitor you had much sharper display than the Coco could ever hope to achieve. The M1/ J1 Level II Basic is much more sophisticated than Color Basic, however it could be argued that Extended Color Basic is as sophisticated as Level II Basic and also opens easy programming ability to the Cocos hi-res graphics.
Speaking of graphics, the J1 is clearly behind the Coco in the graphics department however it should be noted that most buyers of the J1 and indeed the Model III were not buying them for their graphical ability. As for CPUs, while the 6809 is technically superior to the Z80, the Z80 in the J1 runs at almost double the speed of the 6809 in the Coco, and it could be argued that the slower speed negates the sophistication of the CPU. Good results can be achieved on either system by talented programmers.

Model III comparison: With the 4K Level I Model III you got a built-in monitor and little else worth mentioning as the memory and Basic were both inferior to the J1.
A fully expanded Model III 32K business system (26-1066) came in at $3299 and a fully expanded J1 system (with monitor, 16K EI and dual disk drives) come in at $2896, a reasonable saving over the Model III for what was essentially the same performance level, but the screen was green and anti-glare.



Old advertisement from Tandy Electronics 1982 Catalog (released September 1981)

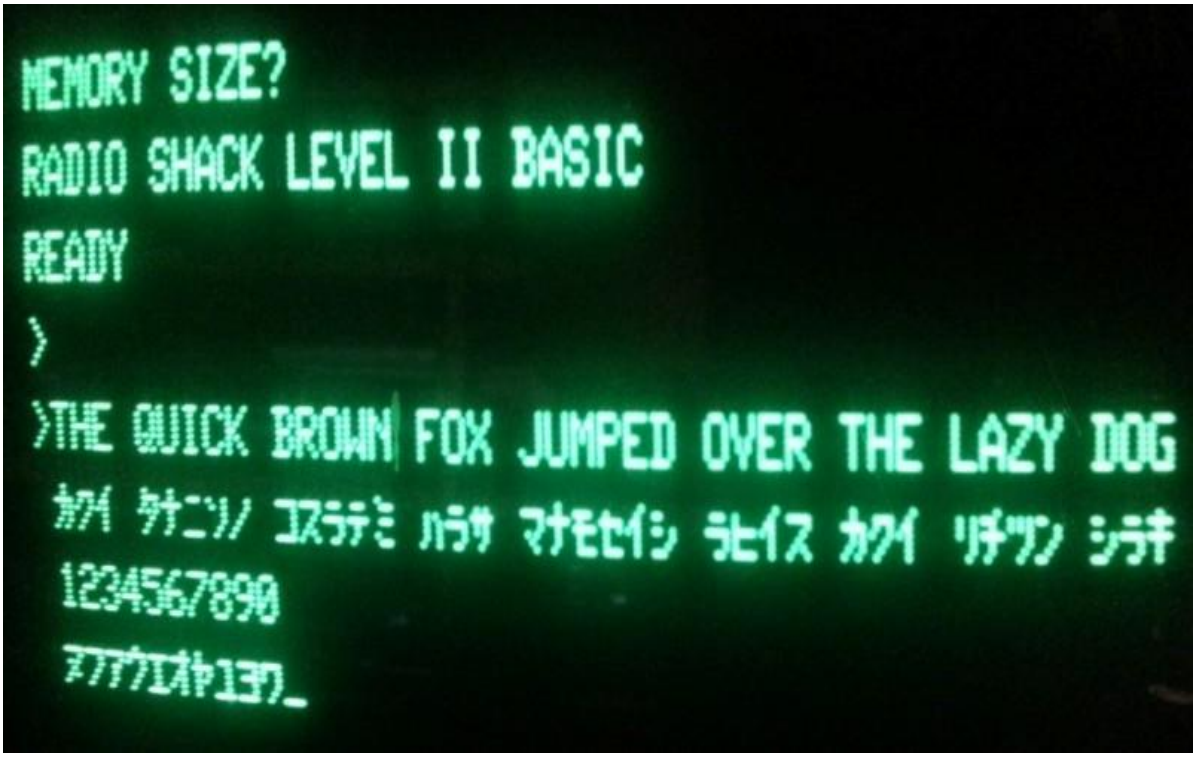
What about the JDM1?

TEC took the opportunity to sell a localised version of the J1 which had a different keyboard layout and character generation ROM installed.
The JDM1 has both English and Kana characters on the keys, as well as an additional key to switch between English and Kana character set, and a different dual-colour power led which illuminated red in English mode and green in Kana mode.
To accommodate the extra key the Enter key on the main keyboard was made smaller.
These JDM1's carried the catalog number 26-7016A and fellow TRS-80 collector John Benson from Lismore sent me pictures of his unit which was personally imported from Japan in 2012. It's one of less than ten JDM1s known to still exist.

Picture: information label from under a JDM1 (photo courtesy of J. Benson) .

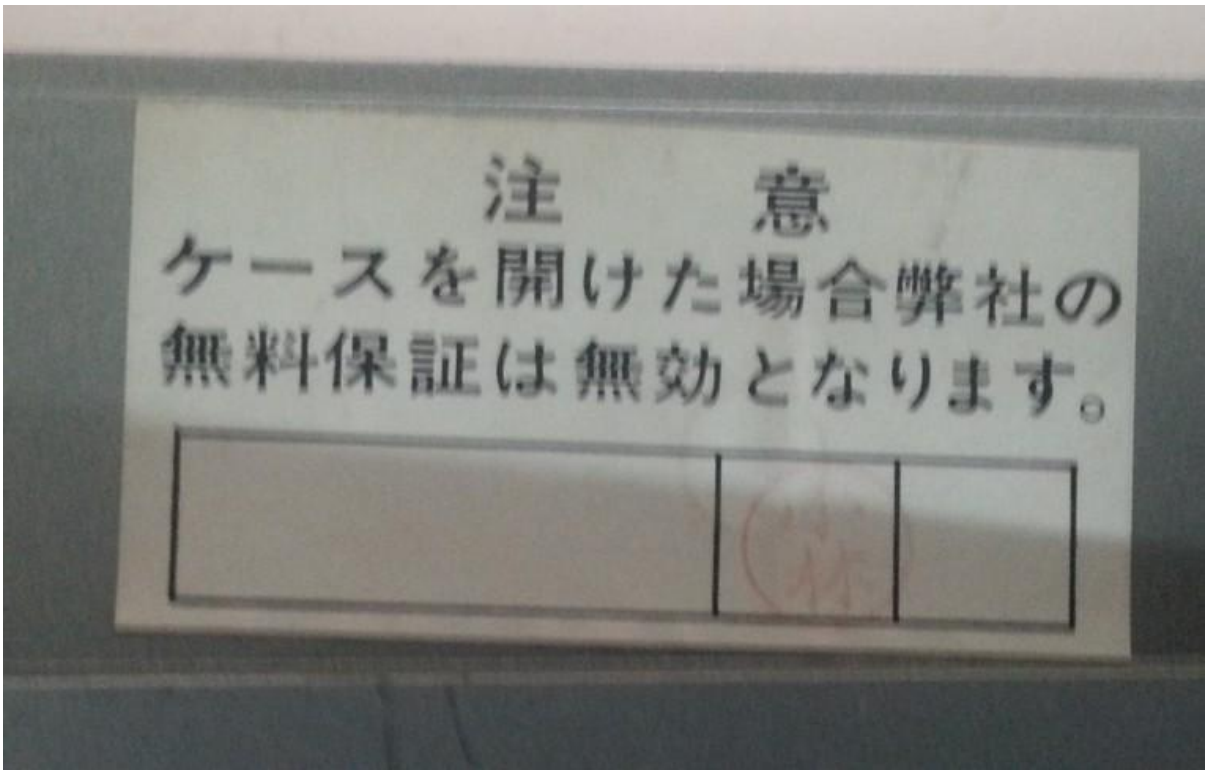
Picture: JDM1 in English mode (photo courtesy of J. Benson) .

Picture: JDM1 in Kana mode (photo courtesy of J. Benson) .



Picture: JDM1 showing it's Kana character set (photo courtesy of J. Benson) .

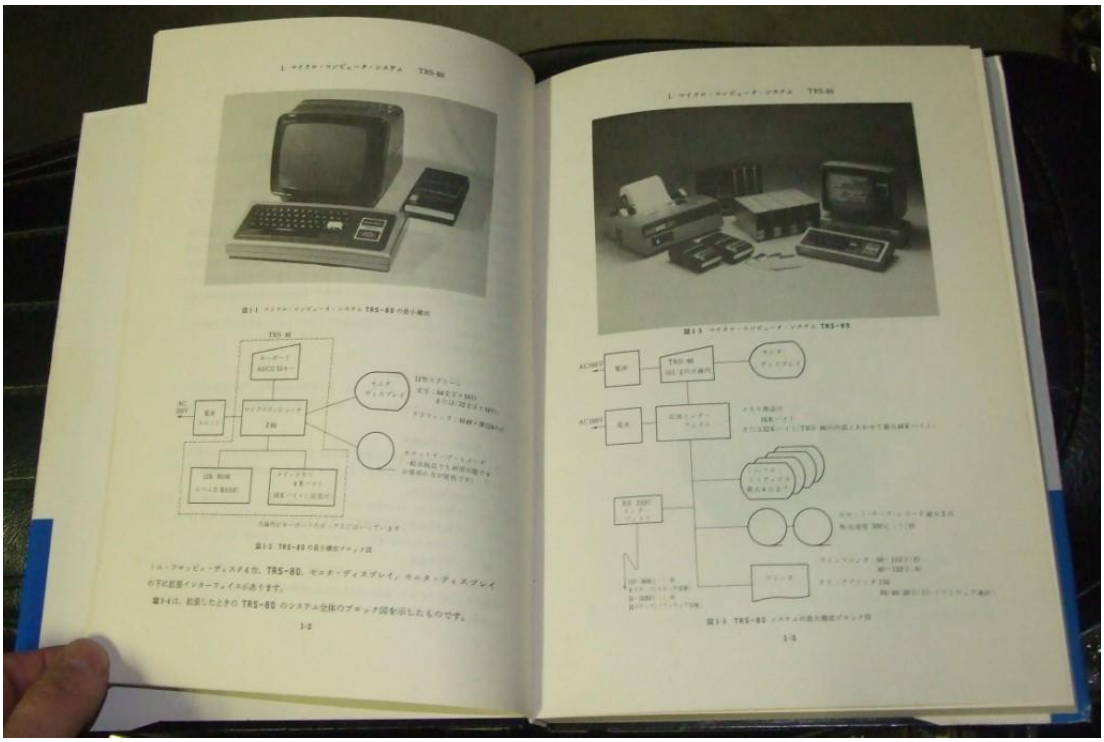
Picture: JDM1 full keyboard with both English and Kana symbols (photo courtesy of J. Benson) .



Picture: JDM1 version of the infamous Warranty Void sticker over one of the case screws (photo courtesy of J. Benson) .

A JDM1 in its home country would be a bit difficult to use with the standard English language Level II Basic Reference manual, so that was translated into Japanese and included with each JDM1:

Picture: Level II Basic Reference Manual Japanese version in Hard Cover.



Picture: from inside the JDM1 Level II Basic Reference Manual.

Concluding

These days the J1 is hard to find. Back when I started peddling TRS-80 Model Is in the late 80s and early 90s and only sourcing computers within Australia, perhaps one in three that I acquired was a J1, the rest being regular M1s. These days with the booming international trade of vintage computers between collectors and ease of shipping, M1s have flooded the market all over the world. People regularly ask me if I have J1s for sale but unfortunately I had not seen one for at least 10 years, before one arrived last week. This is given me the impetus to put together this rather long and expansive overview of the J1. Thanks go to John Benson and Brett Paulin for keeping the J1/JDM1 alive in the TRS-80 collector's consciousness.

ianm@trs-80.com

1. Unverifiable history is exactly that. Conclusive information from Tandy, Radio Shack, and other sources about the J1 has been very thin on the ground. My timeline and conclusions listed in the first section of this article are made by gathering what scant information I had and making logical leaps between them. Some facts and timeframes are bound to be wrong, and if people in-the-know notice any blaring omissions or errors please contact me so I can put these right in a future article. I know, for example, that at least one person in the USA bought a J1 26-1003A at a Radio Shack Computer Centre. It's therefore possible that a handful of J1s did make it to the USA though officially Radio Shack would have had to deny their existence after December 31, 1980 in order to placate the FCC.

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

OCR Methods

Larry Kraemer

Try as I might, I can't take a PDF of an old manual and get as good results OCRing it as I can by printing it first.

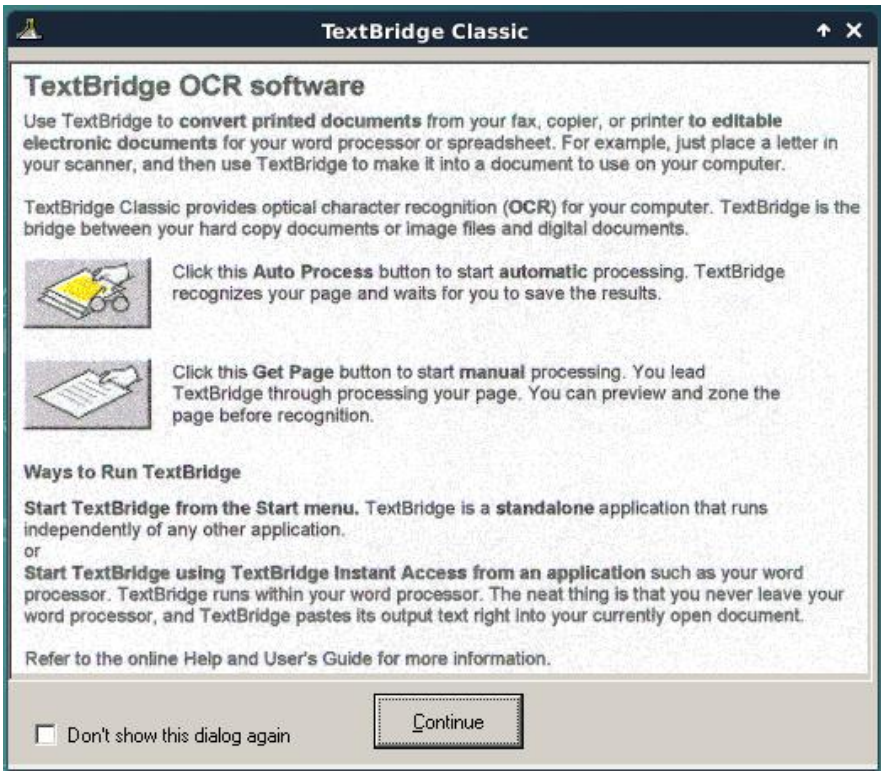
My first method:

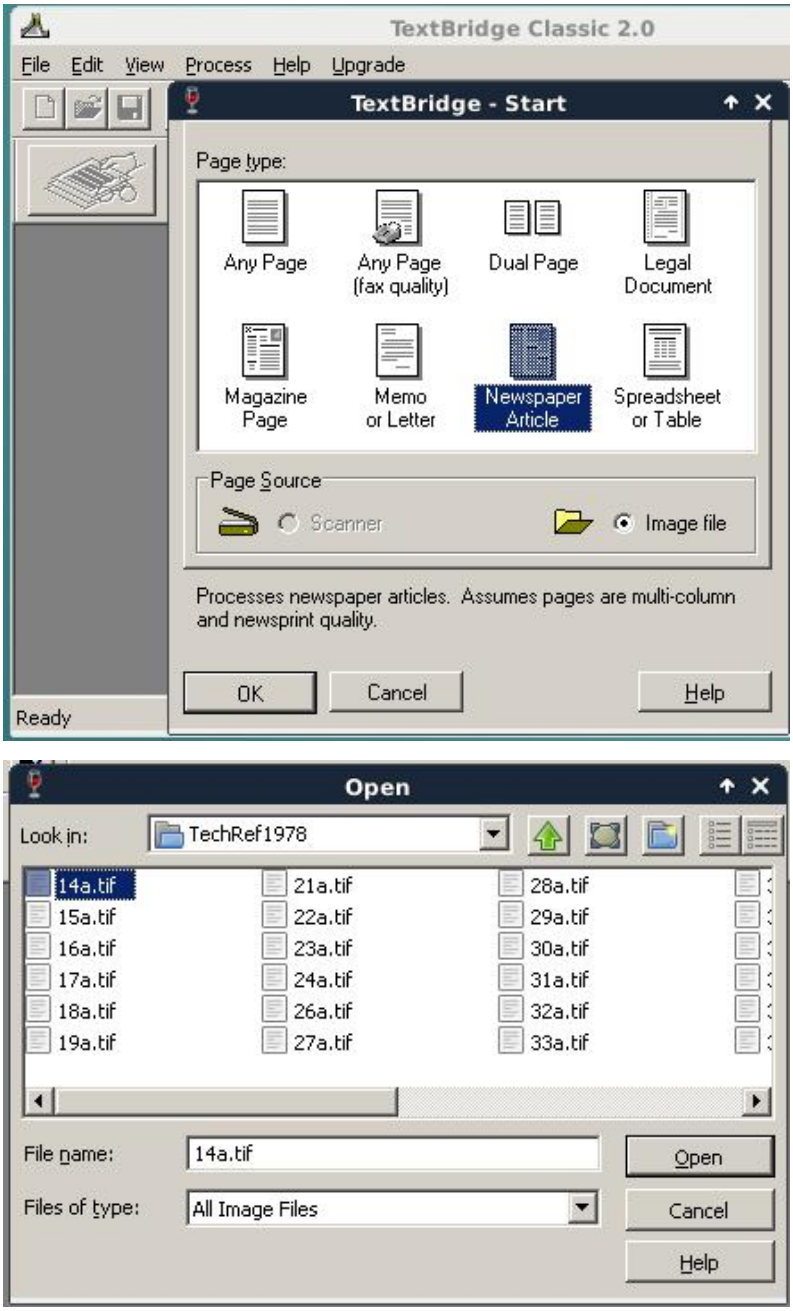
Print all the PDF pages you wish to scan as HIGH QUALITY. Scan all these pages at 600 DPI,
and as a color document, then save as a .tiff file. This creates a file that is
5048 x 7019 pixels, 24 Bit (BPP), 600 DPI.

Open each .TIFF document in rastervect (www.rastervect.com) and do the 1BPP conversion at -30 correction.
The file is now 5048 x 7019 pixels, 1 Bit, 600 DPI.

Save the document as a .TIF, repeat for all other .TIFF files.

Open each TIF in TextBridge Classic 2.0 and OCR the complete Document as a Newspaper article type, save the .txt files. The following PNG's show the TextBridge selections:





Then save the file as 14a.txt.

It should scan at 95+% accuracy. Correct any errors, then reformat as needed.

System Clock

The waveform at pin S of 742 will resemble a sine wave at 10.6445 MHz. The oscillator should not be measured at this point, however, due to the loading effects test equipment would have at this node. 742. pin 6! is the output of the oscillator buffer. Clock measurements may be made at this point, The output of the buffer is applied to three main sections: the CPU timing circuit. the video divider chain, and the video processing circuit,

To attach all the *.txt files into one document do:
cat {14..50}a.txt > Techref1978.txt

The problem with using a PDF is that most PDF's are something less than 300 DPI. But, for Tesseract
and Textbridge Classic 2.0, you need at least 300 DPI, or 600 DPI to get a good conversion.

Convert (an ImageMagick program) will create a 300 DPI TIF file from a lower DPI PDF file that
Tesseract and TextBridge Classic 2.0 can handle.

The command line code is:
convert -density 300 -monochrome 14.pdf 14y.tif
tesseract 14y.tif 14y

But, as you can see it's accuracy isn't as correct as it should be:

9%

System Clock

The System Clock is shown on Sheet 2 of the foid-out Schematics at the hack of this book. YT is a “23.6445 iVin. fundamentaicut crystai. it is in a series resonant circuit consisting of two inverters. 242. pins 3 and 2, and 3 and 4. form two inverting amplifiers. Feedback between the inverters is suppiied by C43. 3 47 pF capacitor. R45 and R52 force the inverters used in the osciilator to operate in their linear region.

The waveform at pin 5 of 242 wiil resemble a Sine wave at 19.6445 MHZ. The osciilator shouid not be measured at this pornt. however. due to the ioading effects test equipment wouid have at this node. Z42, pin 6, is the output of the oscil-lato:r buffer. Clock measurements may be made at this pornt. The output of the buffer is applied to three main sections: the CPU timing circuit, the video divider chain. and the video processmg circuit.

My first method did a better job using TextBridge Classic 2.0.

Irfanview can also be used with the CAD Plugin (OCR_KADMOS) to also make a good OCR'd text document.

For anyone interested in this method of creating a .DOC from a multipage PDF using Irfanview with
OCR_KADMOS.....:

1. Save the multipage PDF to a subdirectory, then burst the PDF into single pages with the following commands:

cd /path/to/subdir
pdftk m1ps.pdf burst

2. Convert the PDF pages to .tif format using the Imagemagick convert program:

convert -density 300 pg_0001.pdf pg_0001.tif
convert -density 300 pg_0002.pdf pg_0002.tif

3. Execute Irfanview (with the CAD plugin installed - OCR_KADMOS) and open the .tif:
Draw box around both sections of text on the page to be OCR'd, extending the bottom of the box about two lines further down.

4. Start the OCR module:
Draw a box around the left half of text to be OCR'd, then save the .txt file, and then repeat for right half, exit OCR Plugin.

5. Combine the two .txt files and clean them up:

The OCR_KADMOS plugin module does about 93% accurate conversion as shown here:

System Clock

The Svstem Clock is shown on Sheet 2 of the
fold-out Schematlcs at the back of this book.
Yl is a 10.6445 MHz. fundamental-cut crvstal.
It is in a series resonant circuit consisting of two
inverters. Z42. pins 1 and 2. and 3 and 4. form
tvvo jnverting amplifiers. Feedback between the
inverters is supplied by C43, a 47 pF capacitor.
R46 and R52 force the inverters used in the
oscillator to operate tn their Iinear region.

The Wave~Orm at pin 5 of Z42 will resemble a sine wave at 10.6445 MHz. The oscillator should not be measured at this point. however. due to the Ioading effects test equipment would have at this node. Z42, p~n 6. is the output of the oscil-Iator buffer. Clock measurements may be made at this po~nt. The output of the birffer is applied to three main sections: the CPU timing cjrcu~t, the video divider chain„ and the video processlng circuit~

Larry Kraemer is an ex-Military USAF Officer, and a retired Electronics & Instrumentation Technician, that has a hobby of Amateur Radio (Advanced Class), and Computers, with Computer Repair experience.

EMAILS TO THE EDITOR

G'DAY DUSTY

I RECENTLY STUMBLED UPON YOUR SITE WHILST LOOKING FOR TRS80 RELATED STUFF.

I HAVE BEEN WORKING WITH COMPUTERS SINCE THE EARLY 1970'S BUT COULDN'T AFFORD ONE UNTIL SOME TIME IN THE 80'S WHEN I ACQUIRED A SECOND HAND MODEL III WITH TWIN FLOPPY DRIVES.

SOMETIME BEFORE THIS I WROTE A LEVEL II BASIC PROGRAM TO TEACH MYSELF MORE ABOUT PROGRAMMING.
NO "HELLO WORLD" PROGRAM FOR ME.

HERE IS THE FIRST PROGRAMME I WROTE.
```
10 CLS:FORI=-32768 TO 32767:PRINT CHR$(PEEK(I));:NEXT
```

THIS WORKS WITH MINOR MODIFICATIONS "SUCH AS CHANGING THE COUNTER STATEMENT TO READ 0 TO 65535" ON MOST COMPUTERS WITH LEVEL II OR EQUIVALENT BASIC THE RESULTS VARY FROM COMPUTER TO COMPUTER, WITH IBM'S AND THEIR CLASS BEING VERY BORING.

MOST EXCITING WAS AMSTRAD CPC464 AND THE COMMODORE 64.
BEEN THROWN OUT OF MANY COMPUTER SHOWROOM BY TYPING THIS FROM MEMORY INTO ALL THEIR COMMODORES AND AMSTRADS FOR FUN:)

ANYHOW THE MODEL III IS LONG GONE BUT I FOUND A COUPLE OF BITS THAT I'M SENDING DOWN TO IAN MAVRIC ALONG WITH A DISK I'M HOPING HE'LL BE ABLE TO GET SOME FILES OFF FOR ME.

I ONLY RUN EMULATORS AT PRESENT DUE TO LACK OF SPACE.

ANYHOW THANK YOU FOR YOUR BIG PART IN KEEPING THE TRS80 SCENE ALIVE.

RICK BUHRE VK4AIM

```
10 CLS: FORI=0TO65535: IFI>32767THENJ=32767-ELSEJ+I:
PRINTCHR$(PEEK(J));: NEXT
```

THANKS FOR THE EMAIL RICK, I HOPE MAV CAN HELP YOU OUT OK.
I LOVE THE BIT ABOUT GETTING THROWN OUT OF COMPUTER STORES. I THINK MANY OF US HAVE BEEN THERE AT ONE TIME OR ANOTHER!

G'DAY DUSTY

I RECENTLY STUMBLED UPON YOUR SITE WHILST LOOKING FOR TRS80 RELATED STUFF.
I HAVE BEEN WORKING WITH COMPUTERS SINCE THE EARLY 1970'S BUT COULDN'T AFFORD ONE UNTIL SOME TIME IN THE 80'S WHEN I ACQUIRED A SECOND HAND MODEL III WITH TWIN FLOPPY DRIVES.
SOMETIME BEFORE THIS I WROTE A LEVEL II BASIC PROGRAM TO TEACH MYSELF MORE ABOUT PROGRAMMING. NO "HELLO WORLD" PROGRAM FOR ME.
HERE IS THE FIRST PROGRAMME I WROTE.

```
10 CLS:FORI=-32768 TO 32767:PRINT CHR$(PEEK(I));:NEXT
```

THIS WORKS WITH MINOR MODIFICATIONS "SUCH AS CHANGING THE COUNTER STATEMENT TO READ 0 TO 65535" ON MOST COMPUTERS WITH LEVEL II OR EQUIVALENT BASIC THE RESULTS VARY FROM COMPUTER TO COMPUTER WITH IBM'S AND THEIR CLASS BEING VERY BORING.
MOST EXCITING WAS AMSTRAD CPC464 AND COMMODORE 64.
I'VE BEEN THROWN OUT OF MANY COMPUTER SHOWROOM BY TYPING THIS FROM MEMORY INTO ALL THEIR COMMODORES AND AMSTRADS FOR FUN.

ANYHOW THE MODEL III IS LONG GONE BUT I FOUND A COUPLE OF BITS THAT I'M SENDING DOWN TO IAN MAVRIC ALONG WITH A DISK I'M HOPING HE'LL BE ABLE TO GET SOME FILES OFF FOR ME.

I ONLY RUN EMULATORS AT PRESENT DUE TO LACK OF SPACE.

ANYHOW THANK YOU FOR YOUR BIG PART IN KEEPING THE TRS80 SCENE ALIVE.

RICK BUHRE VK4AIM

G'DAY AGAIN DUSTY
OK JUST A BIT. NO WAY WOULD I SEND IN A PROGRAMME OF ANY SORT IF IT WERE NOT INTENDED FOR PUBLICATION SO GO AHEAD.
I'M USING SEVERAL EMULATORS.

1. DAVID KEIL'S TRS-80 MODEL I/III/IV EMULATOR V6.2 ORIGINALLY ON MS-DOS 6.22 BUT NOW IN A WINDOWS & VIRTUAL MACHINE WITH DR-DOS 7.32
2. ARNO PUDER'S DROIDTRS ON MY MOBILE PHONE THE ULTIMATE PORTABLE TRS80 STILL IN ALPHA. CAN'T GET MI TO GO BUT MIII WORKS OK
3. MATTHEW REEDS LATEST WHICH I JUST PURCHASED. SEEMS TO BE GOOD.

BYE RICK

G'DAY DUSTY
I REALISED THAT WHILE THE ONE LINER I SENT YOU WORKED FLAWLESSLY,WHEN RUNNING ON A TRS80 COMPUTER IT APPEARS TO BE DOING NOTHING FOR QUITE A WHILE. IT ISN'T, IT IS DOING QUIT A BIT, BUT IT ISN'T SEEN.
SO TO CORRECT THIS I'VE REWRITTEN IT SO THAT IT STARTS AT THE INTERESTING BITS FIRST.

HERE GOES.

```
10 CLS:FORI=0TO65535:IFI>32767THENJ=32767ELSEJ=I:PRINTCHR$(PEEK(J));:NEXT
```

Mav's Quarterly

- Updates on the state of the Biz, in no particular order -

by Ian Mavric

Undelivered packages: Yes, it happens occasionally and while I do my best to send things in a timely manner (usually 2-5 days after receiving payment - if it's much longer then I will email to advise you), on the average I get one Return-To-Sender and one Lost In The Mail every six months. Without pointing a finger at any individuals in particular, an alarming trend in post offices not even bothering to try to deliver parcels (not just mine, but any parcels) results in them sitting at the local post office, the post office not telling the receiver the item is waiting for them, then sending it back to me after 10-14 days. PEOPLE, if you know in your locale that the post office won't even attempt to deliver to you, then you MUST pay the extra for a tracking number service in addition to postage. I then send you the tracking number and you can follow it up with your post office, DAILY if need be, to get your item. I can't be held responsible for YOUR post office not alerting you that something has arrived and sending it back to me.

Untracked packages: I remember a time, not too long ago, when I could pack and send anything, anywhere in the world, with the plain Air Mail service out of Australia, very confident that the item would be delivered in a timely manner. Admittedly this was during the pre-internet pre-online shopping era, but no one needed a tracking number and unless I got an address wrong, everything would arrive. This is still the case to many places and I confidently send things without tracking numbers to the USA, Canada, Australia, New Zealand. I used to also include the UK and France until some untracked packages started to come back (see Undelivered Packages, above). The choice is yours as to whether you know you live in a safe enough area with reliable enough postal service or not, and choose appropriate shipping with or without tracking.

Australia Post rates hike: Prices have gone up again, so now a 500g parcel to the USA or Canada costs $15.85 (up from $14.10) and to Europe costs $21.50 (up from $18.90). 501g-1kg package to the USA and Canada costs $33.40 (up from $28.90) and to Europe costs $36.95 (up from $33.10). Similar prices increases to everywhere else in the world have been implemented, so if you are buying the same thing from me this year as last year, that's why the shipping cost that I need to pass on to you use risen. As always, I don't profit on the shipping cost of my items.. what it costs me to ship is how much I charge you.

My Web Site: Some have asked me how often I update it? The answer is, whenever I feel like it :-) In a perfect world I'd update it weekly or whenever I have something new that is ready to sell, however like most of us I am unfortunately time-poor. Just keeping up with orders, repairs, system restorations, writing these articles and now the Trash Talk pod-casts, I'm very busy. Not that I don't like it this way, but something has to give and that means my web site gets an update approximately every 3-4 months, or whenever I have a new product to bring to the market-place, whichever is first.

Rare parts not on my web site: If you are looking for something that is not on my web site, then ask. I can't put everything I have on hand on the web site, and I might have just the part you need to complete that restoration or upgrade you have been putting off.

Clearly Superior FreHD: In the last few months I've sneaked the FreHD Clearly Superior series onto the marketplace for people who want a com-plete product ready to plug in and use the moment it arrives. These come mounted in a small neat enclosure (which I affectionately call BoHx) with rubber feed, and all the required cables and power adapter or internal connection power cables. No messing around with self supplied 5VDC sources, and bare exposed circuit boards on your desk. Auto-boot Eproms are also supplied for those who wish to open their computer to install one, or for those who don't, I include the traditional hard disk boot floppy. The Clearly Superior FreHD covers the widest range of system configurations, and now can even be enjoyed by the least technical indi-vidual. Of course FreHD is still available in its previous "Kit C" form, as well as the ever popular "Kit B" major components starter kit, and the "CKD" kit for those who like all the parts to arrive ready to solder to-gether.

Coming in April(?) - Model III HRG: Ok so not all things go to plan and sticking with the time-poor excuse, above, this one is taking a little longer to get to market. Hopefully it will be here by September.

The Right Stuff: One stop TRS-80 shop for restored systems, hardware upgrades, spare parts, books, magazines, software, cables and advice.

Ian Mavric
ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

Time to teach an old dog some new tricks

John Mineur

If you are a fan of TRS-80 machines like myself, you have probably started thinking about moving away from traditional floppy storage due to spiralling costs and ever increasing reliability issues.

One possible solution to this problem is to use the much written about FreHD unit designed by Frederic Vencoven. That being said, the FreHD although perfect in functionality, definitely lacks a certain desktop charm to many enthusiasts like myself.

My challenge was simple then, to house the FreHD unit in an appropriate container but where to start? Having looked through my collection for inspiration I finally decided to repurpose my old TRS-80 Voice Synthesizer, which in my opinion ticked all the boxes.
Furthermore, I would be able to add a small amplifier module and utilise the speaker mounted within the enclosure for the all-important recreational software applications (aka games).

Most of the components were purchased straight from eBay:

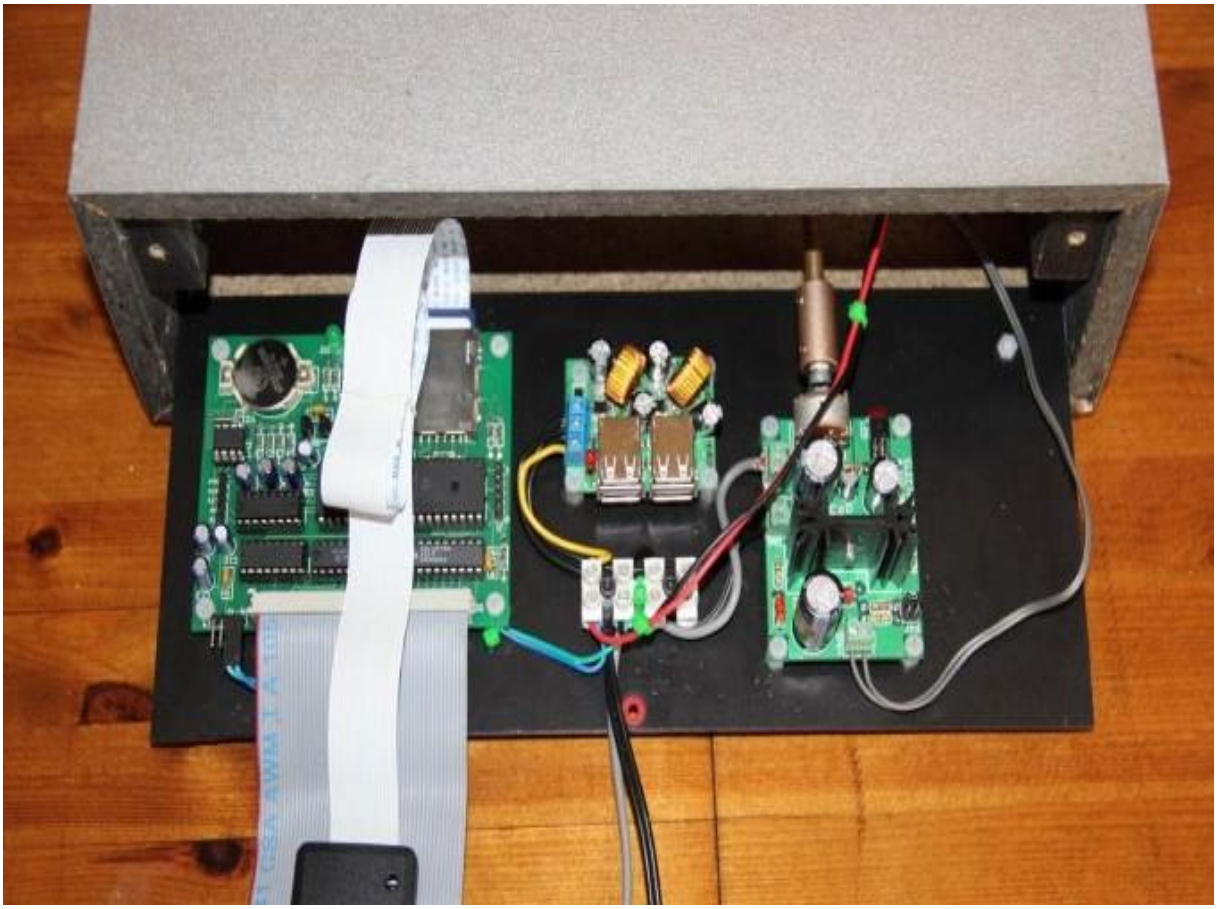
Nylon pillar mounts
PVC foam board (3mm)
10w mono amplifier module
12v to 5v regulator module
SD extension socket
Potentiometer extension bar (6mm)
12v (1.5a) power supply
Plugs, sockets and cable
5v LED
Aluminium potentiometer knob (6mm)
Power strip

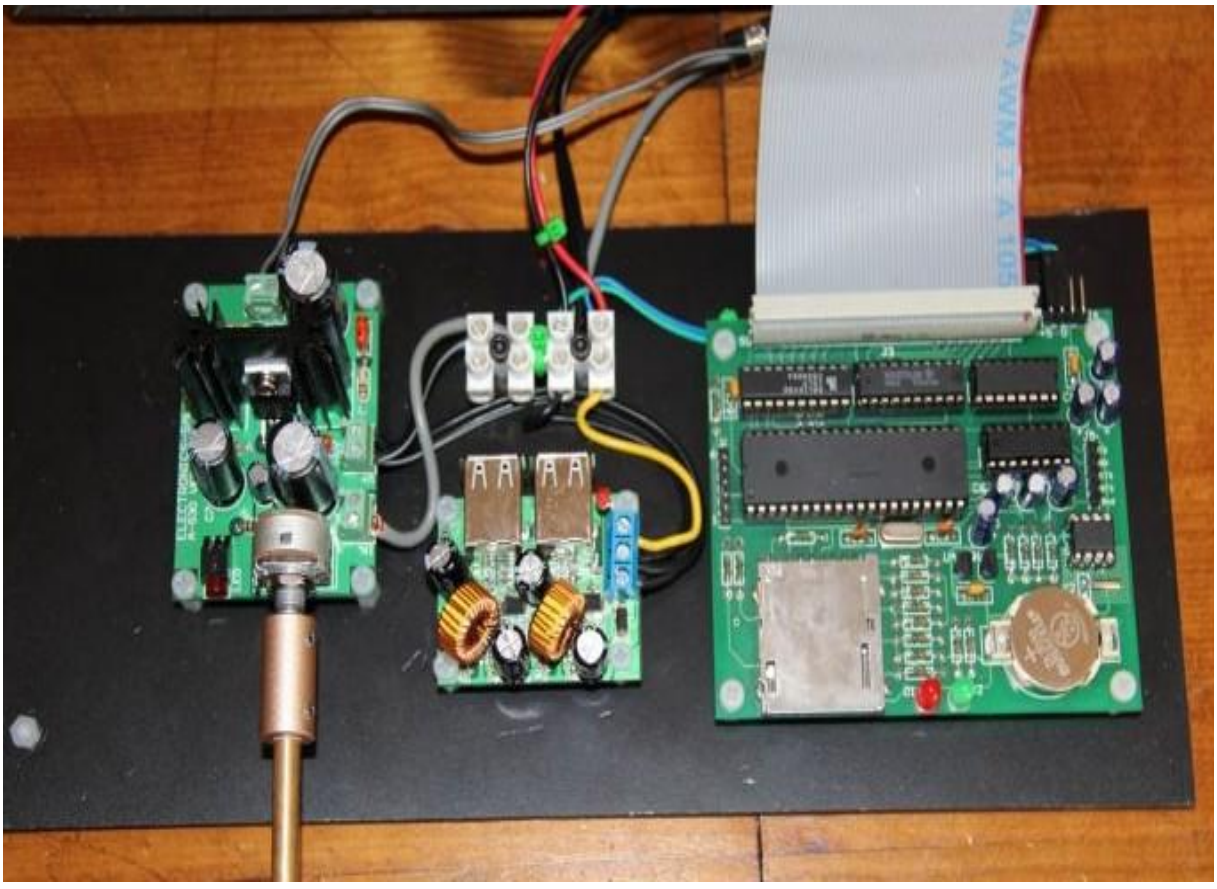
When it came to the FreHD unit I had to go to my old friend Bas Gialopsos who also threw in a 50-way cable for a very reasonable price and countless free cups of coffee.

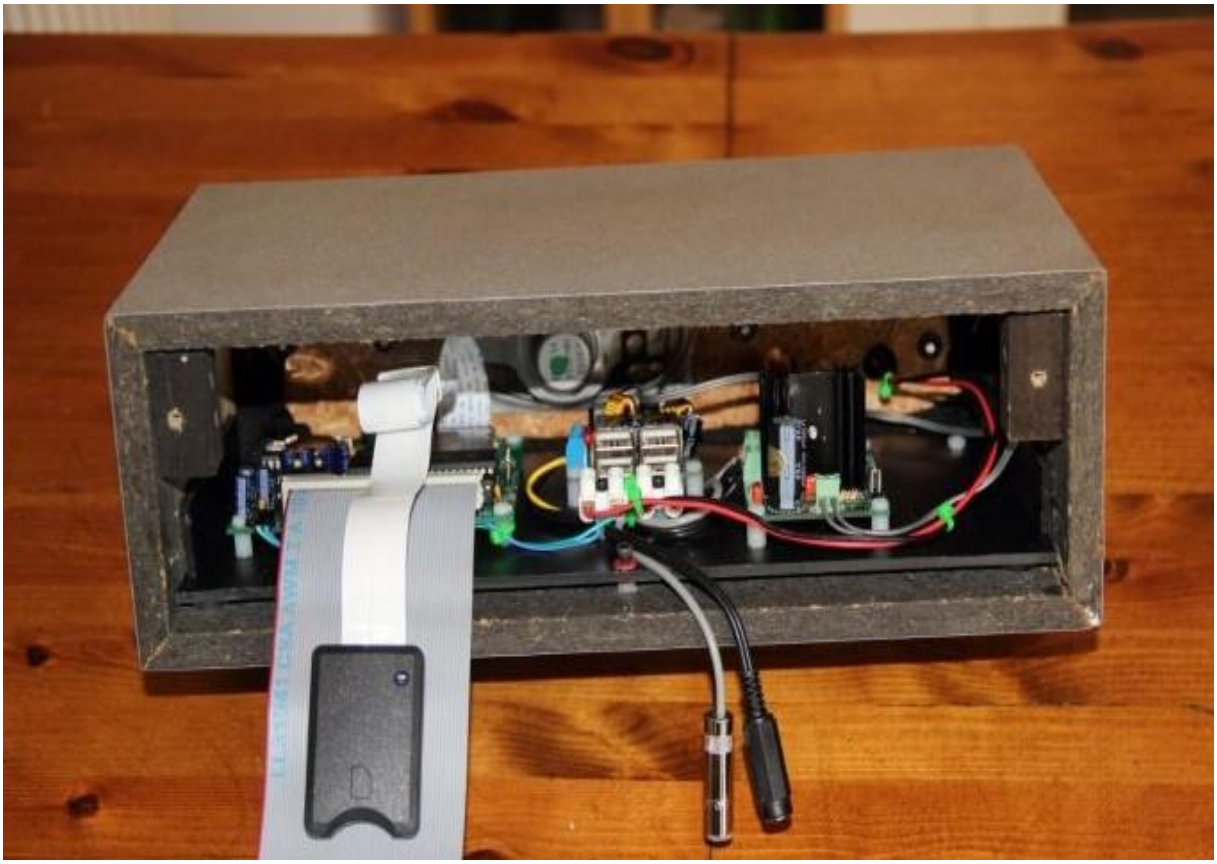
PVC foam board was used to mount the components due to its lightness and the ability to be cut with a craft knife.

An hour or so later the transformation had taken place and I was now the new owner of a genuine TRS-80 Audio & Data Centre. Looking at the final product I know that it won't be everybody's cup of tea but it certainly brings a big smile to my face.


RADIO SHACK VOICE SYNTHESIZER
Part No. 26-1180
Serial No. 212039MGJ







TRS8BIT'S 2016
COMPETITION

I THOUGHT THE 2016 COMPETITION SHOULD BE SOMETHING QUICK AND EASY(ISH), WITH AN ORIGINAL AND UNIQUE PRIZE.
(ACTUALLY, IT'S 4 PRIZES!)

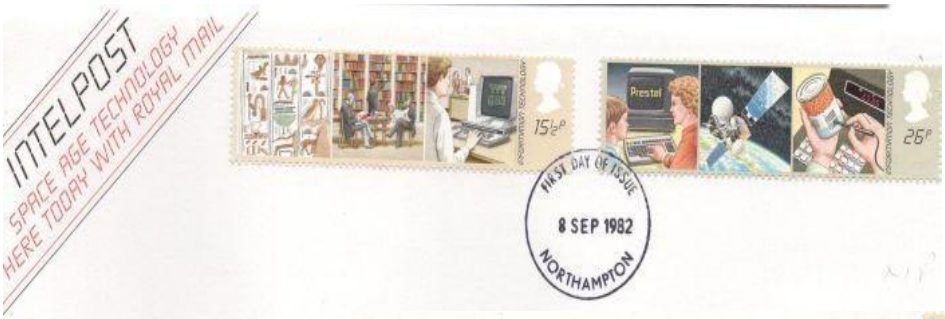
SO, ALL YOU HAVE TO DO IS EMAIL ME, (DUSTYM @ FABSITESUK.COM), A ONE-LINE BASIC PROGRAM, ON ANY SUBJECT YOU FANCY. THE PROGRAM MUST ONLY BE ONE LINE LONG. YOU CAN EXTEND THE LINE LENGTH TO THE MAXIMUM PERMISSIBLE BY USING THE TANDY EDITOR.
ALSO, THE PROGRAM MUST BE SELF-CONTAINED WITHIN THE LINE; I.E. NO PRE-LOADING OF CODING INTO MEMORY BEFORE RUNNING.

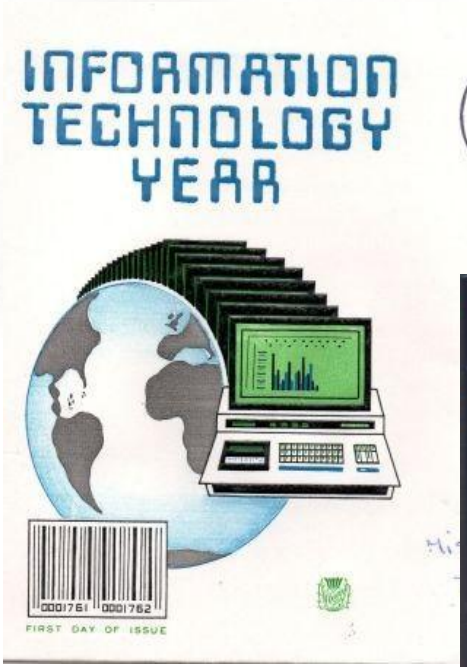
DEE WILL AWARD THE PRIZE TO THE PERSON(S) WHO, IN HER OPINION, SEND IN THE MOST ORIGINAL/AMUSING PROGRAM

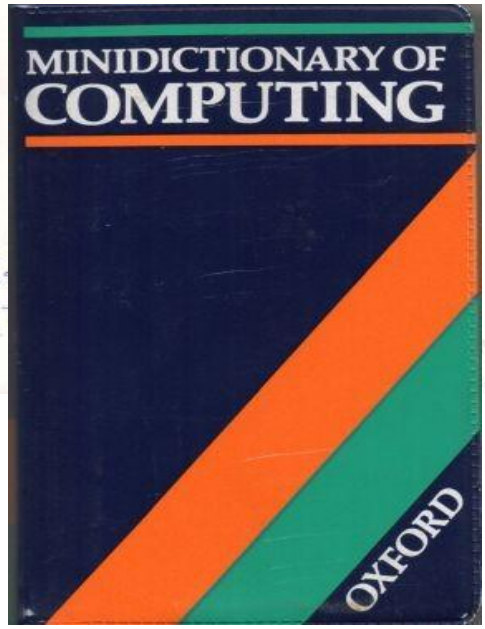
AND THE PRIZE :- AN ORIGINAL 1982 ROYAL MAIL ISSUED, SET OF POSTAGE STAMPS (SEE OVER) CELEBRATING 'INFORMATION TECHNOLOGY YEAR', COMPRISING OF BOTH THE 1ST DAY ISSUE COVERS, AND A SET OF 'MINT' STAMPS. I'VE ALSO INCLUDED AN OXFORD MINI COMPUTING DICTIONARY FROM THE SAME ERA.

CLOSING DATE FOR ENTRIES - 1ST DECEMBER 2016, 21:00 HRS GMT.

THE 2016 COMPETITION PRIZE







LOOKING FOR FAST, INEXPENSIVE, UNLIMITED MASS STORAGE FOR YOUR TRS-80 MODEL I/III/4/4P/4D?

The amazing

"FreHD"

-Emulates a TRS-80 hard drive, but faster than any hard drive!
-Works with your favourite DOS (LS-DOS, LDOS, CP/M, Newdos/80 2.5)
-Uses SD card for storage medium
-Bonus free Real Time Clock function!
-Designed in Belgium and proudly built and shipped from Australia
-Kit form or fully assembled

Order yours today:
http://ianmav.customer.netspace.net.au/trs80/emulator

The Ultimate Copying Machine

Hans Rietveld

Always Having trouble with copying disks I built an ultimate copy machine.

It consists of an old MSDOS computer with an opti green motherboard with an ISA bus and award bios, a 486-DX2-66 processor and a 720 and a 1.44 floppy drive.

It is very important to use the Adaptec 1542CP controller to connect the floppies.
It is the only controller which can handle single and double density data.
It is not possible to connect a SCSI hard disk to the Adaptec as the award bios doesn't scan the memory blocs in the right way.

Any trick to solve this problem didn't help.
The only way is to use a multi I/O adapter to connect an IDE hard disk. I installed MSDOS 6.22 and later Windows 98SE too, to support some extra tools.

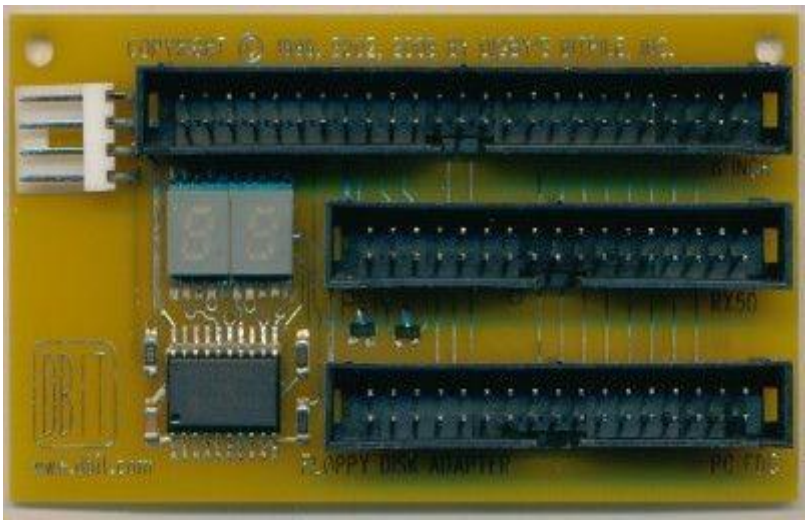
The Adaptec controller is connected with a flat cable to the option board and passes trough the board to both of the floppy disks.
I brought this cable outside the computer and split the cable in two parts with a 34 pin male and female connector.
In the BIOS I can swap the A: and B: drive in case of a
floppy boot with different type of floppy.

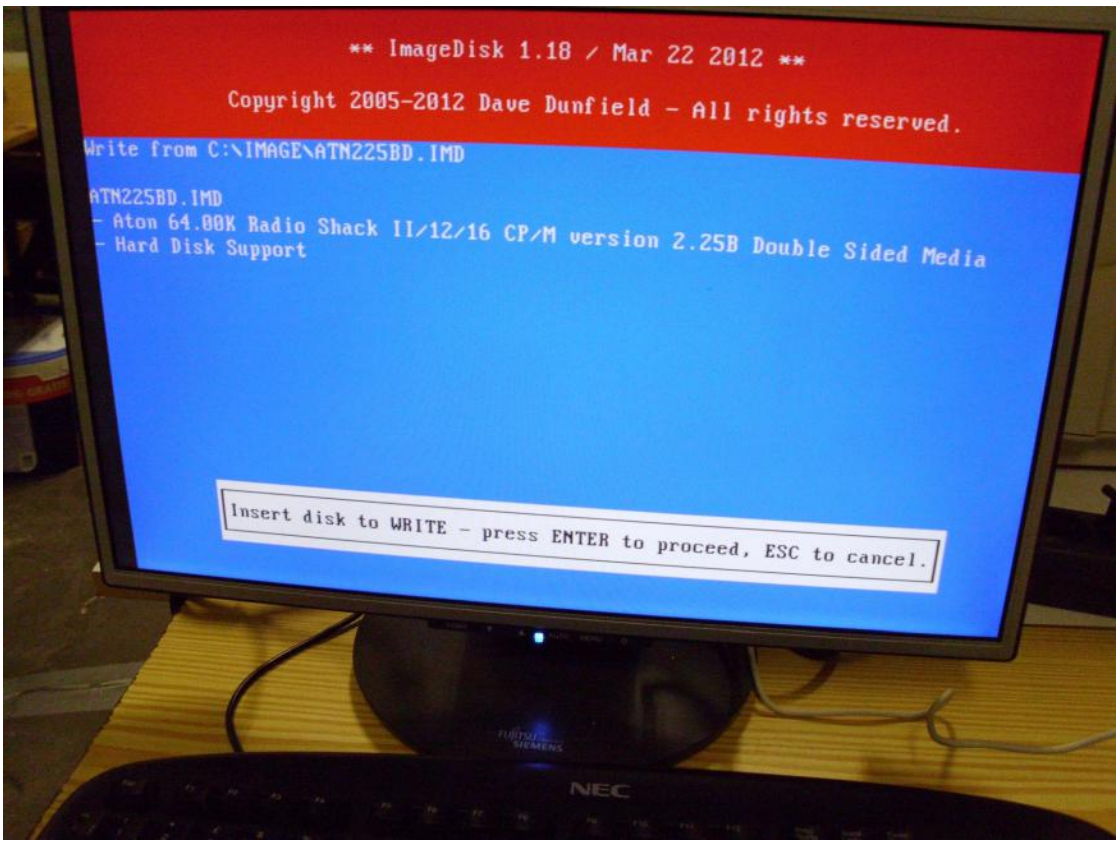
If I connect the two parts together I can use both floppys.
If I open the connectors I can connect to one part of the cable to one or two 5.25 drives or my new FDADAP adapter.
(This is really a great tool to painlessly support, 8" drives. It is a really must have device for Model II users)

With this machine I can Read, Write and Edit all Tandy 720, 1.44, 5.25 and 8" SS and DS disks in Single and Double density.
I can backup all my disks to my hard drive now and rebuild them in a moment.

The software I use is COPYIIPC, TELEDISK, IMD (Image) and many other tools.

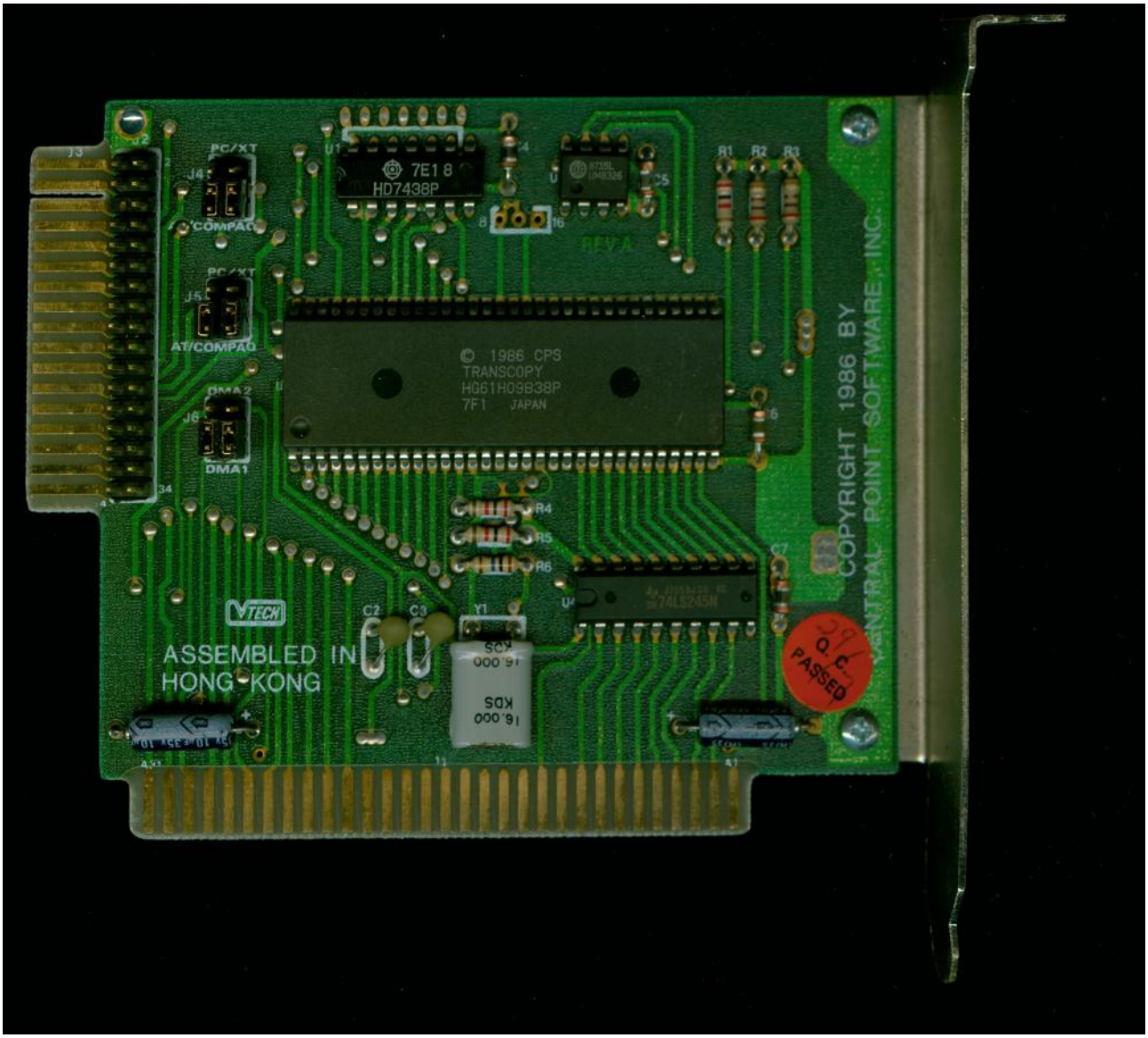
It is a lot of work to build this machine but it is worth the costs and the time a spent for it.

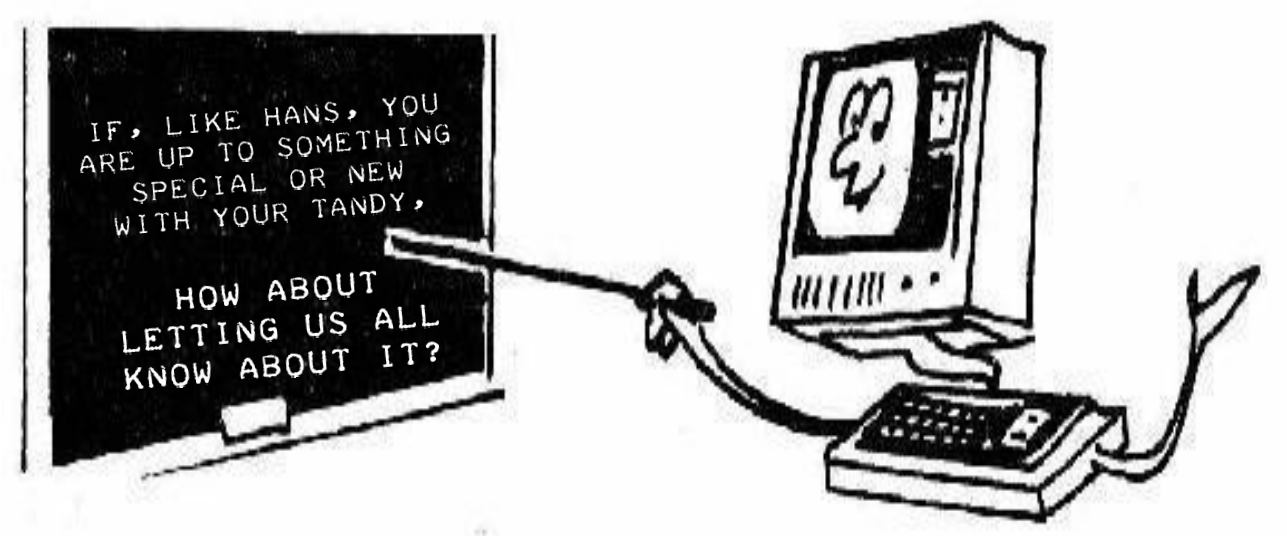



** ImageDisk 1.18 / Mar 22 2012 **
Copyright 2005-2012 Dave Dunfield - All rights reserved.
Write from C:\IMAGE\ATN225BD.IMD
ATN225BD.IMD
- Aton 64.00K Radio Shack II/12/16 CP/M version 2.25B Double Sided Media
- Hard Disk Support
Insert disk to WRITE - press ENTER to proceed, ESC to cancel.


B: 1.44M
A: 720K
HI
RESET
TURBO


HI


PC/XT
AT/COMPAQ
DMA2
DMA1
7E18
HD7438BP
1986 CPS
TRANSCOPY
HG61H09B38P
7F1 JAPAN
REV A
COPYRIGHT 1986 BY
CENTRAL POINT SOFTWARE, INC.
VTECH
ASSEMBLED IN
HONG-KONG
16.000
KDS
Q.C.
PASSED


IF, LIKE HANS, YOU ARE UP TO SOMETHING SPECIAL OR NEW WITH YOUR TANDY,
HOW ABOUT LETTING US ALL KNOW ABOUT IT?

TRS-80 Emulators . com

TRS32: A Model I/III/4/4P Emulator For Windows

written by Matthew Reed

Unregistered Shareware Version:

- Works under all current versions of Windows
- Full Windows application — no low-level hardware conflicts!
- Model I, Model III, Model 4, and Model 4P emulation
- Four floppy disk drives (with optional realistic disk drive sound)
- Cassette tape drive with graphical on-screen controls
- Exatron Stringy Floppy emulation
- Printer support
- Serial port for RS-232 communications
- Joystick support (using a Windows joystick — TRISSTICK and Alpha Products joysticks are emulated)

Registered Version:

- All features included in the shareware version
- Built-in emulation of an Epson FX-80 dot matrix printer (including graphics and control codes)
- High resolution graphics (Radio Shack and Micro-Labs)
- Up to 1 megabyte of additional memory in Model 4 and 4P modes
- Hard disk support
- Orchestra 85/90 music generation

Interested?

- Read the TRS32 emulator documentation
- Download the shareware version
- Register online

Actual Dotwriter output — emulated FX-80
click for larger sample

MATTHEW'S EMULATOR IS QUITE ASTOUNDING.
IT'S MY FIRST CHOICE WHEN DEVELOPING IDEAS FOR TRS8BIT.
HE OFFERS FULL SUPPORT AND IT IS STILL UNDERGOING ENHANCEMENT.
IF YOU HAVEN'T REGISTERED YOUR SHAREWARE VERSION YET, PLEASE DO SO AND ENCOURAGE
AND SUPPORT MATTHEW'S ENTHUSIASM TO CONTINUE WITH HIS WORK.

ED.

ROMDOS/80 explained

by Garry Howarth (Gazza)

(ex SYDTRUG member)

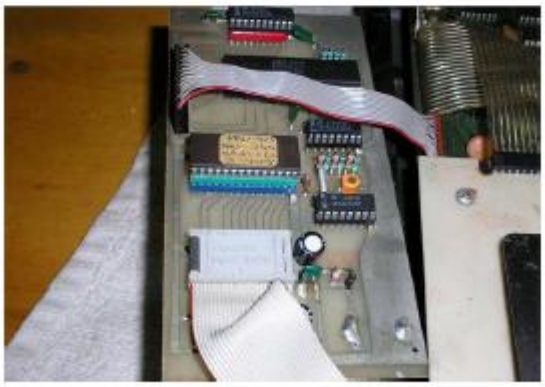
All of this was done nearly a 1/4 of a century ago, some of my notes, source code, diagrams etc have been lost, some of it survives, so a lot of what follows is from my fading memory and may not be 100% correct.

Firstly, what is ROMDOS/80 and what does it do? ROMDOS is Newdos/80 stored in a 27512 EPROM. This allows the computer to boot from either a system or data disk. This frees-up drive 0 for more important things, like playing games. We now have a System 80 with 5 drives The only problem is that this extra drive only has 64k of memory and the data can't be changed without a UV light and an EPROM burner. The joys of ancient technology.

All requests for system files are redirected to the ROMDOS driver which then transfers the system file from the EPROM to it's correct place in the dos overlay area. ROMDOS also includes an RTC based on a National 58274

Version 1 of the ROMDOS/80 board was started in early 1988. This version included rom based Newdos/80, a real time clock calendar with battery backup and a CPU turbo, which, if my notes are accurate was designed by Stephan Keller and Chris Tinney of Sydtrug. Port addressing was done with dip switches.

In mid 1988 I designed version 2 of the ROMDOS/80 board. A simplified version of the previous board. No CPU turbo and fixed port addressing. The CPU turbo was built on it's own board mounted at the front of the computer.



Picture 1. CPU turbo board

Both of the earlier versions still required booting from a system disk and the driver was loaded into HIMEM with the AUTO command. The driver then took control and all system files were then loaded from the EPROM. The computer could now be run without a system disk. The driver was self-relocating, self-protecting and in the first version would search through a range of port addresses until it found the board. If the board was disconnected the computer would operate normally.

The final version, V3, appeared in early 1989. This board is based on an 8255 PPI. This final version includes a fully patched version of Newdos/80 with mods written by Alan Johnstone, some mods by Jack Decker and possibly others, an RTC and drivers for 8 additional function keys (16 with the shift key) and what seems to be a programme to send control codes to the printer. Still trying to work that one out. With ROMDOS V.3 the computer can now be booted from a data disk.

Picture 2. ROMDOS board

This is the ROMDOS board after I cleaned off 15 years worth of dust, dirt and dead spiders.
The 8255 divides the 27512 into 32 * 2k blocks, mapped at 3000H. Block '0' contains the drivers and the remaining 2K blocks contain the system files. The 8255 also controls the 58274 RTC.

ROM 'A' on the CPU board also has to be patched. Problem, where to put the patches. After the 'Ready' message there is a series of carriage returns. By removing these carriage returns I had just enough space for my patches.

After nearly 25 years I was having trouble remembering how the System 80's roms worked until I found a copy of TRSDOS 2.3 Decoded. This book has a good description of how the boot sector is loaded. After reading this chapter, all is now clear - I have seen the light.

After loading the boot sector the computer would normally jump to 4200H and continue loading DOS from floppy. Not any more. It now jumps to my patch in ROM 'A'

On reset the rom will load the boot sector, it then attempts to switch the EPROM to bank '0' and checks to see if it is connected, if not, it will continue to boot from floppy. It then does a few other checks. If the disk is not a Newdos80 disk or if the space bar is held down it will boot from floppy as per normal.

Having ticked all the boxes it jumps to 3000H, transfers the driver from the EPROM to himem, passes control to the driver, switches the EPROM to block '1' and transfers SYS0/SYS from the EPROM to ram.

Because SYS0/SYS is larger than 2k the EPROM is switched to the next block and loading is continued until the complete file is transferred. The same applies to SYS6/SYS. All other system files can be transferred in one go.

The start of each block holds the number of bytes to be moved and the destination address in ram. These are loaded into the BC and DE register pairs and the LDIR command is used to transfer the data from the EPROM to ram.

When SYS0/SYS has finished loading, the EPROM is reset to block '0' and the time and date are read from the 58274 and stored in the proper DD/MM/YY format. Control is then passed to SYS0/SYS and the remainder of Dos is loaded from the EPROM.

The system file loader has been patched and all requests for system files are now sent to the driver. On entry the HL register holds the required system file number and using this the driver calculates the correct block number, switches the EPROM to that block and transfers the required system file to the overlay area in ram. It then resets the EPROM to block '0' and passes control back to Dos

The computer will still use the System and Pdrive parameters on the data disk. So they can still be set in the usual way.

Other Programmes

The file ROMDOS/CMD can be loaded via the AUTO command. This programme will exit to Dos with an error message if the ROMDOS driver has not been loaded.

Booting with the Space Bar held down will force a boot from floppy and will not load the driver.

If all is OK, the programme will then tie the function keys and clock onto the interrupt chain. Any programme or process that switches off the interrupts can cause the clock to lose time. This programme should allow Dos to maintain perfect time.

The ROMDOS board also includes a programme to send control codes and text to the printer. A poor man's word processor. This one had me running around in ever decreasing circles.

I could see how it worked but couldn't nut out how to access it. What's the point of having it sitting in the EPROM if I can't get to it?

After examining the code I found a call to 4461H. The Newdos manual calls this the *name routine enqueue. and after more looking through memory I found the word –PRN

I'd completely forgotten about the Newdos *name routine enqueue.

I typed in *PRN and up popped the printer menu - as easy as. Gave myself an uppercut.

*PRN is now a valid Dos command that can be used from the command line, Mini-Dos, Basic or within any programme that allows Dos commands.

After loading, ROMDOS/CMD requires no user memory. It uses routines stored in the EPROM and memory that was used by the now redundant system file loader.

CLK/CMD or CLK/BAS are used to set the time and date of the RTC.

Both are quick and dirty but they do the job.

CLK/BAS MUST be run with the ROMDOS drivers disabled. This programme takes full control of the 8255 and if Basic needs to load a system file from the EPROM - and it will - it wont be able to find it, resulting in a lock-up.

ROMDOS/80 was just an exercise to teach myself electronic design, construction and programming. Many, many hours were spent with my head buried in books and source code trying to nut out the System 80's roms and the inner working of Newdos. Most of it now forgotten.

Way back then we didn't have the use of PCB design programmes with auto routing, so the double sided PCB was initially laid out on grid paper and the final artwork was done with Bishop Graphics self-adhesive tapes and pads on transparent plastic. Exposed under UV light and etched with ferric chloride. The hard part was keeping both sides perfectly aligned but in the end it all worked out OK.

Picture 3. ROMDOS/80 at boot

Over 20 years later and I'm still trying to get the ferric chloride stains off my fingers.

To day, with compact flash, USB memory, floppy emulators etc. the old ROMDOS is looking like a real dinosaur, but it did what was asked of it and over 20 years later is still working perfectly. It even has it's original battery.

and that as they say is that - ROMDOS Explained - or the basics of it.

What's next?

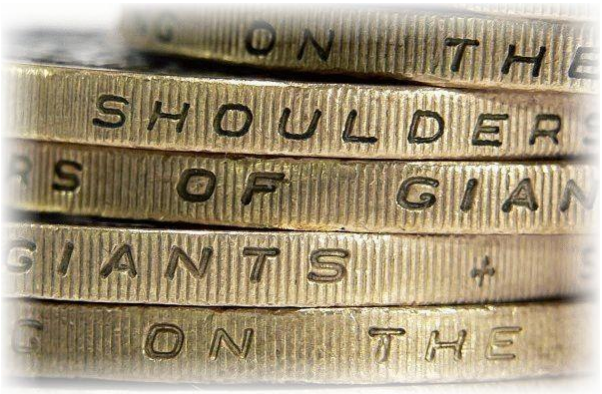
Unfortunately it takes up far too much bench space, so it looks like it'll be stored away for another 15 years, when I'll have to go through this whole process again.

I should also give a thank you to Glen McDiarmid, even though he doesn't know it, his excellent programme Macaamon was a huge help in getting this project going.

and...just when I thought the old beast was running perfectly, it started suffering from random reboots. (sigh!). Found the problem though. A faulty socket!

It never ends.

THIS ARTICLE FIRST APPEARED IN AND IS REPRINTED FROM SYDTRUG NEWS C/- PETER WIGNELL PO BOX 95 NARWEE NSW 2209 AUSTRALIA



USING NEWTON'S QUOTE "STANDING ON THE SHOULDERS OF GIANTS" AS INSPIRATION, I OFFER ANOTHER IN THE SERIES OF SOME OF THE KNOWN, AND LESSER-KNOWN NAMES WHO INSPIRED MUCH OF WHAT WE, TODAY, TAKE FOR GRANTED.

Grace Hopper

Grace Hopper was largely responsible for the development of high level languages, and identifying the first bug!

COURTESY OF SPERRY LTD

Computer science is generally regarded as a strictly male preserve. But, increasingly, women are taking their place alongside men, as equals, in the development and application of computers. A woman pioneer of computing was Grace Hopper, whose most significant contributions were in the field of software — she created the first compiler and helped invent the language COBOL. But she was also the first person to isolate a 'bug' in a computer, and successfully 'de-bug' it.

After doing postgraduate work at Yale, Grace Hopper returned to her original university, Vassar, as a member of the mathematics faculty. Here she remained until the age of 39, when she was called up for war service with the Naval Ordinance Computation Project. In 1945, she was ordered to go to Harvard University to assist a physicist, Howard Aiken, in the building of a computer. Aiken had approached IBM in 1937 with the idea of constructing a computer using adapted tabulating equipment. His first computer, although mechanical in design, was successful enough to encourage IBM to invest in an improved model that would use electromechanical relays. The machine that was subsequently developed was known as the Harvard Mark II.

In these early days, machines had to be programmed by rewiring them for each new task. Thus, in the hot summer of 1945, Grace Hopper found herself literally enmeshed in the wiring of the computer. Ballistic computing facilities were urgently needed for the war effort, and Commander Aiken would often come into the workshop and demand: 'Why aren't you making numbers, Hopper?' After one troublesome breakdown of the computer, when the fault was eventually found to be a moth that had flown in through the open windows and been hammered to death in a relay switch, Grace tersely replied: 'We are debugging the machine!' This first recorded 'bug' was carefully removed from the relay with a pair of tweezers and is preserved at the Naval Museum in Virginia in the log book of the Harvard Mark II. It is glued beside the entry for 15.45 on 9 September 1945.

In the same year another computer, ENIAC , was being built by two engineers, John Mauchly and Presper Eckert. After the war, the two men set up their own business to manufacture a commercial version of the machine, and invited Grace to join their team. Her main contribution to the development of this computer, called UNIVAC (UNIVersal ACcounting machine), was in creating software for it. And it was during her attempts to construct programs for business use on UNIVAC that Grace first sought out ways to short-cut the need for rewriting certain subroutines that recurred over and over again. By employing what was then considered the remarkable idea that a computer could write its *own* programs, Grace created the first programming language, together with the compiler needed to translate it into machine code. This was given the name 'A–O'. When this compiler was first presented it caused incredulity amongst computer professionals who thought their machines could only perform arithmetic or manipulate symbols. They were amazed to see a computer jump to a subroutine in its library store on encountering an imperative verb at the beginning of what was almost a normal English sentence.

In May 1959, Captain Hopper was invited by the Pentagon to join a working committee that was to attempt to create and standardise a single language for computers in commercial use. In less than a year the committee produced the first version of the COmmon Business Oriented Language (COBOL). Grace contributed a great deal to the committee's attempt to distil the best aspects of each of the existing languages and thus create a language acceptable to the industry through its sheer quality. It is a measure of the success of the committee's work that COBOL is still one of the most widely used languages today.

COBOL

COBOL was one of the first programming languages to be written with the intention of making it easily accessible to non-mathematicians. The language encourages the use of generalised procedures written in a narrative style of English, rather than coded routines peculiar to a particular problem.

A program in COBOL is built out of four units. The name of the program, its author, and other reference information comprise the Identification division. Although COBOL programs are intended to be portable (capable of being used on many machines), any details describing the particular computer for which the program was originally written are noted in the Environment division. Since the same data may be used in many parts of the same program, COBOL has a separate Data division. Finally, the procedures that are to operate on the data are listed in the Procedure division

Surfing the Internet… from my TRS-80 Model 100

Stepping into the 21st century with the mother of all laptops

by Sean Gallagher

This article first appeared at

http://arstechnica.com/information-technology/2015/08/surfing-the-internet-from-my-trs-80-model-100/

Mobility redefined: TRS-80 Model 100, Raspberry Pi B, a serial cable, a USB-to-serial converter, and a swag backup battery to power the Pi. Onscreen, I'm on Google's home page with Lynx.

Sean Gallagher

The true test of a man's patience is crimping pins onto the end of a cable that leads to building a custom serial cable—especially if it's the first time you've even handled a serial cable in a decade. So as I searched under my desk, using my phone for a flashlight, I wondered whether I had finally found the IT project that would send me over the edge. On a recent day, I set out to turn my recently acquired vintage Radio Shack TRS-80 Model 100 computer into a working Internet terminal. And at this moment, I crawled on the floor looking for a DB-25 connector's little gold pin that I had dropped for the sixth—or maybe sixteenth—time.

Thankfully, I underestimated my patience/techno-masochism/insanity. Only a week later, I successfully logged in to Ars' editorial IRC channel from the Model 100. And seeing as this machine first saw the market in 1983, it took a substantial amount of help: a Raspberry Pi, a little bit of BASIC code, and a hidden file from the website of a certain Eric S. Raymond.

Meet the machine

In case you're not familiar with it or perhaps have confused it in some way with the slightly more famous TRS-80 desktop, the TRS-80 Model 100 (affectionately known among retro-computing buffs as the "T100") is the Radio Shack-branded version of an early "laptop" computer developed by Kyocera and Microsoft. It was the last system for which Bill Gates wrote a significant amount of code. As we reported in our initial hands-on tour of the Model 100, he considered it his favorite machine ever. (Sadly, Gates was unavailable to take this trip with us down memory lane.)

The machine has some nostalgic significance to me as well—I filed one of my first assignments as a technology journalist with a Model 100, connecting to MCI Mail over dial-up in a phone booth using acoustic couplers. At the time, the machine was a reporter's dream: 20 hours or more of life on four AA batteries plus built-in text editing, address book, calendar, and communications applications burned into an onboard ROM chip. It was easy to overlook the fact that even the top-end Model 100 only had 24 kilobytes of RAM. Literally any modern device surpasses that figure.

My current Model 100 came with a bit of a handicap; it didn't include the AC power supply, the original cables, or the cassette drive used to store and retrieve programs. The documentation was a photocopied, ring-bound duplicate for NEC's version of the same system, so there were a few minor but significant differences. And while many current Model 100 enthusiasts have upgraded the ROM of their systems to extend their capabilities, this one came with the stock ROM from 1984.

To date, I have a well-documented history of trying to drag 1980s technology into the 21st Century. I wasn't going to let a little thing like "no possible way of loading a TCP/IP stack" get in my way.

Step 1: Learning to communicate

The Model 100's built-in serial port is shared by a 25-pin DB-25 RS-232 jack and an internal modem. While the modem was tempting (a return to dial-up!), it can manage only 300 bits per second. The RS-232 connection can do up to 19,200 bits per second—a much more Internet-friendly speed, if the buffer and data bus on the Model 100 could keep up. All I needed was a null modem cable and a USB-to-Serial converter, then I could turn it into a

serial terminal and run an Internet-connected shell session.
Based on the research I had done, it seemed like I would need to make my own null modem cable. While there are some pretty elaborate serial cables used to connect the Model 100 to PCs for file transfers using souped up ROMs and PC software, it looked like my best option (given my current lack of a soldering iron) was a three-wire connection that mirrored the one used for direct serial connections to the Raspberry Pi's pins.

So I ventured to the local electronics store to get a length of cable, male DB-25 and female DB-9 plugs, and the assorted odds and ends required to crimp the pins I needed. The cable worked in a serial test with an old PC laptop, so I went and put covers on the ends to make them look pretty. As it turned out, that wasn't a great idea. The DB-25 plug, covered, didn't fit into the narrow recess where the Model 100's serial output is concealed. I took the housing off the end and went back to the slapdash look.

Meanwhile, I found a source for a USB-to-serial adapter—Retrofloppy—while looking into how to get another venerable computer up and running again. Lo and behold, they had a null modem cable as well, intended for the Apple II Super Serial card. A few days later I had both in hand and successfully performed a serial communications test with my MacBook Air, using GoSerial from FurrySoft. This gave me a basic file transfer capability, but I wanted more.

This initial success with the MBA made me a bit foolhardy. I thought I could turn the Model 100 into a serial console for Mac OS X Yosemite. After all, it's Unix, right? Well, apparently not as much as it used to be—the configuration files for serial consoles in Yosemite appear to be mostly vestigial, and though there were some resources online that suggested I could pull it off, they were all several years old.

For sanity's sake, I set aside the MBA effort and returned to my original goal: the Raspberry Pi and Raspbian (the Debian Linux distribution built for the ARM-based system-on-card). I have a couple of the older Raspberry Pi Model systems, now made obsolete by the new Pi 2 Model B, so I decided that one of them would make a good companion to the even-more-obsolete Model 100.

FURTHER READING

HOW TWO VOLUNTEERS BUILT THE RASPBERRY PI'S OPERATING SYSTEM

Raspbian makes the Pi go faster—and supplanted Fedora as the #1 OS.

Raspbian is, because of the nature of the Pi, set up by default to handle serial console sessions—but through the pins on its bus rather than through one of its two precious USB ports. It also is configured to connect serially, by default, at 115,200 bits per second. And there's another small (literally) problem: the screen of the Model 100 is only 40 columns wide and 8 lines high, and one of those lines is consumed by menu choices for its TELCOM terminal program.

These issues are, mercifully, resolved with a few configuration changes to Raspbian's startup and terminal definition files. First, I needed a definition for the terminal configuration for the Model 100. Once upon a time, there was in fact a terminal configuration for the Model 100 in Linux's *terminfo*, but it was cast off into a "UFO" file around 1995. Guess what? Eric S. Raymond has a copy of that UFO file in a gzipped archive hidden in the recesses of his website, which I discovered by URL-hacking my way from a link to his old site from Brian

"Danger" Hicks' page, "The Model 100 Serial Terminal Project." Here's the relevant bit:

```
1     trs100|Radio Shack Model 100:\

2     :am:bs:xt:\

3     :co#40:li#8:\

4     :al=\EL:bl=^G:cd=\EJ:ce=\EK:cl=\EE:cm=\EY%+ %+ :\

5     :cr=^M:dl=\EM:do=^J:ho=\EH:kb=^H:kd=^_:kl=^]:kr=^\:\

6     :ku=^^:le=^H:nd=34:se=\Eq:sf=^J:so=\Ep:up=\EA:\

7     :ve=\EP:vi=\EQ
```

I put that into a text file on the Raspberry Pi called *model100* and then used it to recompile Raspbian's terminal information library:

```
1     tic model100
```

Bam! I had a terminal config called "trs100" to reference.
On the Model 100's side, I needed to configure the TELCOM application to communicate with that serial console session. TELCOM uses a set-up command, "Stat," to configure the speed and communications settings for a terminal session. Stat configures the connection based on a five-character string that uses single characters to define:
The baud rate (1 = 75, 2 = 110, 3 = 300, 4 = 600, 5 = 1200, 6 = 2400, 7 = 4800, 8 = 9600, 9 = 19200);
The size of the bit words of data sent over the connection (7 or 8 bit);
The parity bit used for error checking (E for even, O for odd, N for none, I for ignore);
The number of stop bits (1 or 2);
And whether XON/XOFF is enabled (E) or disabled (D).
So, for example, to configure TELCOM on the Model 100 for my goal speed of 19,200

baud, using 8 bit words, no parity, 1 stop bit, and disabled XON/XOFF, the setup string would be:

```
1     Stat 98N1D
```

Next, I needed to configure Raspbian to open up a serial console connection to the USB-to-serial adapter at startup. Console sessions are configured in the /etc/inittab file in Raspbian, like most Linux distributions, which launches processes during startup. The part of inittab I was interested in looks like this:

```
1     #Spawn a getty on Raspberry Pi serial line

2      T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100
```

I wanted to add an instance of getty, the program in most *nixes that runs terminal console sessions, that connected to the USB serial adapter with a speed and definition that the Model 100 could handle. The device ID for the serial-to-USB adapter was USB0, and I was going for 19,200 for my connection, so I added:

```
1     T1:23:respawn:/sbin/getty -L ttyUSB0 19200 trs100
```

Then I saved and restarted. At first blush, it seemed to be working flawlessly.

Step 2: Going online!

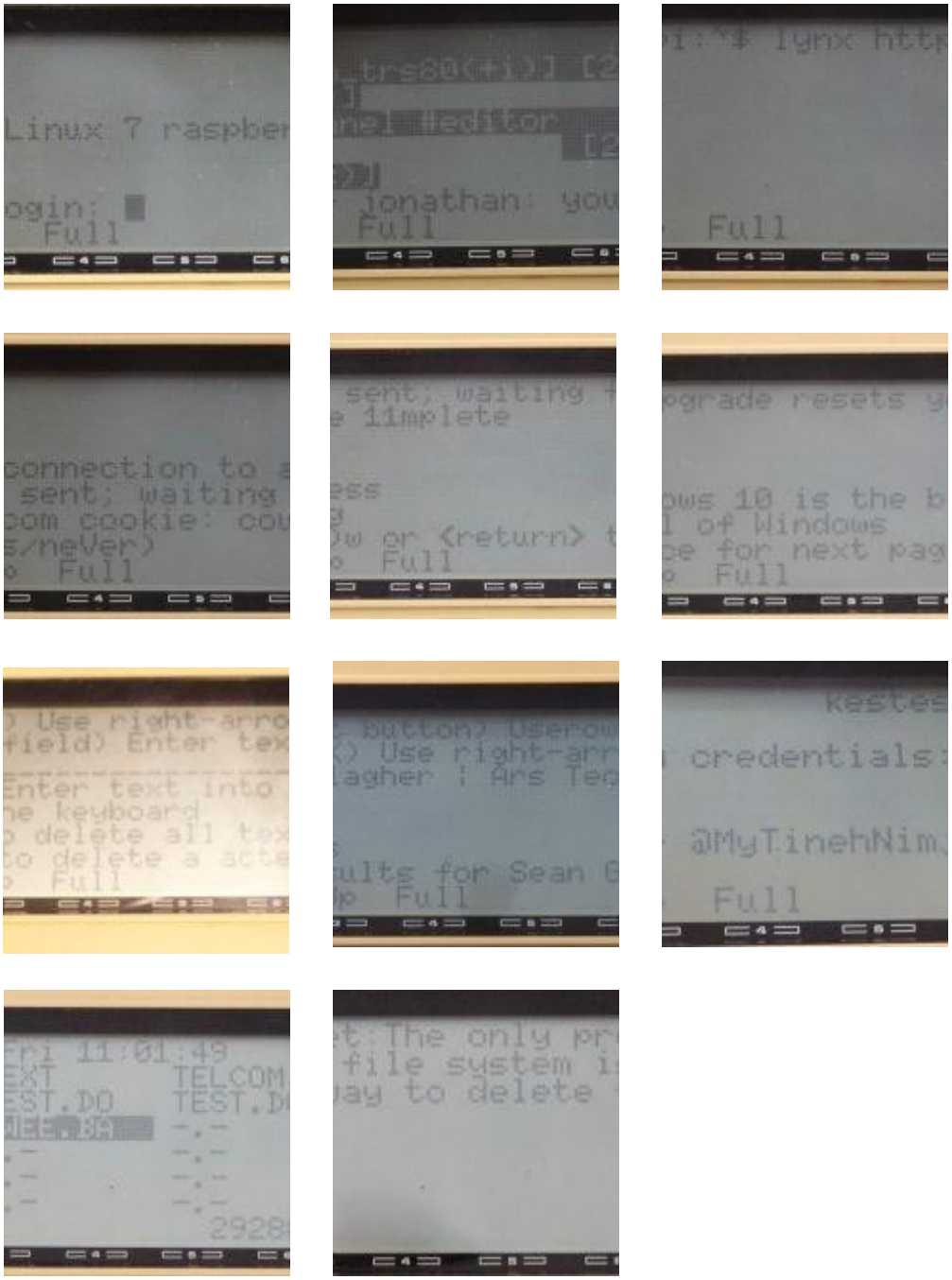
After some appropriate celebratory whooping, I set to getting my newly connected conjoined computers configured for doing actual things on the Internet from the command line so I could actually do work with them. The natural first choice was Internet Relay Chat, since we at Ars essentially use IRC like other people use shouting down the hall or over the tops of cubicles. I did a quick apt-get install of Irssi, a command-line IRC client. Within a few minutes, I was announcing my success on Ars' editorial IRC channel.

From my end, though, it was not very pretty. In part, that was because Irssi is not exactly suited to being squished into 40 columns and 7 lines. But 19,200 baud was apparently more than the Model 100 could handle—there were a lot of truncated lines, dropped characters, and occasional gibberish.

The same was true for browsing the Web. Lynx, the text-based Web browser, overcame the

Model 100's data bus, and it resulted in a stream of gibberish only occasionally interrupted by bits of text that were recognizable as content.

So, optimism crushed, I reset the baud rate at what the Model 100's internal modem is rated for: 300 baud. Suddenly, things got a lot clearer (and slower). I pushed the speed up to 1200, and it seemed to stay stable while not being so slow I had to shave between commands. However, most websites (including Ars) are not built for Lynx. Lynx is also written for



taller, wider screens. The same is true of Irssi, which now returned full lines of chat but wrapped around in ways that left artifacts of text hanging on the screen. Still, I was able to run a Google search and have conversations with other Arsians well enough to consider IRC and Web connectivity to be acceptable within the limits of the challenge. There's a possibility that there's a better way to access Web content from the Model 100, but I think it might require custom coding.

Step 3: fun with command-line Internetting

Next came Twitter. There's an excellent command-line client for Twitter written in perl called ttytter. It offers a number of ways to do batch processing of tweets via chron, so direct messages and other tweets can get dumped to a log file. And you can send status updates through the command line or from ttytter's own runtime prompt.

The final piece I wanted for initial operating capability was e-mail. I didn't want a full-blown mail server and Unix mail capability bogging down the Pi, but there are some simple command-line tools that allow sending and retrieval of e-mail.

For the sending half, I installed SSMTP, mailutils, and mpack on the Pi. (There's a good tutorial on setting up SSMTP and mailutils here and more on mpack here.) In theory, with these installed, as long as I can remember someone's e-mail address (or have it on a business card they've handed me), I can fire off an e-mail to them from the Model 100 or even send them (or myself) longer documents I've created on the machine.

Here's how mailutils sends an e-mail from the command line:

```
1    echo "This is the body of my email" | mail -s "Subject" thepersonIamemailing@gmail.com
```

See that pipe (|) in there? Guess what else the Model 100 doesn't have on its keyboard—"|" (the "pipe" symbol used to chain commands). However, the Model 100 does have a "Graph" button for summoning additional ASCII characters from its reduced key set, so I had to do some digging to find exactly which key combination produced one. (For the re-

cord, it's "Graph _".)

To send e-mails with attachments, I needed to use mpack. The mpack command is a bit more keyboard-friendly for the Model 100:

```
1    mpack -s [subject] [name of file to enclose] [e-mail address of recipient]
```

To retrieve messages from my mailbox, I installed fetchmail. There's a good tutorial on configuring fetchmail for Gmail and writing retrieved e-mails into a file by Cristian Pascottini; I haven't fully debugged my configuration yet, in part because I've got very little interest in trying to actually read my endless e-mail stream on the Model 100. For now, sending is enough. But I may want to use fetchmail to push new files to the Model 100 in the future using its serial file transfer capabilities.

For now I've got command-line tools on the Pi to send and read Twitter posts and to send e-mail. To create attachments for e-mail, I can either write them in the nano editor in a terminal session (which is ugly) or write them in the Model 100's built-in text editor (TEXT) (which is much less ugly) and transfer them to the Pi when I want to send them. For example, I can write blog posts offline and then e-mail them to a WordPress site's account for auto-posting. Using the *cat* command allows me to do an ASCII file transfer from the Model 100 using the TELCOM program's built-in file upload function.

At command line, type *cat > uploadfile.txt* (where uploadfile.txt is the name of the destination file).

Hit F3 on the Model 100 for "Up," and then enter the name of the file to be transferred.

After it streams up, hit Ctrl-D.

You're done. You can *mpack* the file off to another computer via e-mail now.

Step 4: Writing Internet apps for the Model 100

Some of you (this means you, Nate Cardozo) are probably saying, "Well, Sean, that's all... interesting, but you're not actually doing anything with the Model 100 on the Internet. You're just using it as a dumb terminal." Well, it's time to fix that.

The Model 100 has two very interesting features. First, it has Bill Gates' built-in BASIC interpreter. And second, in a very Unix-y/POSIX-y kind of way, it treats its serial device like a file that can be written to and read from. As long as the serial connection is open, a simple BASIC program can act as a front-end to a remote command-line. For example, I wrote this program, TWEET.BA, to script interaction with the ttytter Twitter client:

```
10 LINE INPUT "Type your tweet"; TW$
20 OPEN "COM:58N1D" FOR OUTPUT AS 1
30 PRINT #1, "./ttytter"
31 PRINT #1, TW$
32 PRINT #1, "/quit"
40 END
```

All I have to do is connect to the Pi, log in, and then exit the TELCOM program to execute TWEET.BA from the Model 100's main menu. In this case, I'm using ttytter's interactive mode; if I wanted to (and I probably will), I could write a menu-driven interface for this program and add the ability to retrieve tweets from others, retrieve direct messages, and respond to messages all from the Model 100's native BASIC interface. The same is true for anything else that can be done from a Linux command line on the Pi; I'm working on a BASIC e-mail client next and then perhaps a few more interesting applications—like automating Wi-Fi setup for the Pi.

Going on the road with the Model 100/Pi combo requires some interesting config contortions. While there's a graphical tool to find and connect to Wi-Fi networks in Raspbian's GUI, changing the Wi-Fi network from a serial console session requires editing */etc/wpa_supplicant/wpa_supplicant.conf* to change the SSID, security type and password, and then rebooting the Pi. Instead of interactively editing the config file in nano on the Model 100's screen, a BASIC program could just ask for the setup info and overwrite the old config with the *cat*command. When I get WIFI.BA debugged, I'll post it here.

There are other interesting things to do from the command line. For example, there's tshark, the command-line version of the Wireshark network monitoring tool, and other network monitoring, security auditing, and penetration testing tools. It's possible to write console input back to the Model 100's memory from BASIC as well. But given that it only has about 24k of RAM total, packet logs are probably not the best thing to do that with.

I'll be taking this rig to Def Con with me, so maybe I'll teach this old laptop a few more new tricks in the meantime. (Rest assured, we'll update this article as I do.) But another two *Back*

to the Future challenges wait for me upon my return to Vegas—the Apple II Plus and the Amiga 500. Like the Model 100, both are begging to be connected to the Internet. *At least.*

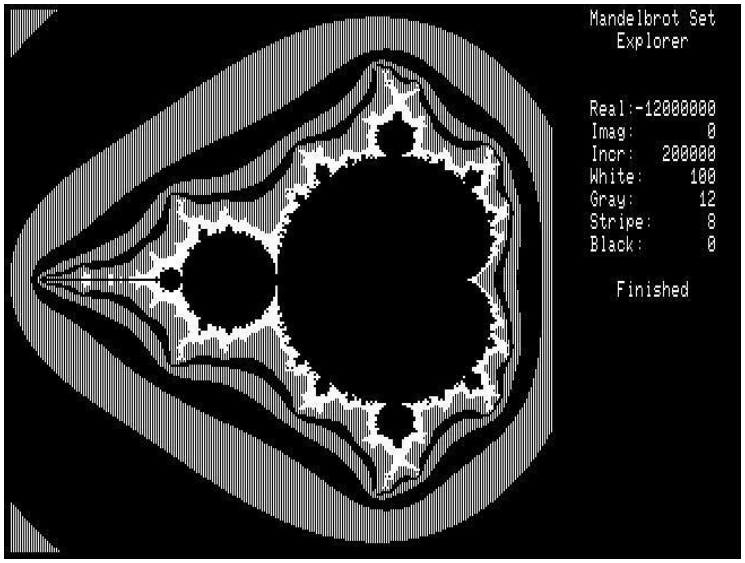
© **Sean Gallagher** / Sean is Ars Technica's IT Editor. A former Navy officer, systems administrator, and network systems integrator with 20 years of IT journalism experience, he lives and works in Baltimore, Maryland.

To checkout this article direct, please visit :-
http://arstechnica.com/information-technology/2015/08/surfing-the-internet-from-my-trs-80-model-100/

LOOKING FOR HIGH-RESOLUTION GRAPHICS FOR YOUR TRS-80 MODEL 4/4D?

Then you simply must have the

"Improved Grafyx Resolution Solution!"



-Improved version of the established Grafyx Solution GX 4.2 board
-640x240 resolution for a total of 153,600 pixels, High Bandwidth
-Easy install in minutes inside your Model 4, Gate-Array Model 4, or 4D
-4P version is just around the corner - stay tuned!
-Compatible with most M3 and M4 Hi-Res software for Radio Shack HRG
-Designed in Belgium and proudly built and shipped from Australia or UK

Order yours today exclusively from:

UK and EU: BetaGamma Computing: bgialopsos@aol.com

Rest of the world: http://ianmav.customer.netspace.net.au/hires/hires.htm

NEW REDUCED PRICE USD$165
(DUE TO AU$ EXCHANGE RATE CHANGES)
WITH ADDITIONAL DISCOUNT FOR FURTHER PURCHASES.
CHECK WITH IAN FOR FULL DETAILS

In Mav's FreHD Workshop

The new FreHD Clearly Superior series

by
Ian Mavric

Since the FreHD (designed by Frédéric Vecoven, in Belgium) has been released it has been very popular with the TRS-80 Model I/III/4 community and become the de-facto standard mass storage device for people wanting to progress beyond floppy diskettes.

The FreHD has always been sold with the weekend hacker in mind, at a price always quite reasonable when compared to other similar products or refurbished TRS-80 hard disk drives. As the buying public has changed over time, I've added variations to the basic FreHD core product. Whereas in the early days people were happy with a naked PCB arrangement (Kit C) or for those who had soldering ability and wanted to build a FreHD for the least amount possible (Kit A or B), I've responded to the requests of those after a different approach to their FreHD.

This resulted in the neat little BoHx for a really polished looking device next to your TRS-80, and the CKD kit, for those who love to solder but don't have boxes and parts, and wanted a Heathkit style kit delivered with all required parts and simple assembly instructions. Yes, I have been listening.
Retro computing as a hobby has grown exponentially in the last five years. Prior to 2011 most people who enjoyed this hobby had an electronics, or assembly language, or Basic programming background, and oftentimes had originally owned these computers back in the day.

They new how to operate them, and how to fix them when they had a failure. As retro computing has taken off, I see a new generation of enthusiasts who just want one for different reasons... among them being that their parents or grandparents may have had one, or they used one at school or university, or they just like the cool retro look.

These people have no electronics or programming background, but still have equal enthusiasm but they require a system that turns on and just works.

A growing number of people emailed me stating they didn't want to be bothered with soldering, or sourcing a 5V power supply, or even where possible, opening their TRS-80.

What they wanted was a plug-in and play device. I had to think about this for a bit; it would need to come complete with all cables and power supply, but still needed to be shipped for around the same low price. It needed to auto-boot if possible, and be packaged in a neat little enclosure.
In short it needed to bring together all the individual FreHD products I sell into complete packages ready to use.

Tandy's advertising department had a name of this type of computer product in the 1980s, they called them Clearly Superior. As a tip of the hat to the old Radio Shack computer division, this is what I refer to complete FreHD packages as.

For the Model I

- Two versions depending on what your Model I system comprises of

Version 1: FreHD/Quinnterface Combo box for the standard 16K Level II Model I

I looked at my FreHD Kit C set up on the table connected to a Quinnterface, both items daisy chained to a 5V power supply, and decided it was too messy.

Even a 16K Model I with just a monitor is, in my opinion, too messy on the table. What was needed was to package the FreHD and Quinnterface into a BoHx, but the plastic enclosures I'd been using for several months were not suitable, so I tried some different enclosures and internal arrangement of components, and settled on the small Pac-Tec high impact ABS plastic enclosure you see in the picture.
It isn't cheap but it looks great!

Supplying power to this device in which two individual items are mounted inside and connected by data and daisy power cable needed a decent 5V power supply.

This make the units sellable in both Europe and the USA, I include one of two different power adapters. A CE-rated one for Europe and a UL-listed one for the USA. Both run on any voltage from 100V to 240V. You just plug it into the FreHD/QI Combo box and plug the data cable into your Model I.
No modifications to your Model I are needed, since the Quinnterface automatically boots from the SD card on the FreHD. Also, since 16K with a hard disk operating system makes no sense at all, the Quinnterface has 32K additional memory which expands your system to 48K.

No other similar product adds both hard disk emulated storage AND memory upgrade in one plug in unit. And the icing on the cake? Three great operating systems are supported: LDOS, Newdos/80 and Dosplus 3.5

Picture: FreHD/QI Combo box for the 16K Level II TRS-80 Model I

Picture: how the FreHD and Quinnterface are packaged inside the heavy-duty plastic box

Version 2: FreHD/M1HD adapter Combo Box for the Expanded Model I system

One of the biggest gripes of the expanded Model I system is how many mains power sockets the system takes up.

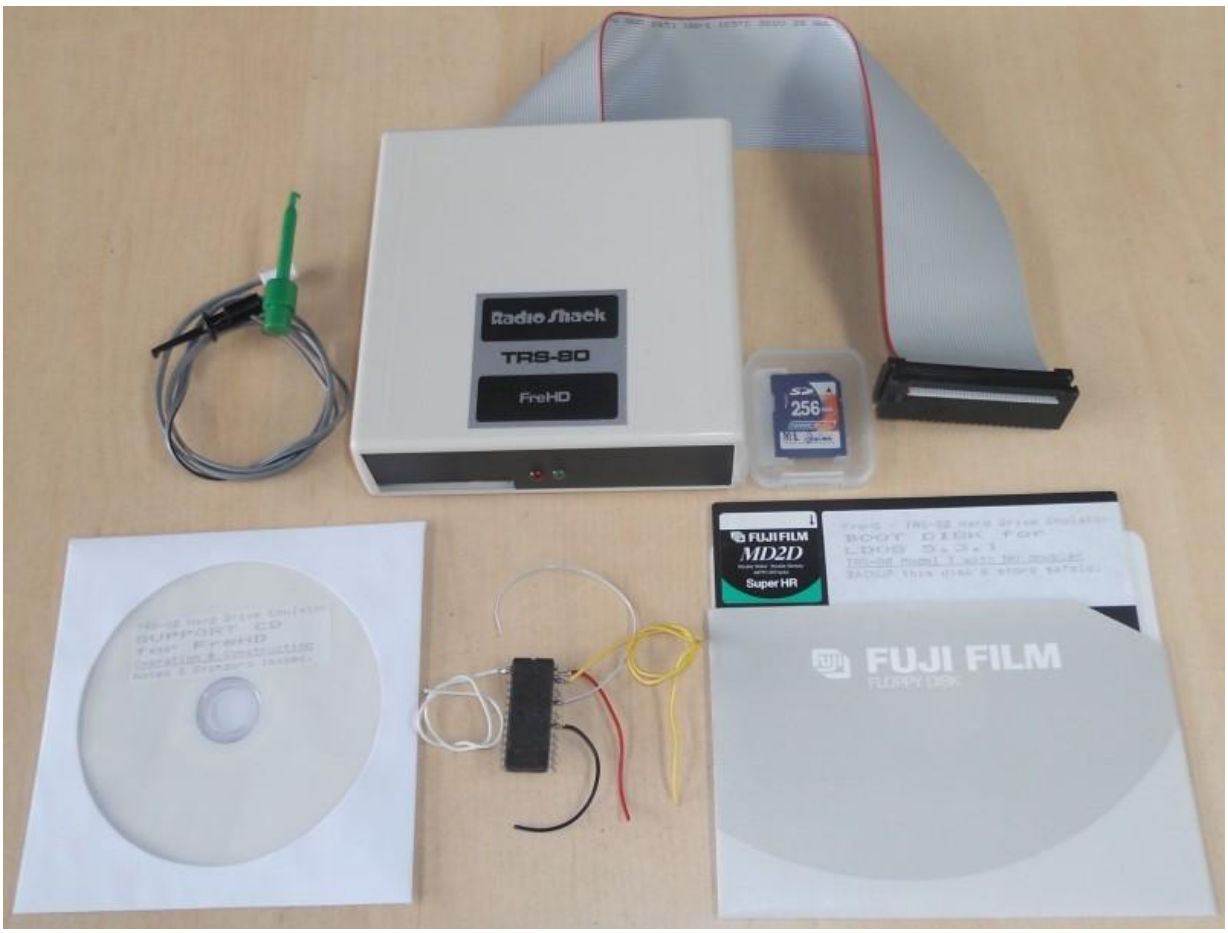
A simple dual drive Model I needs five power sockets, add a tape deck, six, add a 3rd disk drive, seven... it never ends. (My workhorse Model I back in the 80s had two disk drives, tape deck, hard drive, printer and modem - nine power sockets!)

So one of the main objectives was to not add yet another power supply if possible, and power the FreHD from excess volts from the Expansion Interface.

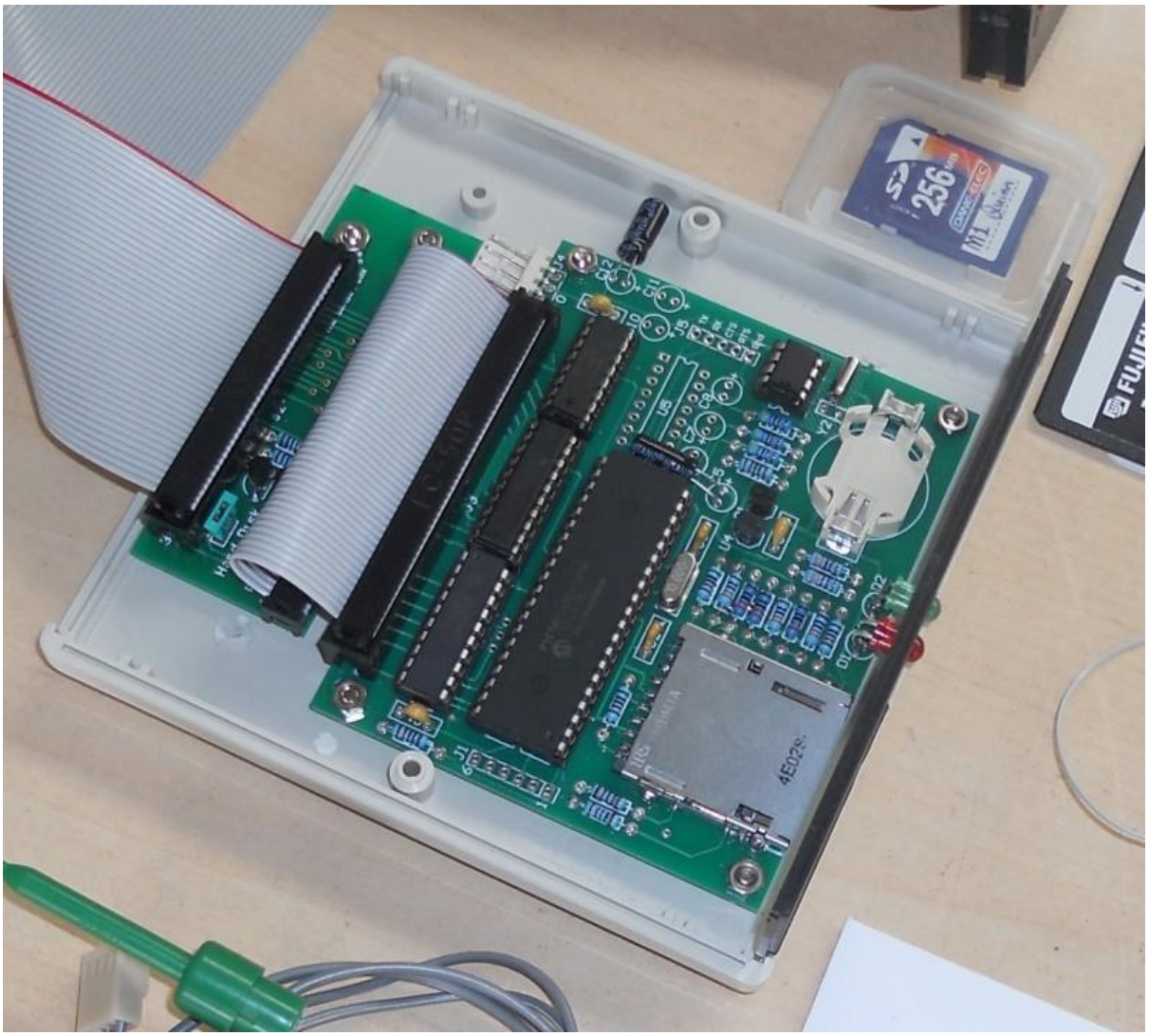
Into the same high-impact plastic enclosure goes the FreHD and this time my reproduction M1 HD adapter as well. It's joined by a very short cable and the M1 HD adapter is powered from the expansion bus. To power the FreHD this time, I supply a power cable with two clips on the end, one is connected to 5V and the other to ground, inside the Expansion Interface. This saves plugging an extra wall-wart 5V power supply into an already crowded power strip.

Two boot options are included with every one of these. The traditional floppy boot diskette, just like Tandy designed the hard disk based Model I back in 1982, or you can install the included FreHD auto boot Eprom.

It takes about half an hour, requires no trace cutting, but you do need to know how to solder. By installing the auto boot Eprom you have access to the same three operating systems mentioned above.



Picture: FreHD/M1HD adapter Combo box for the expanded Model I system



Picture: the M1HD adapter and FreHD neatly mounted inside the plastic enclosure

Version 3: For the Model III, 4, 4P, 4D

Choosing a plug-in and play FreHD for your Model III, 4, 4P and 4D is easy.

I've included the necessary inter-connection cable, and power cable, AND the auto-boot Eprom to install in your TRS-80.

You simply open your TRS-80 and connect the power cable to tap off 5V from the floppy drive, and install the auto boot Eprom where your ROM C chip currently lives.

On the Model 4 Gate Array version and 4D the Eprom replacement is a little more complex, involving soldering four wires. Again, for those who don't like to solder, I can include a hard disk boot diskette and your system will run just like Tandy designed their own hard disk systems.

Once installed, you have access to more operating system choices than any other similar product on the market: LDOS and Newdos/80 on the Model III, and LS-DOS, LS-DOS High-Resolution version, and CP/M on the Model 4. FreHD is also the ONLY product of it's type which will work on those diskless Model III and 4 systems (upgrading them to 48K and 64K respectively is recommended).

Note to collectors: Diskless Model III and 4 computers are often cheaper to buy and have had less use than their disk-based brothers, and they also weigh a lot less - great if lugging around a full Model III or 4 disk computer makes your back ache!

Finally on the 4P auto booting is standard for LS-DOS, LS-DOS High-Resolution version, and CP/M. All 4Ps came with an auto-bootable Bios Rom and we use this to automatically boot your FreHD.

Picture: FreHD kit for the Model III, 4, 4P and 4D includes Eprom and power cables

In most cases these Clearly Superior FreHD kits are tailored to your individual TRS-80 based on information provided to me. An example of this is the diskless version comes with a different power cable for connection directly to the power supply in diskless TRS-80s.

Pricing

It doesn't take a genius to work out that while these complete kits are more expensive than the old Kit C, when you add the costs of all the items additional to make one of these plug-in and play FreHDs you come out way ahead.

Version 1: $USD249.00 and for a limited time includes free shipping! (Separate items cost would be $179 (FreHD) $95 (Quinnterface) $25 (BoHx) so the saving is $50 and now I include free CE certified or UL-listed Power adapter)

Version 2: $USD229.00 and for a limited time includes free shipping! (Separate items cost would be $179 (FreHD) $40 (M1HD adpator) $25 (BoHx) $25 (Eprom) so the saving is $USD40)

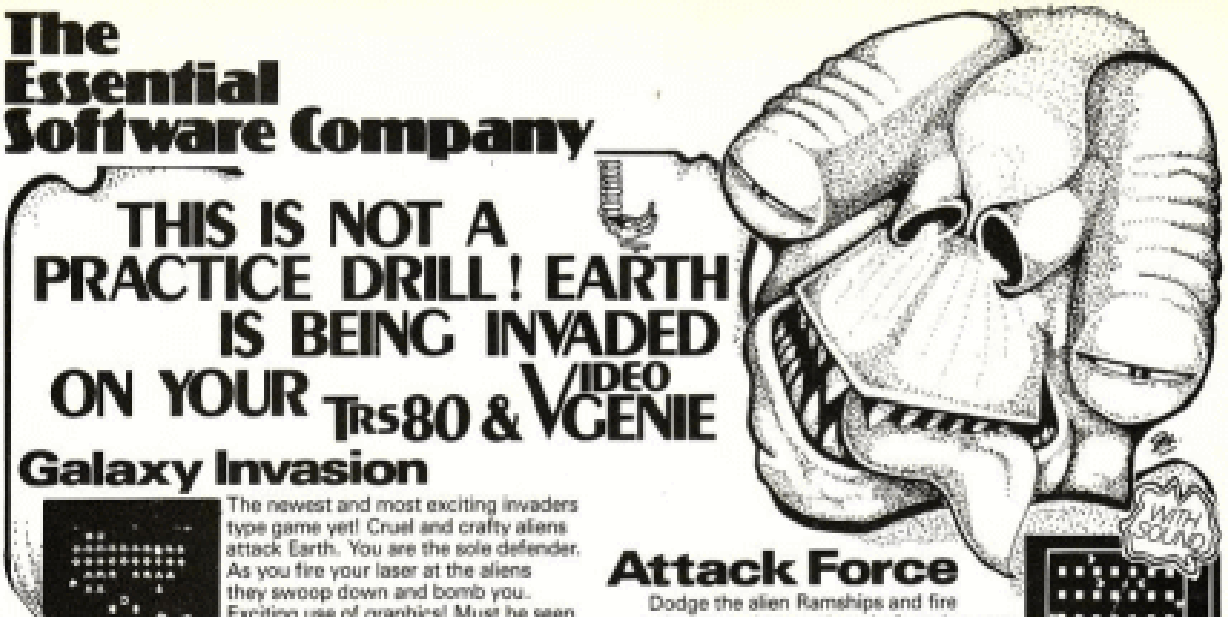
Version 3: $USD210.00 and for a limited time includes free shipping! (Separate items cost would be $179 (FreHD) $40 (HD cable) $25 (BoHx) $20 (Auto Boot Eprom) so the saving is $USD60)

Order yours To-Day:

http://ianmav.customer.netspace.net.au/clearlysuperior/

The Essential Software Company

THIS IS NOT A PRACTICE DRILL! EARTH IS BEING INVADED ON YOUR TRS80 & VIDEO VGENIE



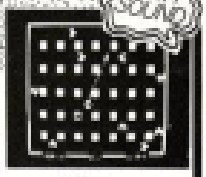
Galaxy Invasion

The newest and most exciting invaders type game yet! Cruel and crafty aliens attack Earth. You are the sole defender. As you fire your laser at the aliens they swoop down and bomb you. Exciting use of graphics! Must be seen.

TRS 80 Level I & II 16K Tape

Video Genie 16K Tape

Attack Force

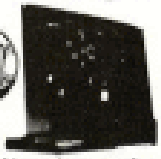


Dodge the alien Ramships and fire missiles to destroy them before they get you. The alien Flagship uses his deadly laser bolt to transform a Ramship into another Flagship or into your ship's double. Look out! Destroy your double and you could destroy yourself.

TRS 80 Level I & II 16K Tape

Video Genie EG3003 16K Tape

SuperNOVA



ORDER NOW

Now the amazing ASTEROIDS arcade game for your TRS 80! Your ship is floating in the middle of an asteroid belt! Your only escape is to destroy them and the crafty alien spacecraft! Blast them with your laser, thrust, rotate or hit hyperspace to survive!

TRS 80 Levels I & II 16K Tape

Video Genie 16K Tape

Robot Attack

THE GAME THAT TALKS

TRS 80 Levels I & II 16 K Tape

Video Genie 16 K Tape

The Newest and Most Astounding Arcade Game that TALKS has just Reached Planet Earth. You can't help yourself. You have to stop them at all cost. Don't let up. Written especially for high quality graphics you'll simply be dazed and excited by the action.

Cosmic Fighter

Your fighter appears below a convoy of Aliens! If you destroy them another set appears who seem to be slightly cleverer than before! Soon your space station nears but before you can dock the station comes under attack! Survival is up to you! The excitement is just beginning!!

TRS 80 Levels I & II 16K Tape

Video Genie 16K Tape

Gobble MAN

Watch out behind you! As you hurry through the maze collecting your energy modules you score points. But don't let the Gobblemen catch you. If you are crafty, sneek up behind them and neutralise them to gain extra points. Just keep a watch. When they attack you they come in fast. Just don't lose your nerve.

TRS 80 Levels I & II 16 K Tape

Video Genie 16 K Tape

3D Adventures

for TRS 80 LEVEL II 16K

3-D means that as you wander through the mazes and buildings, a full screen graphic display constantly shows your position in a perspective format as though you were actually there! This "rat's eye" view adds an entirely new dimension to adventure.

English language commands can be entered at any time to manipulate your environment. The command sets are extensive and sophisticated. Dozens of objects are scattered throughout the mazes and buildings. You can pick them up, burn them, throw them, etc. You may need the sword to fight off an ugly little man. Or a steel rod to hold apart crushing walls. Deathmaze 5000 and Labyrinth allow the traditional one and two word commands. Asylum incorporates our Advanced Language Interpreter (ALI), which allows full sentence input.

Deathmaze and Labyrinth consist of over 550 locations! Asylum tops 1200 locations!

All Tapes £10 ea

ASYLUM Tape £12

THE ESSENTIAL SOFTWARE COMPANY
(Viscounti Ltd.) 01-837 3154
47 Brunswick Centre, London WC1N 1AF

I have a ..... microcomputer.

☐ Please send me your software catalogue. I enclose a stamped self addressed envelope.

☐ Please send me ..... I enclose a cheque / postal order for £ ..... (plus 50p post & packing)

Signature .....

Name .....

Address .....

Postcode .....

My ACCESS No is .....

Wynford and Jane James continue their series on portable graphics with a program which can be used in a variety of adventure games.

Maze-making subroutines

MANY GAMES require a subroutine which makes a maze to provide a variable background against which a battle can be fought or a chase can take place.

The various adventure games are in essence, maze games. The major difference is that in an adventure where you successfully go west across the Lava Sea, there is no guarantee that the reverse journey is possible. In a standard maze, the player who wanders into a dead end at least has the option of back-tracking.

This article describes a program which first generates a maze of random dimensions. It then displays a perspective view of the maze as the player searches for the exit.

The maze-creation routine, and a second routine which draws a map of the maze, are written so that they will run on any micro which has a memory-mapped display and supports Peek and Poke. The only changes required are to two lines in the initialisation routine.

The three-dimensional view of the maze requires several common graphics symbols: the horizontal, vertical and diagonal lines, and the four "corner" symbols. These symbols are certainly available on the Pet, UK-101, Sharp MZ-80K and Superboard, and probably on most other popular micros too. The only change you may need to make in this part of the program is to substitute the symbols in your character set for those in the program — just three line changes. If your micro has a Plot command you will have to re-write this routine, but you should be able to simplify it considerably.

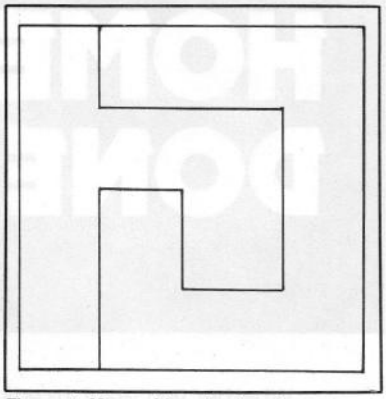


Figure 1. Maze with a dead end.

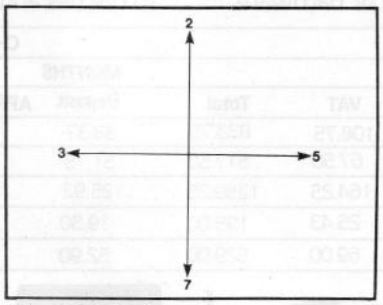


Figure 2a. The four prime numbers which indicate walls W(0) to W(4).

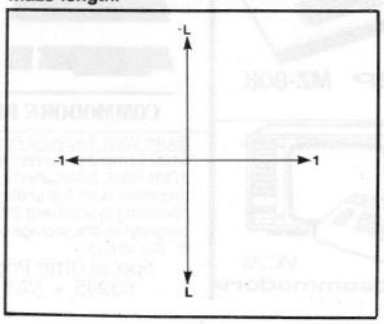


Figure 2b. The change in cell number on moving to a new cell, M(0) to M(3); L is the maze length.

You may be looking sadly at your minimal memory configuration, thinking that this article cannot be for you. Let us reassure you — it was written on a 4K-RAM Superboard-II. If you have more than 4K of memory, you should be able to produce a full-blown adventure using the program as a basis.

Two main problems have to be overcome when creating a maze. Firstly, you must ensure that all parts of the maze are accessible from all other parts. There must be no "closed circuits" — see figure 1 — otherwise the effective size of the maze will be reduced and memory wasted preserving unused portions.

Secondly, you have to be able to record the number and position of the walls in each cell in order to store the layout of the maze — see figure 2.

We chose to indicate a wall in each direction by the prime numbers 2, 3, 5 and 7. Initially all cells are given a value of 210 — equal to $2 \times 3 \times 5 \times 7$ — which indicates a wall in every direction.

Other possible wall arrangements can be stored as multiples of the relevant prime numbers. For example, a cell with walls only at north and south would be given a value of 14 — equal to 2×7.

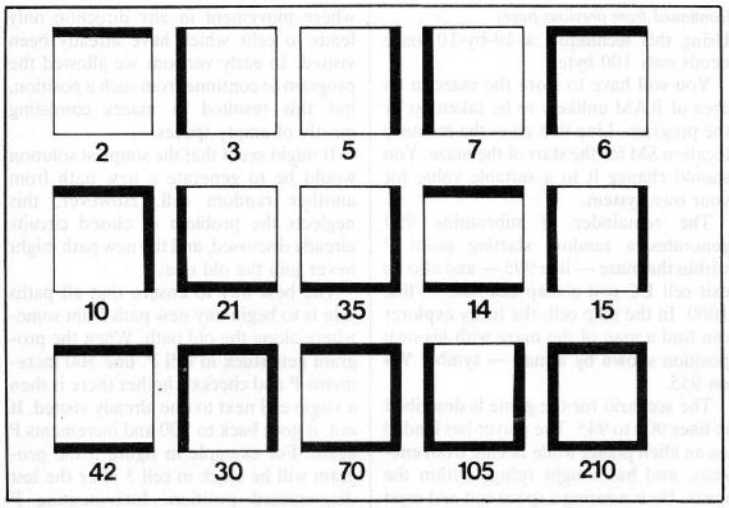


Figure 3. Cell values for each of the possible wall arrangements.

Figure 3 gives all the cell values and the wall arrangements they indicate.

Lines 500 to 595 create the maze. This part of the program is short but not very fast, so the maze is created while instructions are being given in subroutine 900.

Subroutine 900 also carries out the initialisation. Line 950 must be changed: it gives the memory locations of the top left, top right, bottom left and bottom right of the screen memory — TL, TR, BL and BR. It also sets the line length LL, the depth of wall DW, and the cell depth CD.

Obviously LL should be set to the line length of your own micro. The depth of wall DW is the number of pixels to be used to draw each cell wall. The cell depth CD is a necessary limiting factor due to the lack of high resolution on our micro. When a certain number of cells of diminishing size have been drawn to give the illusion of perspective, the centre of the screen is reached. There is clearly no point in trying to go further and attempting to display more cells.

The procedure is easier to follow if you look at the values used in the program. Although the line length LL for the Superboard is 32, the screen memory has an actual line length of 25. With a depth of wall of three, and a cell depth of four, the maximum possible display of left-handed walls uses four sets of three pixels = 12 pixels. The right-hand walls use another 12 pixels, so the perspective display of four maze cells uses 24 pixels altogether. As the screen has a line length of 25, this leaves a single, blank pixel at the centre of the screen to suggest the limits of vision — see figure 4.

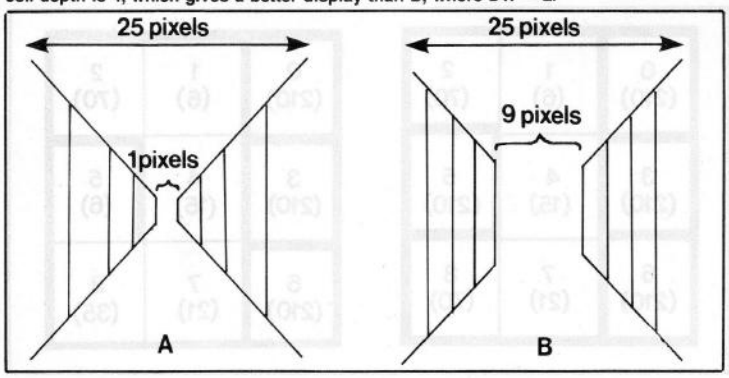
You will have to choose values of DW and CD carefully to get the best display. An important factor to consider is the possible longest length of a corridor in the maze. There is no point in setting CD to 8 if most mazes have dimensions of less than eight cells. The program as written varies the length L and width W of the maze between six and 10 cells. If you want to adjust this, change line 960.

As already stated, all cells are given an initial cell value of 210 to indicate four walls. In early versions of the program the values for each cell were stored in an array. However, each element in an array uses four bytes in the Superboard memory. A 10-by-10 array takes up 400 bytes, which is a large chunk out of a 4K RAM.

As all cell values are 210 or less, they can be Poked as single bytes into memory.

(continued on next page)

Figure 4. Two views down the same 10-cell corridor. In A, the depth of wall, DW, is 3 and the cell depth is 4, which gives a better display than B, where DW = 2.



(continued from previous page)

Using this technique, a 10-by-10 maze needs only 100 bytes.

You will have to store the maze in an area of RAM unlikely to be taken up by the program. Line 995 gives the memory location SM for the start of the maze. You should change it to a suitable value for your own system.

The remainder of subroutine 900 generates a random starting point P within the maze — line 995 — and also an exit cell EC and a map cell MC — line 1000. In the map cell, the lucky explorer can find a map of the maze with his own position shown by a man — symbol W4 on 955.

The scenario for the game is described in lines 900 to 945. The player has landed on an alien planet while fleeing from enemies, and has sought refuge within the maze. He is wearing a space suit and must find a way out of the maze before his limited oxygen supply expires. The oxygen supply OX plus its rate of loss OL are randomly determined in lines 1010 and 1015. While the player is digesting this information, the maze is created in subroutine 500.

Beginning at the cell P within the maze, the program moves in a random direction. Cells in all four directions are examined successively until an un-visited cell adjacent to the present position is found — lines 510 to 570.

The new cell number is found by adding the correct movement increment M(0) to M(3) — figure 2 — to the present cell number, line 540.

Each time the program moves from one cell to another, the wall in the direction of movement is eliminated by dividing the cell value by the corresponding wall value W(0) to W(3). This operation has to be done twice: the present cell value CP is divided by the appropriate wall number W(A) on leaving the cell; and the cell value after movement CM must also be divided by the number for the wall through which it was entered, line 580. See figure 5 for example.

This random progress through the maze continues until a point is reached where movement in any direction only leads to cells which have already been visited. In early versions we allowed the program to continue from such a position, but this resulted in mazes consisting mostly of empty spaces.

It might seem that the simplest solution would be to generate a new path from another random cell. However, this neglects the problem of closed circuits already discussed, and the new path might never join the old one.

The best way to ensure that all paths join is to begin any new path from somewhere along the old path. When the program gets stuck in cell P, line 500 increments P and checks whether there is then a virgin cell next to one already visited. If not, it goes back to 500 and increments P again. For example in figure 5 the program will be stuck in cell 5 after the last diagrammed position. Incrementing P gives cell 6, and as this is an un-visited cell adjacent to the path, the new path would begin from cell 7 to cell 6.

Some checks have to be included because certain movements are forbidden at the maze edges. Line 540 guards against moves off the top or bottom row of the maze. Line 570 prevents the program falling off the sides. For example, in figure 5, although adding 1 to the cell is normally allowed, the move from cell 5 to cell 6 is prohibited because 5 is an edge cell.

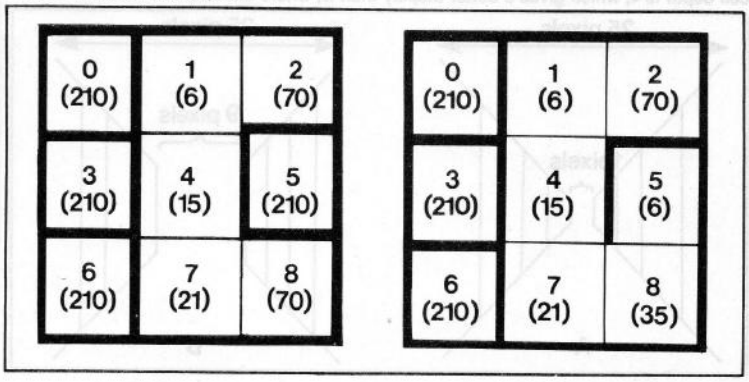
A count C is kept of the number of cells visited. When all cells have been used in the maze the subroutine terminates.

Between two and three cells per second are generated. On average, a 10-by-10 maze takes about 50 seconds on the Superboard, but the time taken is very variable.

Subroutine 600 draws a map of the maze. To make this as universal as possible the only graphics symbol used here is the one with the whole pixel filled.

A wall in any direction is shown by three of these symbols. Because adjacent cells share a wall in common, the maze in figure 5 would use seven pixels horizontally and seven pixels vertically.

Figure 5. Generation of a maze. In diagram A the program has moved through the cells 2-1-4-7-8. The program can only move to cell 5, as in B, since it must always visit new cells. Cell values are given in brackets.



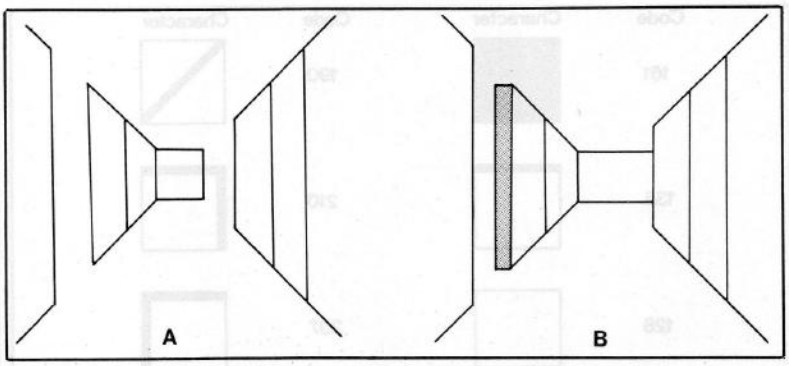


Figure 6. Two views of the same corridor. Display B includes fixes to improve appearance.

The size of the maze that can be displayed obviously depends on the dimensions of your screen display. A maze of side n cells requires 2n + 1 pixels. The maximum size maze that can be shown with 40 characters per line and 25 lines is 19 by 12.

If you enjoy finding your way through a maze against the clock you can use the subroutines 500 and 600 alone, without the perspective view. You will have to add a few lines to enable you to move within the maze. Your starting point can be cell number zero, and the end cell the last cell, number H.

A large maze takes some time to generate, but there is no need to wait while later mazes are prepared. Once a map of the maze has been displayed, the maze stored in memory is no longer needed. With some re-writing, a new maze can be created while you are finding your way through the one shown on the VDU.

If you have no graphics the maze can still be explored, although it is rather difficult "blind". Each direction of movement M(0) to M(3) is associated with movement through a particular wall W(0) to W(3). If the cell value is divisible by the array value W(A) a wall is present in that direction and movement that way is not possible. Otherwise, with no wall, the cell number is incremented by the array value M(A) and the new cell is entered. Checks for walls are used several times in the program — see lines 105, 120 and 160.

The perspective view is calculated by the subroutine 70 to 110, which also calls the drawing subroutine 10 to 60.

The first cell to be drawn is the one in which the player is currently standing. The left-hand side of the cell is examined first to see if there is a wall present and, if there is, it is drawn — line 100. If there is no wall you will be able to see into the cell on the left. Lines 110, 120 calculate the left-hand cell number and examine that cell for a back wall, such a wall then being drawn if necessary.

The same method is used to show the right-hand side of the cell, but with a few variables changed. There is thus a loop from 80 to 140 which is run through twice, first drawing the left side and then the right side. Line 140 shifts the corners in, ready for the back wall or the next cell to be drawn.

Once both side walls have been drawn, line 160 checks for a back wall. If no back wall is present, line 170 changes the cell number 0P to that of the next cell in the direction we are looking. This same line also increments the depth-of-cell count DC and checks that this does not exceed the greatest possible cell depth CD.

If at any time a back wall to a cell is found, lines 200 and 210 draw this back wall, and the subroutine ends. Lines 180 to 190 are a fix to improve the appearance when there is a back wall next to a cell with no back wall. Line 20 in the drawing subroutine is another fix which gives a more solid look to walls seen end on. Figure 6 shows the same view with and without the fixes.

Lines 240 to 440 enable the player to explore the maze. At the start, no view of the maze is shown. After a time TT a torch will be found, and thereafter the view will always be drawn. Line 430 clears the screen and calls the drawing, subroutine. During the initial period of blindness it is possible for the space suit to be damaged, in which case the player may suffer a faster rate of oxygen loss, lines 300, 310. The same fate can overtake anyone foolish enough to blunder into a wall which can be seen.

Four commands are available via Input statements or, alternatively, by keyboard control — lines 250, and 320 to 340.

The player may turn right, left or opposite his present viewing position, but he will remain in the same cell — only the view will alter. Movement is always forwards — in the direction the player is currently facing. The commands are abbreviated to R, L, O and M respectively.

The exit cell has no view beyond it if you are looking directly at it. Once you arrive at the exit cell you automatically escape.

Figure 7 lists the graphics symbols used to draw the perspective view. You should make appropriate substitutions in lines 70, 80, 90 and 960.

With our limited memory we were unable to add all the embellishments we

(continued on next page)

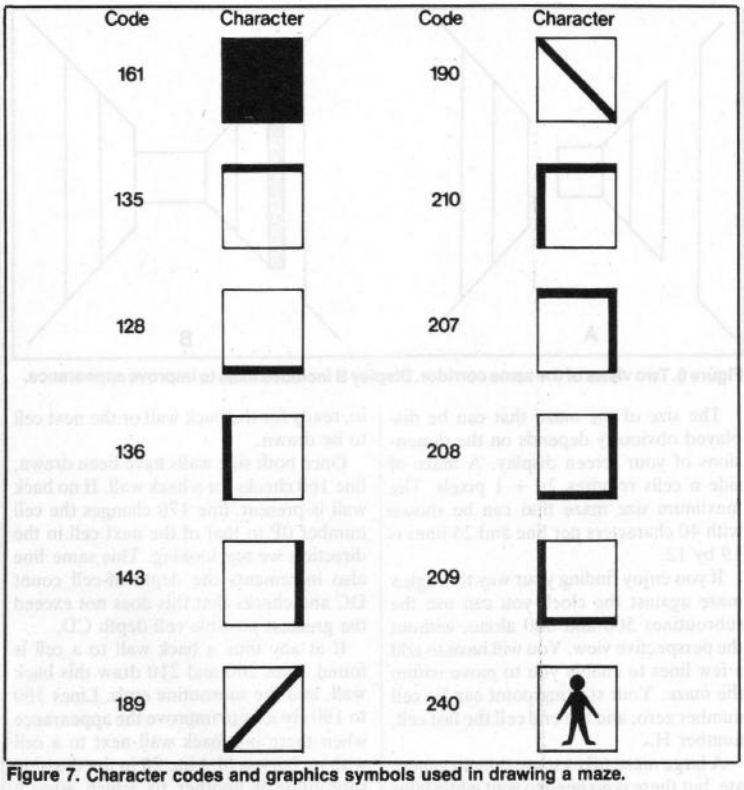


Figure 7. Character codes and graphics symbols used in drawing a maze.

(continued from previous page)

would have liked, but here are a few suggestions to improve the basic game:

- Add a monster which pursues the player through the maze. The monster can move at random initially, but if the player does not escape within a certain time limit, it should begin to move faster and more purposefully as it detects an intruder in its lair. Some cells might contain useful objects — if the creature is radioactive its direction could be indicated by a "geiger" compass, and the intensity of radiation could indicate its distance from the explorer.
- One cell could contain a blaster with a limited charge. The blaster could be fired at the monster in a moment of desperation, or it could be powerful enough to demolish one wall of the maze. The demolition is easily accomplished by dividing the cell value by the array value W(A) in the corect direction. A cunning explorer could blast his way out of the maze if he correctly divined that he was near an outer wall.
- Further traps could be set for the unwary. The map drawn by subroutine 600 might be displayed upside down or reversed; minions of the monster could seize an unlucky explorer and delay his escape; some cells in the maze might be entrances to even more dangerous areas.

It is very easy, when entering a program as long as this one, to make a mistake which becomes difficult to identify. We suggest that you enter this program in sections so that each part can be run to ensure that it is correct. Initially, enter subroutines 500 and 900, and add this line:

```
1029 FQR A= SM TO EM: B=PEEK(A):
 PRINTB;: NEXTA
```

A series of numbers representing the maze will be printed out: all such numbers should be members of the set shown in figure 3.

Next, add subroutine 600 and the line:

```
380 GOSUB 600
```

The maze should be drawn with an exit somewhere along the top. The remainder of the program can now be entered and run.

If you find yourself hopelessly confused and lost after only a few moves, we suggest that you display the map drawn in subroutine 600 at the start, so that you can try to memorise the route out. Line 435 gives the player one look at the map only — it is presumed to be displayed on a VDU which burns out after this brief use. The map can be examined many times if line 435 becomes:

```
IFP=MC THEN GOSUB 600
```

Finally, those with good graphics facilities should be able to draw objects on the floor of a cell. They can be given the correct relative size by referring to the current corner values for the cell, M(4) to M(7). It certainly adds atmosphere if an explorer stumbles across the occasional skull or rib-cage.

```
5 GOTO900
10 V=PEEK(TC-U):Z=PEEK(TC-U+LL):IFABS(U)=1ORDC=0THEN30
20 IFV=32ANDZ=32THENFORA=TC-U+LLTOBC-V-LLSTEPLL:POKEA,W3:NEXT
30 POKETC,TW:POKEBC,BW:B=B+1:IFB<DWTHENTC=TC+U:BC=BC+V:GOTO30
40 IFABS(U)=1THENPOKETC,T1:POKEBC,B1
50 IFTC=BC-LLTHENRETURN
60 FORB=TC+LLTOBC-LLSTEPLL:POKEB,SW:NEXT:RETURN
70 F=ML:U=LL+1:V=1-LL:T1=207:B1=208:SW=143
80 FORX=4TO6STEP2:TW=190:BW=189:B=0:IFX=4THEN100
90 F=MR:U=LL-1:V=-LL-1:SW=136:T1=210:B1=209:TW=189:BW=190
100 LM=M(F):LD=W(F)
101 IFF=3ANDOP=ECTHEN140
105 IFQ-LD*INT(Q/LD)=0THENTC=M(X):BC=M(X+1):GOSUB10:GOTO130
110 TW=135:BW=128:U=U-LL:V=V+LL:N=PEEK(OP+LM)
120 IFN-D*INT(N/D)=0THENTC=M(X)+DW*LL:BC=M(X+1)+DW*-LL:GOSUB10
130 IFABS(U)=1THENU=U+LL:V=V-LL
140 M(X)=M(X)+DW*U:M(X+1)=M(X+1)+DW*V:NEXT
150 IFM=3ANDOP=ECTHENRETURN
160 IFQ-D*INT(Q/D)=0THEN180
170 OP=OP+M(M):Q=PEEK(OP):DC=DC+1:IFDC<CDTHEN70
175 IFDC=CDTHENRETURN
180 R=PEEK(M(4)-1)
185 IFR=32THENM(4)=M(4)-1:M(5)=M(5)-1:IFM(4)<>TL+DW*LLTHEN180
190 R=PEEK(M(6)+1)
195 IFR=32THENM(6)=M(6)+1:M(7)=M(7)+1:IFM(6)<>TR+DW*LLTHEN190
200 FORTC=M(4)TOM(6):POKETC,W1:NEXT
210 FORBC=M(5)TOM(7):POKEBC,W2:NEXT:RETURN
240 PRINTOX;" UNITS OF AIR"
245 PRINT"ACTION":INPUTA$
250 NP=P:Q=PEEK(P):IFA$="M"THENNP=P+M(M)
260 IFNP=PTHEN320
270 IFNP=ECTHENPRINT"YOU ESCAPE!":END
285 IFQ-D*INT(Q/D)<>0THENP=NP:PRINT"YOU MOVE":GOTO380
300 IFRND(1)>.9THENPRINT"YOUR SUIT RIPS WHEN ";:OL=OL+1
310 PRINT"YOU HIT A WALL":GOSUB3000:GOTO410
320 NM=M:IFA$="R"THENNM=M+1
330 IFA$="Q"THENNM=M+2
340 IFA$="L"THENNM=M+3
350 IFNM=MTHEN240
360 IFNM>3THENNM=NM-4*INT(NM/4)
370 M=NM:PRINT"YOU TURN."
380 D=W(M):Q=PEEK(P):OP=P:ML=M-1:MR=M+1
390 IFML<0THENML=3
400 IFMR>3THENMR=0
410 OX=OX-OL:IFOX<0THENPRINT"YOU DIE.":END
420 T=T+1:IFT=TTTHENTT=1:PRINT"YOU FIND A TORCH":GOSUB3000
430 IFTT=1THENDC=0:GOSUB2000:GOSUB70
435 IFP=MCANDRM=0THENRM=1:GOSUB600
440 M(4)=TL:M(5)=BL:M(6)=TR:M(7)=BR:GOTO240
500 P=P+1:IFP>EMTHENP=SM
510 A=INT(RND(1)*4):DC=0
520 A=A+1:DC=DC+1:IFDC>3THEN500
530 IFA>3THENA=0
540 M=P+M(A):IFM<SMORM>EMTHEN520
550 CP=PEEK(P):CM=PEEK(M):IFC>0ANDCP=210THENP=M:GOTO510
560 TM=M-SM:IF(CP=CMORCM<210)ANDC>0THEN520
570 ME=TM-L*INT(TM/L):IF(ME=0ANDM(A)=1)OR(ME=LANDM(A)=-1)THEN520
580 OD=INT(15/W(A)):CP=CP/W(A):POKEP,CP:CM=CM/OD:POKEM,CM
590 P=M:C=C+1:IFC<HTHEN510
595 RETURN
600 GOSUB2000:PRINT"YOU FIND A MAP"
605 A=TL:MS=A:DC=SM:DD=SM+6
610 FORB=DCTODD:FORC=0TO3:CB=PEEK(B):CB=CB-W(C)*INT(CB/W(C))
620 AA=1:IFC=1ORC=3THENAA=LL
630 IFC>1THENAA=-AA
640 BB=LL/AA:P2=A+AA:P1=P2+BB:P3=P2-BB
645 IFB=PTHENPOKEA,W4
650 ECB=0THENPOKEP1,W3:POKEP2,W3:POKEP3,W3
660 NEXTC:A=A+2:NEXTB:DC=DC+L:DD=DD+L:A=MS+(2*LL):MS=A
670 IFDD<=EMTHEN610
680 GOSUB3000:RETURN
900 GOSUB2000:PRINT"YOU HIDE IN AN ALIEN MAZE TO ESCAPE ENEMIES."
910 PRINT"FIND THE EXIT BEFORE YOUR AIR RUNS OUT."
950 TL=53349:BL=54149:TR=53374:BR=54174:LL=32:DW=3:CD=4
955 W1=135:W2=128:W3=161:W4=240
960 W(0)=5:W(1)=7:W(2)=3:W(3)=2:L=INT(RND(1)*5+6)
970 M(0)=1:M(1)=L:M(2)=-1:M(3)=-L:M(4)=TL:M(5)=BL:M(6)=TR:M(7)=BR
980 W=INT(RND(1)*5+6):H=L*W-1:G=L-1:C=0:DC=0:T=0:RM=0
995 SM=576:EM=SM+H:FORA=SMTOEM:POKEA,210:NEXT:P=SM+INT(RND(1)*H)
1000 EC=SM+INT(RND(1)*L):MC=INT(RND(1)*H)+SM
1010 OX=INT(H/3)+1:IFP<OXTHENOX=-OX
1015 OX=OX+H:OL=1:TT=INT(RND(1)*H/8)+1:GOSUB500:GOSUB2000
1030 M=0:NP=P:CE=PEEK(EC):CE=CE/W(3):POKEEC,CE:GOTO380
2000 FORX=0TO25:PRINT:NEXTX:RETURN
3000 FORX=0TO5000:NEXTX:RETURN
```

SUPPORTED BY YOU ALL AND PRODUCED AT TRS-80.ORG.UK

WHERE 8 BITS AND 16K RULES OK

TRS8BIT

PRODUCED BY TRS-80.ORG.UK

HI THERE EVERYONE AND WELCOME TO THE SEPT. EDITION. ONCE AGAIN, IT'S A 'BUMPER' EDITION AND THERE'S LOTS TO GET THROUGH SO I'LL KEEP MY RAMBLINGS SHORT AND SWEET.

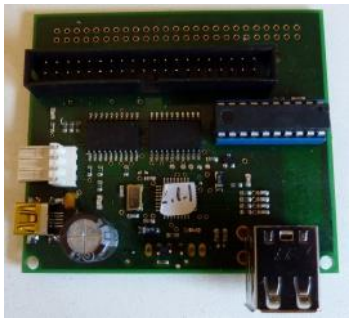
A FEW WEEKS BACK A M2 WENT FOR JUST OVER £200 ON EBAY. TALKING ABOUT M2'S, GAZZA HAS JUST ACQUIRED (FOR SORTING OUT), A WHOLE BUNCH OF THEM. CHECK THE FORUM FOR THE FULL STORY AND PHOTOS.

I HAD A VERY ENJOYABLE EARLY MORNING, LINKING UP TO PETER CETINSKI AND HIS FELLOW TRASH-TALKERS ON SUNDAY 14TH AUGUST 2016. EVEN IF IT WAS AT 02:00HRS!
IT WAS JUST LIKE THE OLD DAYS. IF YOU DON'T BELIEVE ME, CHECK OUT PAGE 28!

HANS RIETVELD HAS BEEN RUNNING A SURVEY ON THE FORUM TO COUNT UP HOW MANY 'LARGE' TANDYS ARE STILL ABOUT. I WAS PLEASANTLY SURPRISED BY THE NUMBER. THEY ARE A BIT ON THE BIG SIDE THOUGH, WHEN IT COMES TO FINDING ROOM FOR THEM!

PETER CETINSKI IS BUSY WORKING ON HIS TRS-BOX SERIAL PROGRAM DROPBOX. FULL DETAILS SHOULD BE IN THE NEXT EDITION.
ALSO, THE NEXT EDITION OF 'TRASH-TALK' IS IMMINENT WHICH HAS AN INTERVIEW WITH SCOTT ADAMS, OF 'ADVENTURE INTERNATIONAL' FAME. CAN'T WAIT!!

REGIS MUNOZ AND PASCAL HOLDRY ARE BUSY BETA TESTING THE 'REGUSB'. FULL DETAILS ARE ON PASCAL'S WEBSITE. THIS DEVICE WILL SUPPORT VARIOUS DISK FORMATS AND ALLOWS YOU TO ACCESS FILES, JUST AS THOUGH THEY WERE ON A PC, VIA THE USB INTERFACE.

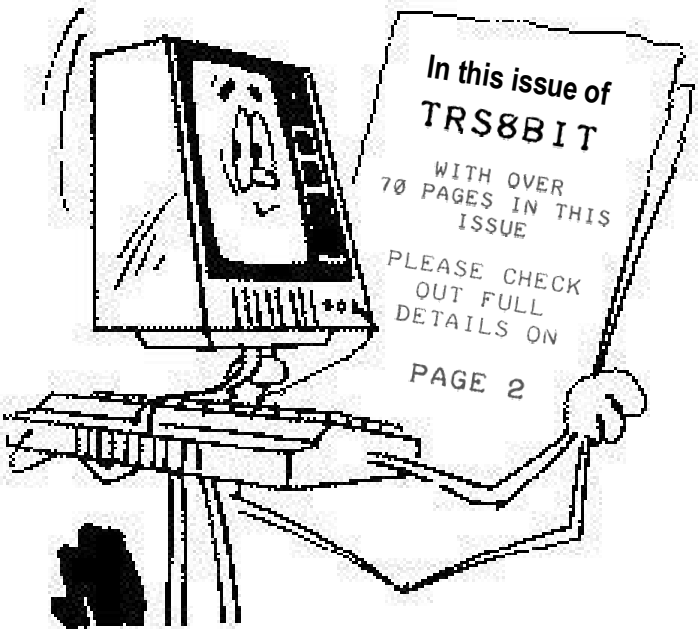


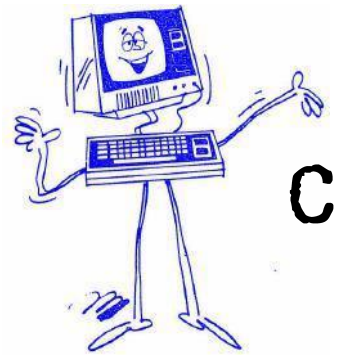
M1 CUSTOM CASES HAVE BEEN IN THE NEWS ON THE FORUM, RECENTLY. DOES ANYONE IN THE UK, HAVE A PHOTO OF THE ORIGINAL 'BLANDFORD BOX'? IT'S JUST A THOUGHT.

I'M RUNNING OUT OF SPACE AND WITH OVER 70 PAGES AGAIN, I HOPE THERE'S SOMETHING OF INTEREST FOR EVERYONE.
'TILL THE DECEMBER (XMAS) ISSUE,

TAKE CARE EVERYONE,

DUSTY





CONTENTS

Pump Up Your TRS-80 with the ES/F Mass Storage System

THESE FACTS SPEAK FOR THEMSELVES!

	CASSETTE	ES/F	MINI-DISK
SPEED (Seconds to load "Blackjack")	56	6 (5" wafer)	6½
CAPACITY (thousands of bytes)	38 (C-20)	64 (75" wafer)	59 (TRSDOS)
RELIABILITY (Designed for digital data?)	NO	YES	YES
SYSTEM COST (First unit plus interface)	$60	$250	$800
MEDIA COST (in quantities of ten)	$3.10 cassette	$3.00 wafer	$3.20 disk

Let's face it. Cassette players were not designed to store digital data and programs. That's why we designed a digital storage system using a continuous tape loop: the Exatron *Stringy/Floppy* (ES/F) and the Wafer. There's no expensive interface to buy—the ES/F comes ready to pump up your TRS-80.*

Once your TRS-80* is pumped up by our ES/F . . . you won't want to deflate it. We're so sure, that we offer an unconditional 30-day money-back guarantee and a one-year limited warranty. Over 2,000 TRS-80* owners have met the wafer . . . why don't you?

TRS-80*

EXATRON'S STRINGY/FLOPPY...

SPEED, CAPACITY AND RELIABILITY FOR ONLY $249.50

CALL
OUR HOTLINE
(800)-538-8559

IN CALIFORNIA,
CALL (408)-737-7111

exatron, inc.
181 Commercial Street
Sunnyvale, Calif. 94086

CIRCLE 270 ON READER SERVICE CARD

*TRS-80 is a registered trademark of Tandy Corp.

An Atari Joystick Adaptor for the TRS-80

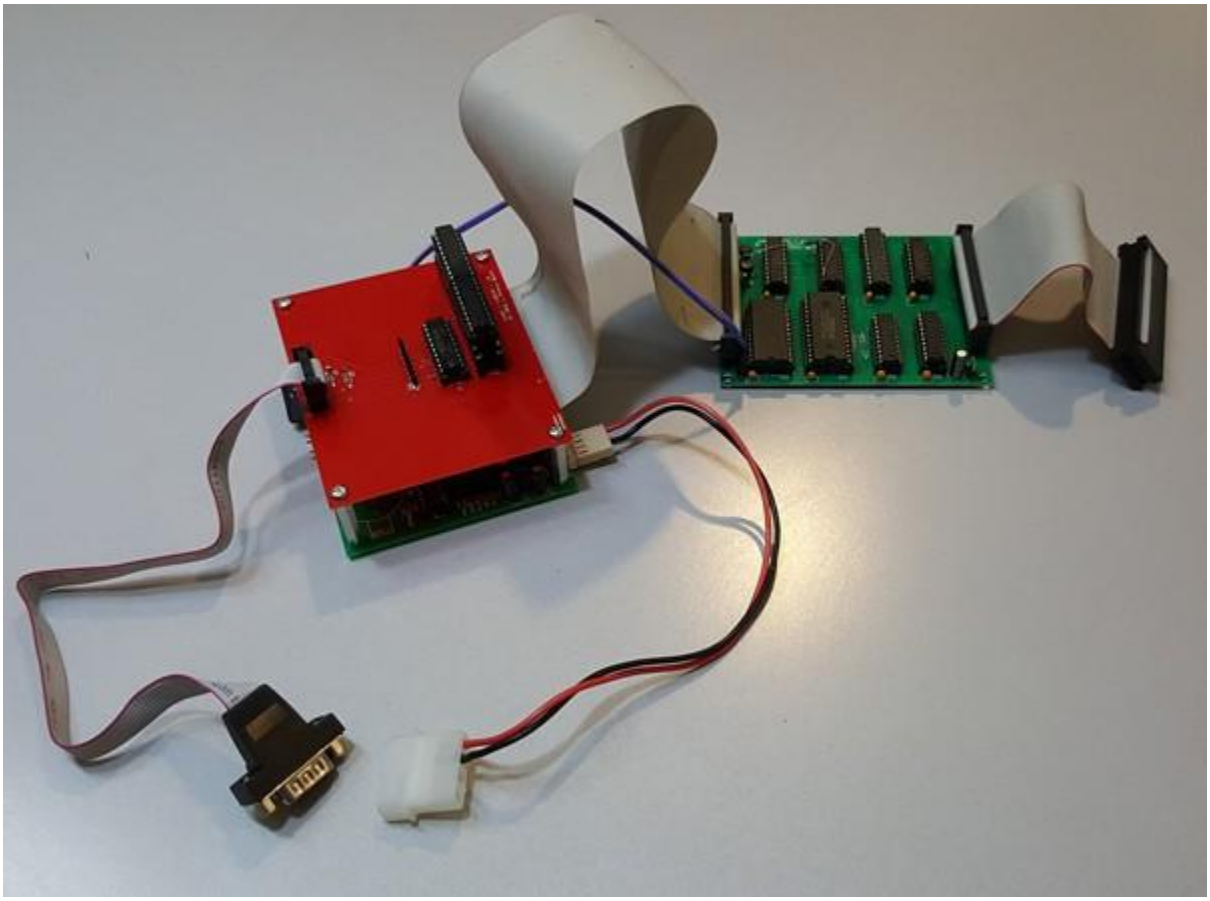
Andrew Quinn

This project was prompted by a question on the VC Forum earlier in 2016…. Can I build an Alpha type joystick adaptor and use it with my TRS-80 and FreHD?

The answer unfortunately is no if the Alpha joystick adaptor design is really like the hand drawn schematic that I found on the web… it does no IO Port decoding so will respond to a read request on any IO Port and conflict with the FreHD.

Given the simplicity of an Alpha style joystick (it is just a set of switches that pull selected data lines to ground), it did get me thinking that building a companion adaptor board for the FreHD should be possible… or at least worth an attempt.

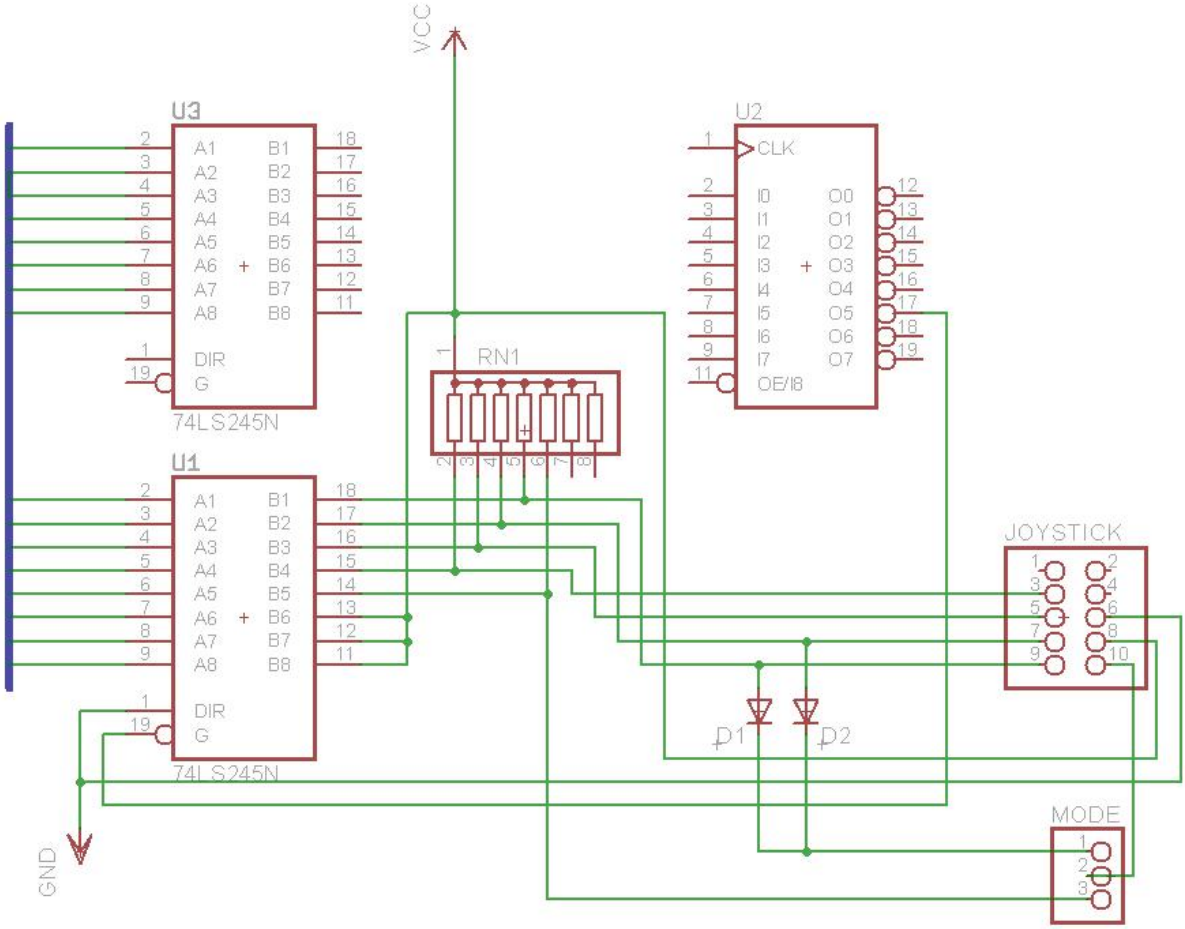
The result is the FrePLAY adaptor, shown here with a FreHD and Model 1 Expander (as Ian calls it… the "Quinnterface").



The FrePLAY adaptor is a companion board that mounts above the FreHD. It can be used when the FreHD is connected to a Model I, III, 4 or 4P.

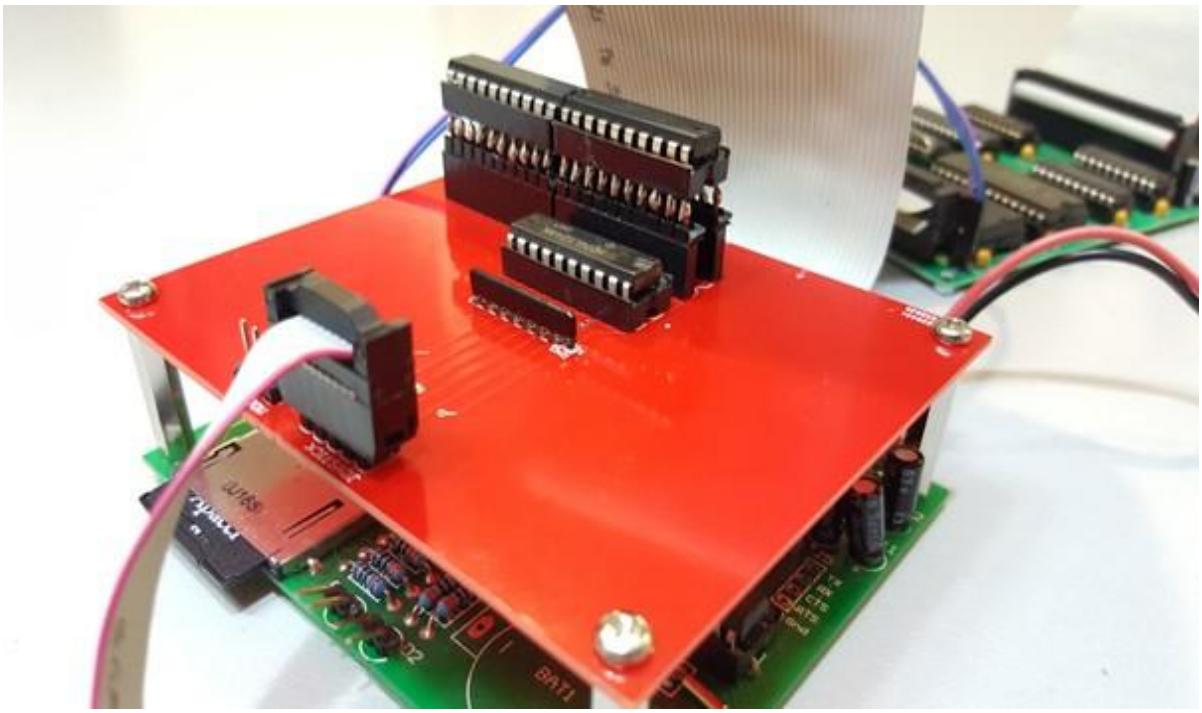
Connections to the TRS-80 bus are made by removing U2 (16V8 GAL) and U3 (74LS245) from the FreHD and inserting the FrePLAY header pins into their sockets to join the two boards. 25mm spacers are used to ensure a solid connection between the boards.

U2 is programmed with a modified equation files that maintain full FreHD compatibility but also decode the Joystick IO ports. U2 and U3 are then inserted into the FrePLAY board.



The design takes advantage of one of the unused output pins (pin 17) on the FreHD U2 GAL. This is used to enable U1 (74LS245) which gates the joystick port value onto the TRS-80 data bus when IO Ports 0x00 or 0x13 are read. All data lines are pulled high by resistor network RN1 and the joystick pulls these low when moved. The Mode header allows a jumper or switch to be installed for selection of the 5 bit (4 directions and fire) or 4 bit (4 directions with fire pressing up and down at the same time) joystick modes.

The version 1 board shown in the pictures uses Arduino style 10 pin stacking headers to join the two boards.
The great thing about them is they are cheap and easily available on eBay. The bad thing about them is that despite my initial positive test results they really don't work reliably as an IC socket for U2 and U3 - hence the ugly socket extensions you can see in the photo below.



The version 2 board has a better solution. It is a slightly modified design that uses proper IC sockets on the top of the board offset from the pin headers that protrude from the bottom of the board.
Arduino headers connect to these and to the FreHD.

I will provide more information on the V2 board at http://www.quicktrip.co.nz/jaqblog when they arrive.

Andrew Quinn
jaquinn@ihug.co.nz

Andrew Quinn is a Software Development Manager who also collects and restores TRS-80's and other vintage computers. He lives in Auckland, New Zealand.

www.trs-80.org.uk
2016 Competition

Write a one-line BASIC program, on any theme (game, utility etc.) you wish.
The whole program, including any assembler routines if present, must be contained within one line of code and run on any TRS-80 computer.
The program must be self-contained within the line; i.e. no pre-loading or linking to another program in memory before running.

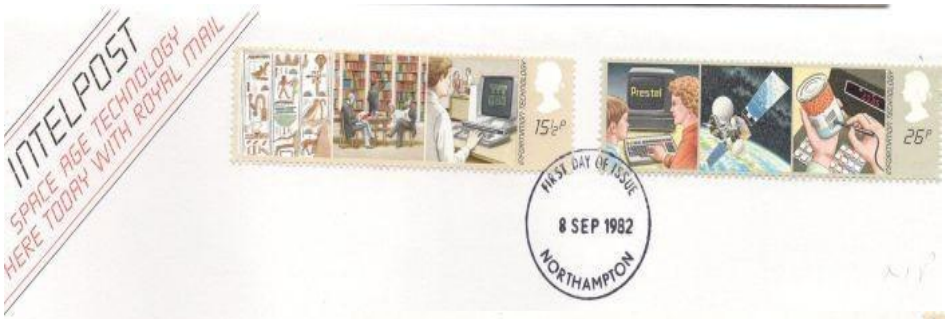
Dee will award the prize to the person(s) who, in her opinion, sends in the most original, amusing or just brilliant program.

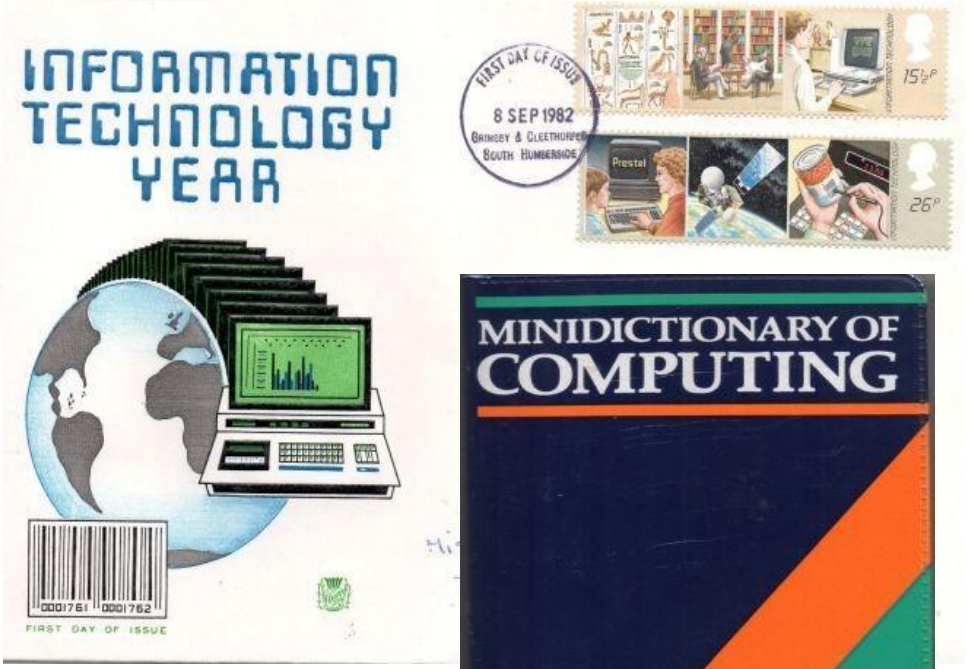
And the prize :- An original 1982 U.K. Royal Mail issued, set of postage stamps celebrating 'Information Technology Year', comprising, of both the 1st day issue covers, and a set of 'mint' stamps.
Also included is an Oxford mini computing dictionary from the same era.

For full details, please see the website

www.trs-80.org.uk

THE 2016 COMPETITION PRIZE





TRUETYPE FONTS
FOR YOUR
TANDY

HERE'S A SUPER LITTLE TREAT FOR YOU ALL. HAVE YOU EVER FANCIED HAVE A M1, 3, 4 OR EVEN A CO-CO TRUETYPE FONT FOR YOUR PC?

WELL, THANKS TO THE GENEROSITY OF KREATIVEKORP AT

HTTP://WWW.KREATIVEKORP.COM/SOFTWARE/FONTS/TRS80.SHTML

THERE IS A .ZIP FILE AVAILABLE FOR DOWNLOADING.
I JUST LOVE THOSE 'JUMPING' LOWERCASE g's j's and y's !

The Ultimate TRS-80 Font

Download the world's most complete TRS-80 text font set, in TrueType format. Works on Mac OS X, Windows, or any operating system that supports TrueType.

The "Another Man's Treasure" series consists of 22 fonts - the largest of any series in my Retro Computing Fonts collection. This is because there were so many models of the TRS-80, each with its own character set, in one of two video modes, or four in the case of the Model 4!

Model	Revision	Aspect Ratio	Font Name	Sample Text
I	Original	64-Column (2:3)	Another Mans Treasure MIA 64C 2X3Y	Another Mans Treasure MIA 64C 2X3Y The Quick brown fox JumPed c
		32-Column (4:3)	Another Mans Treasure MIA 32C 4X3Y	Another Mans Treasure MIA 32C 4
		2Y (1:2)	Another Mans Treasure MIA 2Y	Another Mans Treasure MIA 2Y The Quick brown fox JumPed over the lazy dogs.
		Raw (1:1)	Another Mans Treasure MIA Raw	Another Mans Treasure MIA Raw The Quick br
	Lowercase	64-Column (2:3)	Another Mans Treasure MIB 64C 2X3Y	Another Mans Treasure MIB 64C 2X3Y The quick brown fox jumped c
		32-Column (4:3)	Another Mans Treasure MIB 32C 4X3Y	Another Mans Treasure MIB 32C 4
		2Y (1:2)	Another Mans Treasure MIB 2Y	Another Mans Treasure MIB 2Y The quick brown fox jumped over the lazy dogs.
		Raw (1:1)	Another Mans Treasure MIB Raw	Another Mans Treasure MIB Raw The quick br
III	(Any)	64-Column/2Y (1:2)	Another Mans Treasure MIII 64C	Another Mans Treasure MIII 64C The quick brown fox jumped over
		32-Column/Raw (1:1)	Another Mans Treasure MIII 32C	Another Mans Treasure MIII 32C
4	Original	64-Column (5:8, 12-Scan)	Another Mans Treasure M4A 64C	Another Mans Treasure M4A 64C The quick brown fox
		32-Column (5:4, 12-Scan)	Another Mans Treasure M4A 32C	Another Mans Treasure M4A
		80-Column (1:2, 10-Scan)	Another Mans Treasure M4A 80C	Another Mans Treasure M4A 80C The quick brown fox ju
		40-Column (1:1, 10-Scan)	Another Mans Treasure M4A 40C	Another Mans Treasure M4A
		2Y (1:2, 12-Scan)	Another Mans Treasure M4A 2Y	Another Mans Treasure M4A 2Y The quick brown fox jumped over th
		Raw (1:1, 12-Scan)	Another Mans Treasure M4A Raw	Another Mans Treasure M4A Raw Th
	International	64-Column (5:8, 12-Scan)	Another Mans Treasure M4B 64C	Another Mans Treasure M4B 64C The quick brown fox
		32-Column (5:4, 12-Scan)	Another Mans Treasure M4B 32C	Another Mans Treasure M4B
		80-Column (1:2, 10-Scan)	Another Mans Treasure M4B 80C	Another Mans Treasure M4B 80C The quick brown fox ju
		40-Column (1:1, 10-Scan)	Another Mans Treasure M4B 40C	Another Mans Treasure M4B
		2Y (1:2, 12-Scan)	Another Mans Treasure M4B 2Y	Another Mans Treasure M4B 2Y The quick brown fox jumped over th
		Raw (1:1, 12-Scan)	Another Mans Treasure M4B Raw	Another Mans Treasure M4B Raw Th

SPECIAL FEATURE

Pascal Presents ...

I though you would all be interested with just some of the re-engineering work that Pascal Holdry, from France, has been so busy with over the recent months.
It's quite remarkable just what he's been doing and well worth a Special Feature just so you can see the amount of effort he's putting in.

He's asking if someone out there has a M1 CP/M card which he could borrow and also he is looking for photos of "PCBs Voice synthesizer 26-1180" and the dimensions of the "PCB" to redo the prototype on a real "PCB"

Please contact him direct if you can help.

Keyboard (Keyboard) AZERTY / QWERTY 4P

Right out of the TRS-80 4P several keyboard types were proposed

1. The QWERTY (English)
2. The type QWERTZ (German)
   The type AZERTY (French)
   Although its home page is in English or French or German, it was not very sold in France.Thereby:

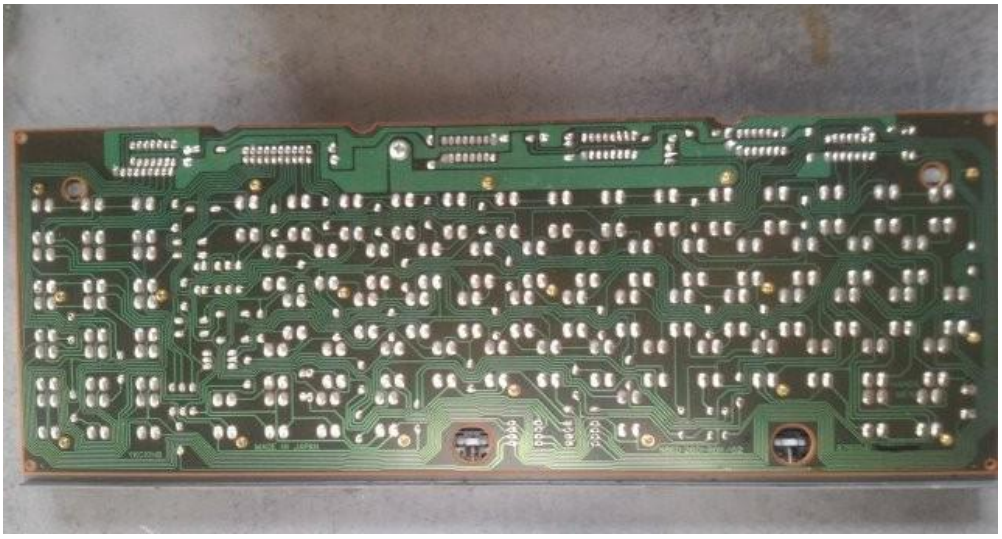
1. The majority of softwares of this machine was designed for a QWERTY keyboard
2. The passage of the QWERTY keyboard in QWERTY by hardware with an electronic addition
3. It takes a special TRS-DOS QWERTY 6.01.1F (F for French)
4. Some software in English no longer functioning or only work poorly
5. Some extensions (graphics cards) do not work with this version of TRS-DOS
   The updated version of DOS in French stopped very quickly (around 1986), while the last update DOS English in 1991 LS-DOS 6.31H

View of the "PCB" QWERTY keyboard (Eric Photo)

Bottom view of the "PCB" QWERTY keyboard (Eric Photo)

View of the "PCB" QWERTY keyboard, you can see the addition of electronic (Eric Photo)

View of the "PCB" QWERTY keyboard, you can see the addition of electronic (Eric Photo)



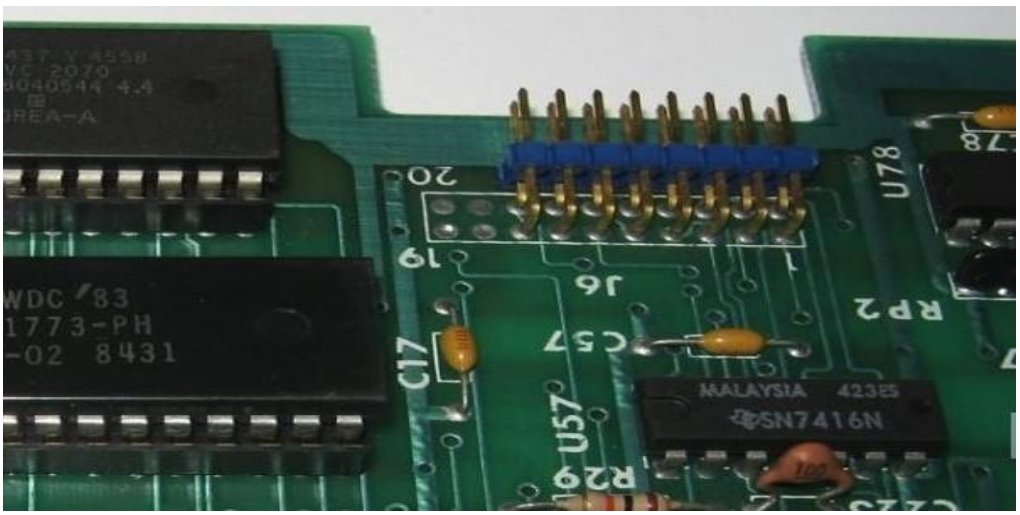
The connectors between the two keyboards are not identical

1. 16-pin connector for the QWERTY keyboard

20-pin connector for the QWERTY keyboard

View of the two connectors keyboards (QWERTY and AZERTY) (Eric Photo)

View of the QWERTY keyboard connection (Eric Photo)



For connecting the QWERTY keyboard (Eric Photo)

It takes a 20-pin connector for the QWERTY keyboard to power the electronics + 5V and GND.
My information:

1. We must cut the extra 4 pins for connecting the QWERTY keyboard, otherwise it is not possible to insert the connector of the qwerty keyboard.
   It is not enough that align the pin No. 1 male and female connectors

---

“Doubleur_NEW_PERC”

Regis Munoz and myself have built a card prototype based on the operating principle of the reference card for the doubler Density "The PERCOM II"

Goal:

To retrieve existing softwares
Keep track of our digital heritage
Provide a diagram for troubleshooting and understanding the operation of the card.

Specifications:

Be compatible with a maximum PERCOM II card
Function without setting (plug and play)
Delete the operation of the card with a "PROM" found today.
Use easy supply components.
Use components of a low price.
Keep in mind the 70/80

Use dimensions for use in an MDX-2 Expansion Board

Banish all the exotic components

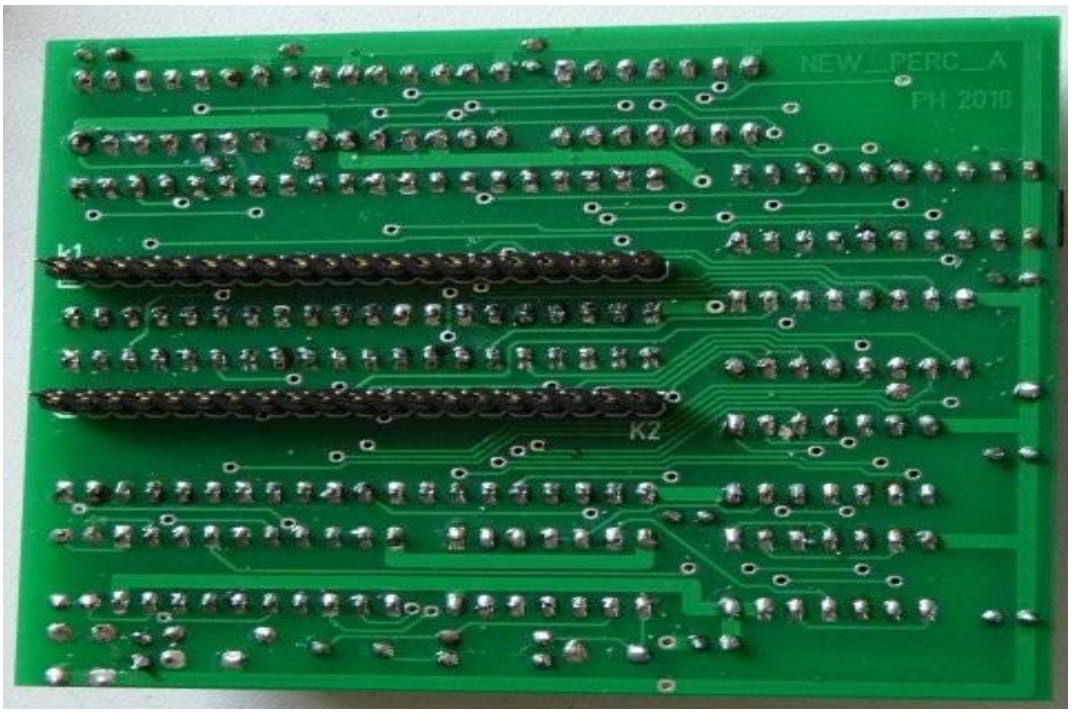
The program components will be programmable timer with a very low range. ... etc

For this we used "GAL16V8" very easy supplies, programmable controllers with a few tens of euros. We excluded "PAL" and "PROM" that can be programmed only one time. The "GAL" X times are programmable and therefore recoverable on old electronic cards.

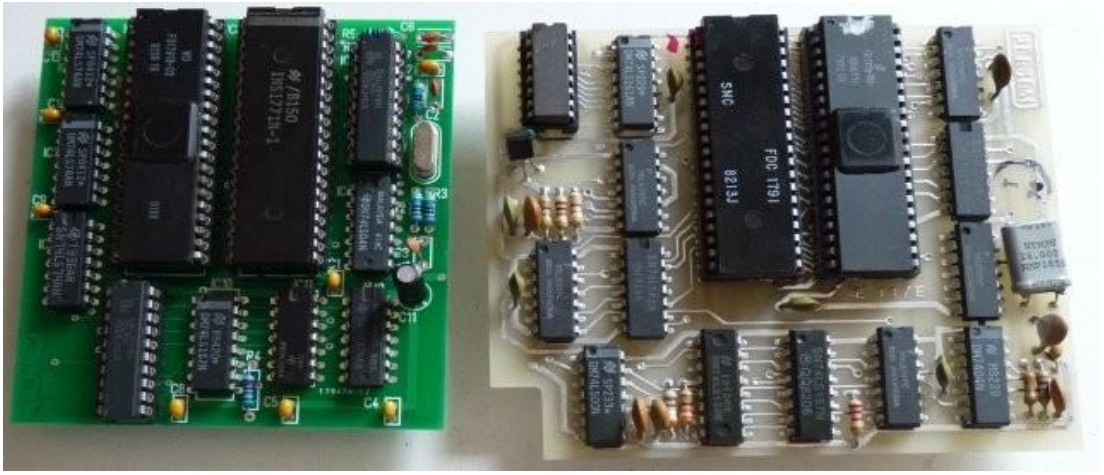
I am responsible for the "hard" part of Regis and the "Soft"

Top view

Bottom View



Comparison with a doubler PERCOM



Equations for the LAG IC4

Designer Regis MUNOZ;
Company University of Tulsa;
Assembly None;
None Location;
Device g16v8a;
/ ****************************** ************** * /
/ * * /
/ * GAL for Doubler_New_Perc_A card * /
/ * ------- ----------- Regis MUNOZ ---- * /
/ * * /
/ * LAG # 2: DECODE & LOGIC - IC4 * /
/ * * /
/ ********************** ************************ * /
/ PINS INPUTS *************** ******* *********** /
Pin 1 = in_r5c7; / * Input entre R5 / C7 * /
Pin 2 = in_r6c6; / * Input entre R6 / C6 * /
Pin 3 = wd; / * Input signal of WD-31 PIN of IC2 - FD1791 * /
Pin 4 = late; / * Input LATE signal of PIN-FD1791 * 18 /
Pin 5 = early; / * Input EARLY signal of PIN-FD1791 * 17 /
Pin 6 = a0; / * Input signal * A0 /
Pin 7 = a1; / * Input signal A1 * /
Pin 8 = dal3; / * Input signal DAL3 * /
Pin 9 = dal4; / * Input signal DAL4 * /
Pin 11 = DAL5; / * Input signal DAL5 * /
Pin 12 = dal6; / * Input signal DAL6 * /
Pin 13 = dal7; / * Input signal DAL7 * /
Pin 14 = we; / * Input signal WE * /
Pin 17 = in_r1c2; / * Input entre R1 / C2 * /
/ OUTPUT PINS *************** *************** /
Pin 15 = out_ic1_3; / * PIN-3TB * of IC1 /
Pin 16 = out_r1; / * * To R1 /
Pin 18 = dwd; / * PIN-10TB * of IC10 /
Pin 19 = out_r5r6; / * To R5 / R6 * /
/ ** ** Logic Equations /
out_r5r6 = wd;
dwd = (in_r6c6 & early & late!) # (& In_r5c7 early & late!) # (Wd

```
& early & late!);
out_r1 = we;!
out_ic1_3 = (& in_r1c2 (we dal3 #) & (# dal6 dal7) & (# dal4 DAL5) &!!!!
(a0 a1 #!));
/ * ******** ************ ********************** END /
```

Equations for the LAG IC9

```
Name NewPercA;
partNo NEWPERCA-RM;
20/06/16 Date,
Revision 05;
Designer Regis MUNOZ;
Company University of Tulsa;
Assembly None;
None Location;
Device g16v8a;

/ ************ ********************************* * /
/ * * /
/ * GAL for Doubler_New_Perc_A card * /
/ * - --------- ----------- Regis MUNOZ * /
/ * * /
/ * LAG # 1: CLOCK GENERATOR - IC9 * /
/ * * /
/ ***** ***************************************** * /

/ ******* ****************** ******** INPUTS PINS /
Pin 1 = clk; / * Input Clock 4 / 8MHz * /
Pin 2 = LD; / * PIN-15TB * of this GAL /
Pin 3 = IC5_9; / * To-PIN 9 of IC5 * /
Pin 4 = IC5_8; / * PIN-8TB * of IC5 /
Pin 5 = IC7_8; / * PIN-8TB * of IC7 /
Pin 6 = IC7_5; / * To PIN-5 of IC7 * /
Pin 7 = IC7_9; / * To-PIN 9 of IC7 * /
Pin 8 = WF_VFOE; / * To 4.7k resistor and signal WF / VFOE # of IC2 * /
Pin 11 = ow!; / * To GND: Register output enable * /

/ OUTPUT PINS ************** *************** /
Pin 12 = RCLK; / * Output signal RCLK * /
Pin 13 = FDDATA; / * Output signal FDDATA * /
Pin 14 = FDCLK; / * Output signal FDCLK * /
Pin 15 = OLD; / * To PIN-2 of this LAG * /
Pin 16 = Q3; / * To PIN-11 of IC5 (input clock) * /
Pin 17 = Q2; / * NC (do not connect) * /
Pin 18 = Q1; / * NC (do not connect) * /
Pin 19 = Q 0; / * NC (do not connect) * /

/ ** Declarations ** Definitions and Intermediate Variable /
Field count = [Q3..0]; / * Table of counter inputs * /
Field prom = [K3..0]; / * Outputs of counter table * /

TABLE count => {prom
'to 0 =>' h -1; / * 0001 * /
'1 =>' h -1; / * 0001 * /
```

```
'to 2 =>' h'2; / * 0010 * /
'to 3 =>' H'3; / * 0011 * /
'to 4 =>' H'3; / * 0011 * /
'to 5 =>' h'4; / * 0100 * /
'to 6 =>' h'4; / * 0100 * /
'to 7 =>' h'4; / * 0100 * /
'to 8 =>' h'E; / * 1110 * /
'to 9 =>' h'E; / * 1110 * /
'of 10 =>' h'E; / * 1110 * /
'of 11 =>' h'E; / * 1110 * /
'of 12 =>' h'F; / * 1111 * /
'of 13 =>' h'F; / * 1111 * /
'of 14 =>' H'0; / * 0000 * /
'of 15 =>' h -1; / * 0001 * /
}

/ ** ** Logic Equations /
OLD = (IC7_5 & IC7_9)!
FDDATA = (# IC5_9 IC7_8)!
FDCLK = (# IC5_8 IC7_8)!
RCLK = (# IC5_9 WF_VFOE!);

/ * Data Inputs of counter (clock) * /
Q0.d = # (LD & Q0!) (LD & K 0!)
Q1.d = # (LD & Q1 & Q0!) # ((LD & K1!) LD & Q1 & Q0)!
Q2.d = (LD & K2) # (LD & Q2 & Q1) # (LD & Q2 & Q0) # (LD & Q2 & Q1 & Q0)!!!!
Q3. d = # (LD & Q2 Q3 &!) # (LD & Q1 & Q3!) # (LD & Q3 & Q0!) # (& LD & Q3 Q2 Q1 & Q0 &!) (LD &
K3!);
```

The schematic, Gerbers and .JED file as detailed are available for free from the Downloads menu on my website

http://www.prof-80.fr/

M1 Keyboard Printer Interface Prototype

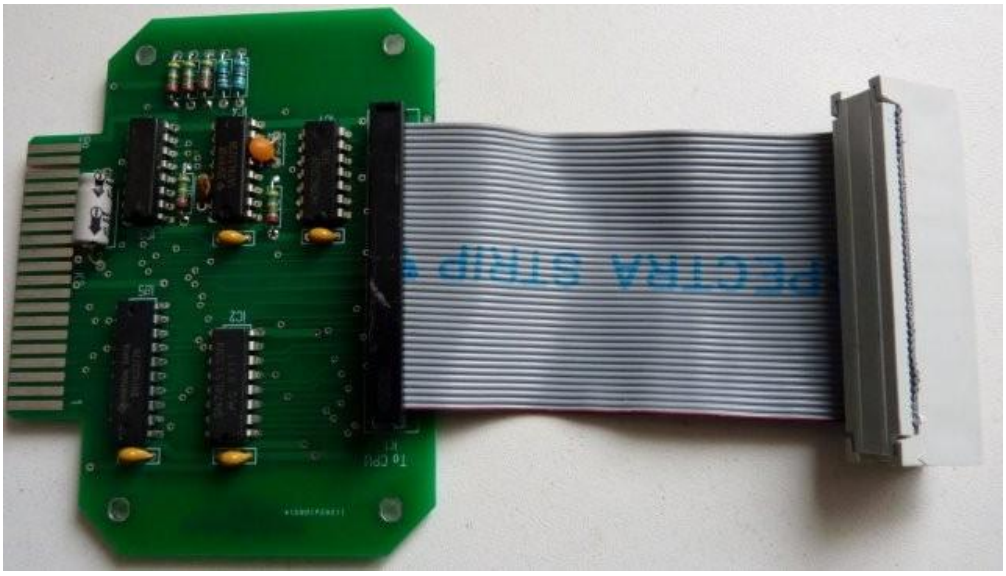
I built a prototype board based on the principle of the Tandy parallel interface

Goal:

1. The opportunity to work with a printer
2. Keep track of our digital heritage

Provide a diagram for troubleshooting and understanding the operation of the card.

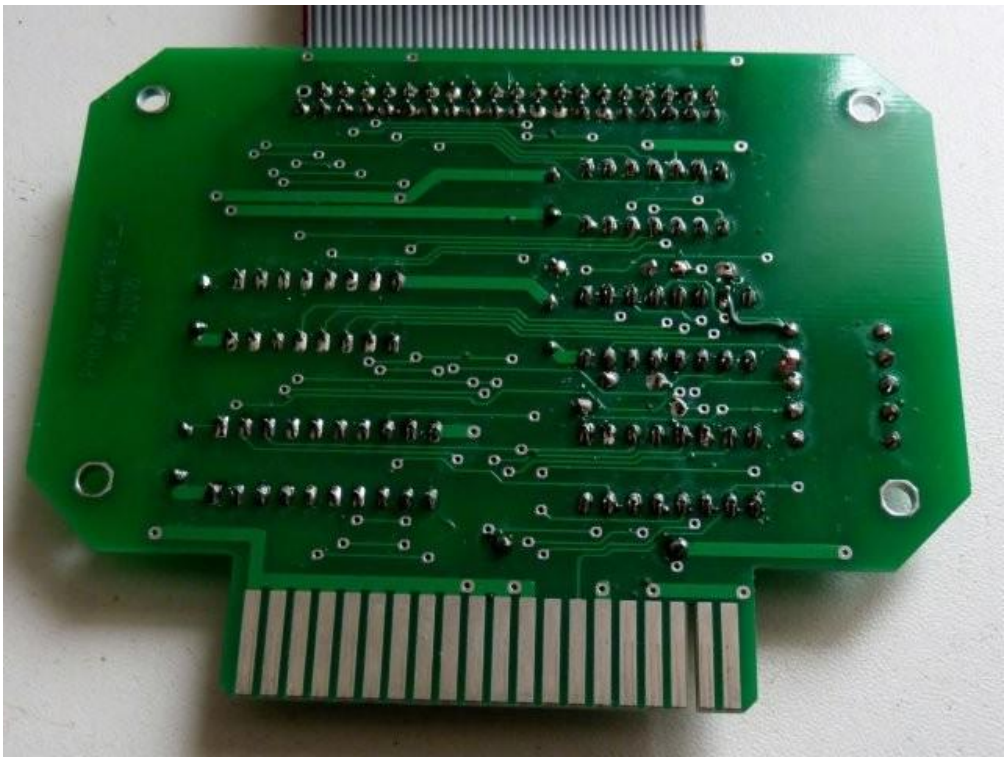
View all



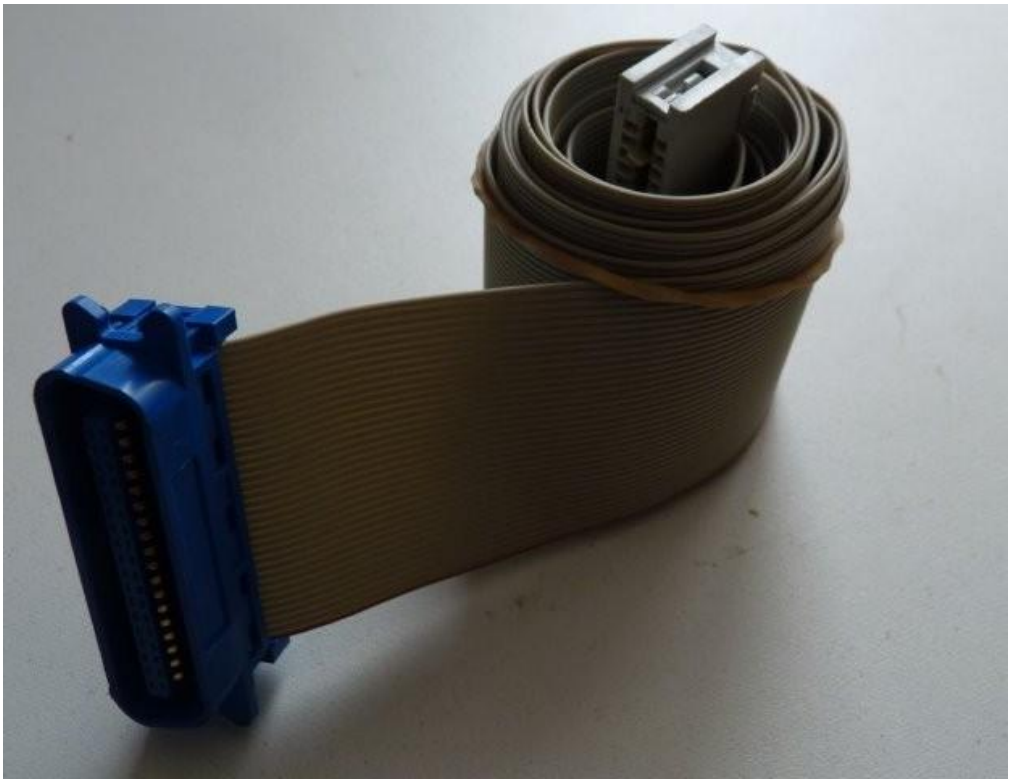
Top view Version No. 1

Top view Version No. 2

Bottom View



The printer connection cable



Connector assignment "Centronic"

Parallel Interface

A 36-pin plastic female connector located at the right rear of the printer provides the means for connecting the printer to a Computer.
The pin arrangement of the connector, signal summary, and the interface timing are shown below.

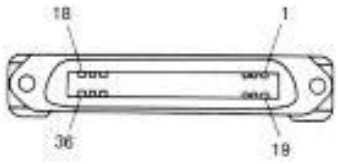


Fig. 8 Pin Arrangement

SIGNAL PIN	RETURN PIN	DESCRIPTION
1	19	$\overline{\text{STROBE}}$
2	20	DATA 1
3	21	DATA 2
4	22	DATA 3
5	23	DATA 4
6	24	DATA 5
7	25	DATA 6
8	26	DATA 7
9	27	DATA 8
10	28	$\overline{\text{ACK}}$
11	29	BUSY
12	–	PE
13	–	SELECT
14	–	OV
15	–	NC (Not Used)
16	–	OV
17	–	CHASSIS GROUND
18	–	5V (80 mA)
30	–	OV
31	–	NC
32	–	$\overline{\text{FAULT}}$
33	–	NC
34	–	NC (Not Used)
35	–	NC
36	–	NC

The parallel interface is supplied with + 5V printer on pin No. 18 of connector "Centronic".

To avoid reversing the courcircuit HE9 connector (card edge) A polarizer is inside. Some printers have a + 5V output at Pin No. 18 and are protected against a possible court-circuit, (but not for all printers).

The diagram and the Gerbers for this card are available free in the Downloads page of my website.

http://www.prof-80.fr/

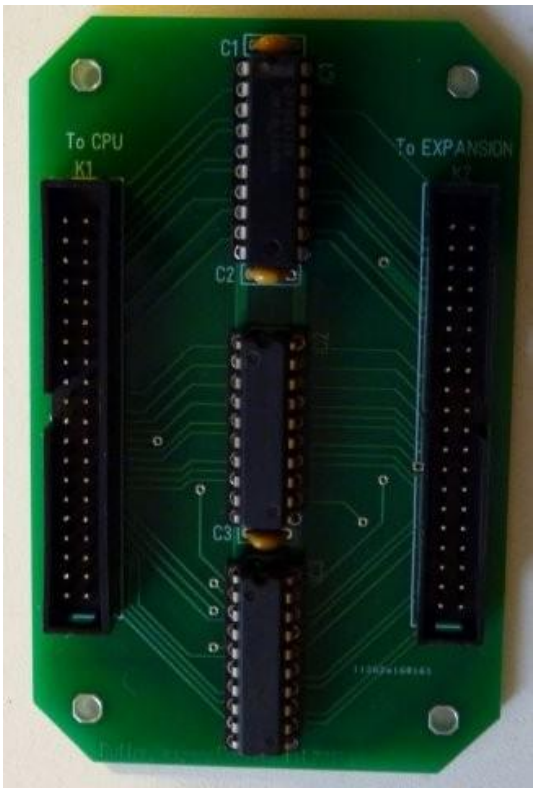
Prototype Expansion Interface/M1 buffer

I built a prototype board based on the principle of the Tandy E/I buffer

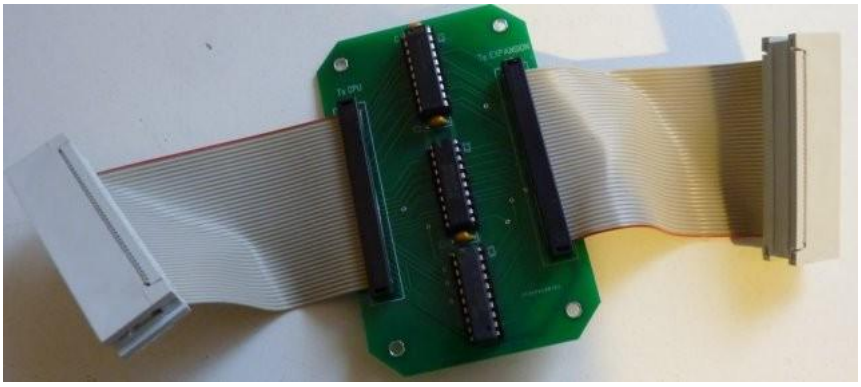
Goal:

1. To retrieve the boxes with expansions lost Buffers.
2. Keep track of our digital heritage
   Provide a diagram for troubleshooting and understanding the operation of the card.

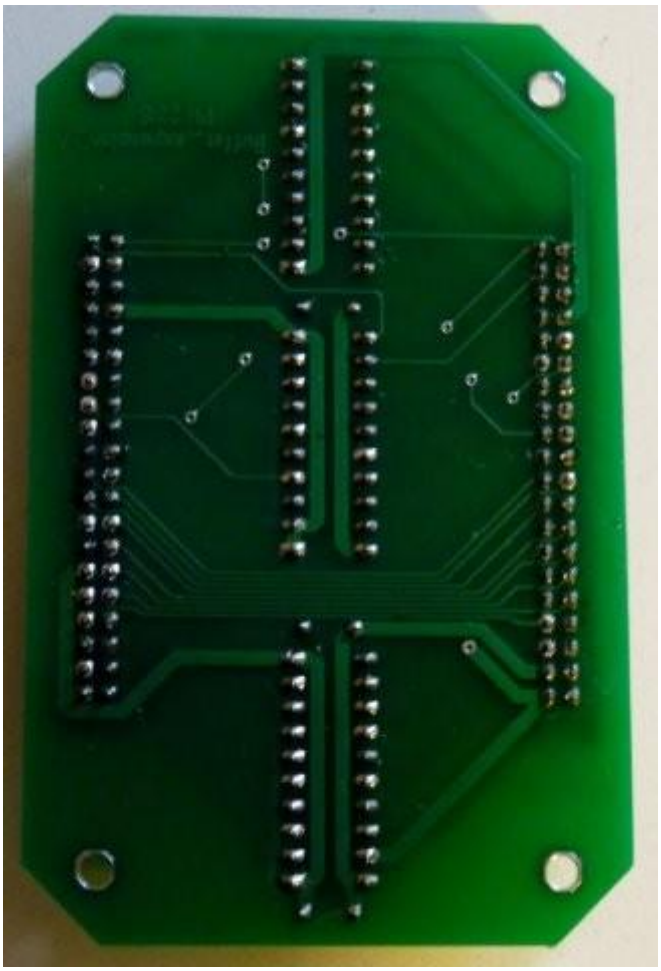
Version No. 1 with H10 connector



Version # 2

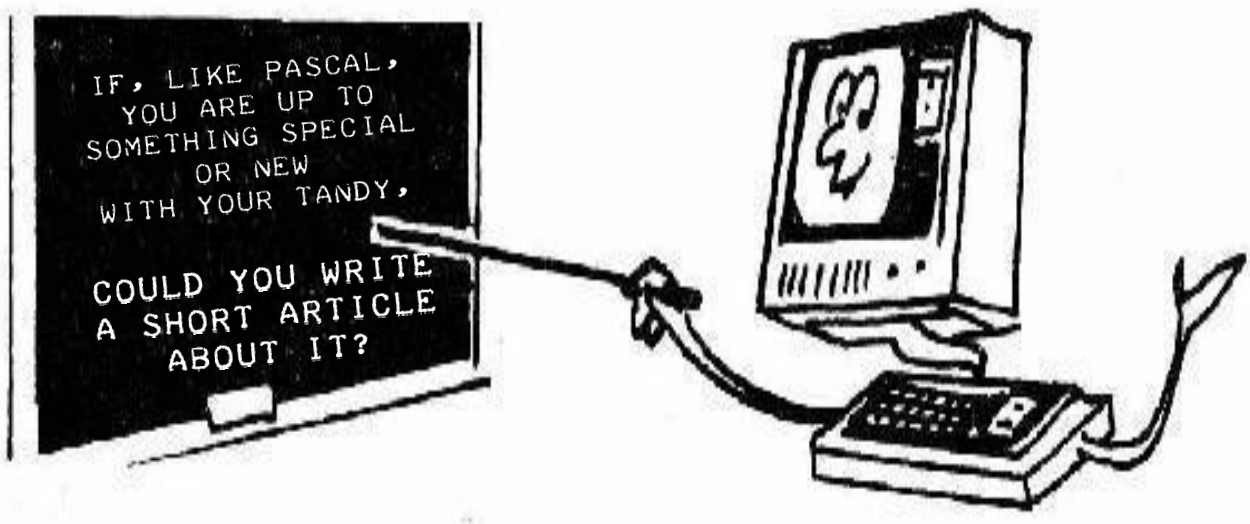


Bottom View



The diagram and the Gerbers for this card are available free from the Downloads page of my website

http://www.prof-80.fr/



TRS-80 Emulators . com

TRS32: A Model I/III/4/4P Emulator For Windows

written by Matthew Reed

Unregistered Shareware Version:

- Works under all current versions of Windows
- Full Windows application — no low-level hardware conflicts!
- Model I, Model III, Model 4, and Model 4P emulation
- Four floppy disk drives (with optional realistic disk drive sound)
- Cassette tape drive with graphical on-screen controls
- Exatron Stringy Floppy emulation
- Printer support
- Serial port for RS-232 communications
- Joystick support (using a Windows joystick — TRISSTICK and Alpha Products joysticks are emulated)

Registered Version:

- All features included in the shareware version
- Built-in emulation of an Epson FX-80 dot matrix printer (including graphics and control codes)
- High resolution graphics (Radio Shack and Micro-Labs)
- Up to 1 megabyte of additional memory in Model 4 and 4P modes
- Hard disk support
- Orchestra 85/90 music generation

Interested?

- Read the TRS32 emulator documentation
- Download the shareware version
- Register online

Actual Dotwriter output — emulated FX-80
click for larger sample

MATTHEW'S EMULATOR IS QUITE ASTOUNDING.
IT'S MY FIRST CHOICE WHEN DEVELOPING IDEAS FOR TRS8BIT.
HE OFFERS FULL SUPPORT AND IT IS STILL UNDERGOING ENHANCEMENT.
IF YOU HAVEN'T REGISTERED YOUR SHAREWARE VERSION YET, PLEASE DO SO AND ENCOURAGE
AND SUPPORT MATTHEW'S ENTHUSIASM TO CONTINUE WITH HIS WORK.

ED.

I THOUGHT IT WOULD BE INTERESTING JUST TO SEE WHAT WAS HAPPENING ON THE TANDY FRONT, 35 YEARS AGO, IE IN 1981. I'VE CHOSEN 3 ARTICLES FROM THE AUGUST 1981 EDITION OF 'NATGUG NEWS'. ONE ON HARDWARE, ONE ON SOFTWARE AND STARTING WITH A SHORT REVIEW OF THE **WEEK-LONG** WORKSHOP IN JULY 1981.

MILTON KEYNES WEEK-LONG WORKSHOP - JULY 1981

AS PROMISED MEMBERS WERE GIVEN A BUFFET AT THE STONY OFFICE. BEFORE SETTING THEIR SYSTEMS OUT AT THE COLLEGE. THE COMPETITION WAS WON BY DAVID FRANKLAND FROM LEEDS, SCORING OVER 132000 IN 15 MINUTES! HE WAS AWARDED A SPECIAL SHIELD WITH THE DETAILS ENGRAVED THE NEXT DAY BY A LOCAL SPORTS SHOP.

SATURDAY EVENING
LAURIE SHIELDS TOOK MEMBERS THROUGH THE Z80 SIMULATOR, WHICH INVOLVED THEM WEARING FUNNY HATS AS DATA BUS OR PROGRAM COUNTER, AND PROCEEDED TO SHOW HOW THE PROCESSOR WORKED. THE SESSION WAS THOROUGHLY ENJOYED WITH REAL ALE PASSED AROUND TO REFRESH THOSE PARTS NOT REACHED BY LOGICAL THOUGHT.

LEON HELLER DID A LONG SESSION ON COMMUNICATIONS, WHICH ENDED UP WITH A LOGON TO FORUM-80 IN HULL, 80-NET IN STONY, AND THE UNIVERSITY OF ASTON. THIS OBVIOUSLY INSPIRED DUSTY MILLER TO POSTPONE HIS SLEEP IN HIS MAXI, HUNT AROUND FOR AN UNLOCK ROOM WITH A 'PHONE AND SUCCESSFULLY LOGON WITH HIS **4K SYSTEM**, TO A BB IN THE STATES !! - 4K DOES RULES OK'.

SUNDAY
IT WAS WITH MUCH PLEASURE THAT FREDK. BROWN AND BOB MACEY PLUS MESDAMES BROWN MADE SUNDAY, WITH FREDK. RUNNING THROUGH FORUM 80 AND THE LNW80 BOARD WHILE I TOOK HIS FAMILY DOWN TO A LOCAL SPOT WHO THEN ENJOYED THE HIRED BOATS AND BIKES.

LAURIE SHIELDS WENT THROUGH HIS ASSEMBLER FOR BEGINNERS WITH A VERY APPRECIATIVE ROOM. AS A RESULT OF THESE SESSIONS THE GROUP WILL HAVE ITS OWN PUBLICATION, ON ASSEMBLER FOR BEGINNERS.

TUESDAY
CHRIS CAIN DID A VERY DETAILED SESSION ON CHAINING IN NEWDOS/80, AND AS A RESULT I MANAGED TO CHAIN THE INFINITE BASIC STEPS EG BINS-BASIC 50000-RUN'INSERT/BAS AND SO ON. FOR THOSE WHO ARE INTERESTED, JEREMY SAN HAS A CHAIN FILE BUILDER.

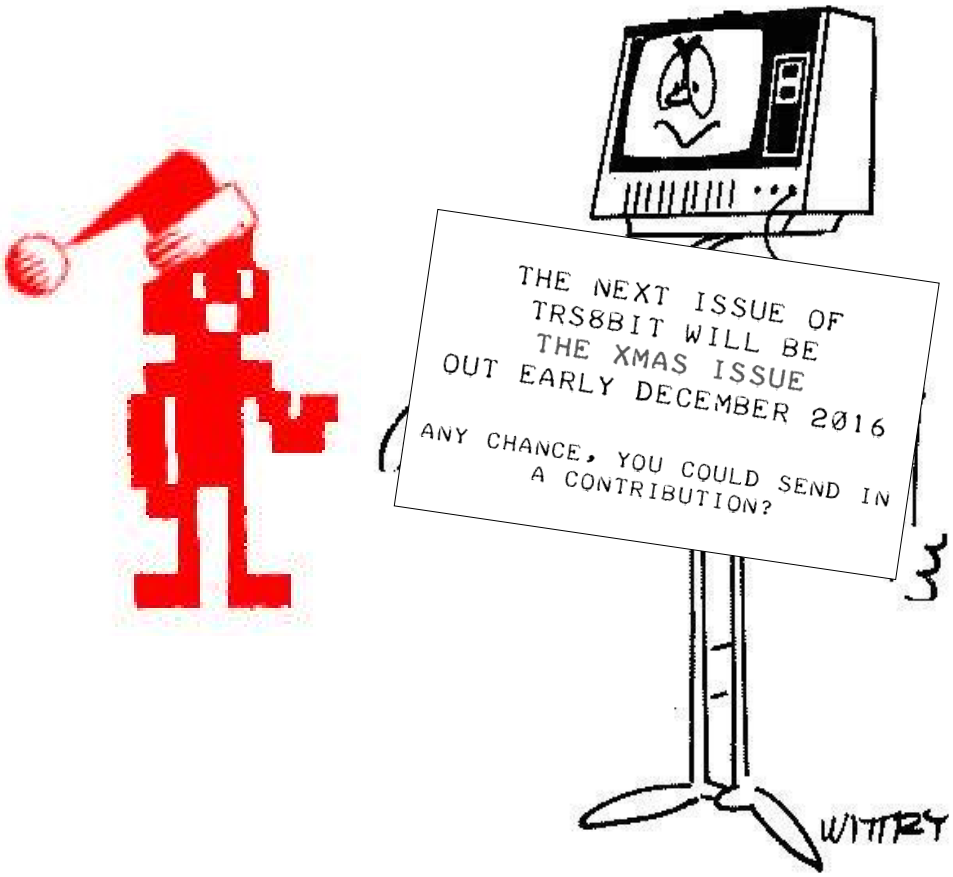
THURSDAY
STEVEN BIRD WAS KEPT AWAKE AND ASKED TO DO A SHORT SESSION ON SPOOLING USING NEWDOS/80. THIS UNFORTUNATELY CONFIRMED TO THOSE PRESENT, THAT IT WAS NOT QUITE AS GOOD AS THEY HAD HOPED. LDOS APPEARS TO HAVE A MUCH BETTER SPOOLER AND I AM SURE READERS WOULD LIKE SOME CLEAR EXAMPLES OF HOW TO USE THIS VERY POWERFUL FUNCTION. (HOW ABOUT IT ALLAN, RIC, STEVEN OR SGT. MURPHY?)

ALL THROUGH THE WEEK
A LOCAL MEMBER LENT A MODEL 3 AND AS A RESULT THE GROUP NOW HAVE THE PATCHES TO MAKE SUPERZAP/BAS RUN ON A M3.

THE MOST SYSTEMS SET UP AT ANY ONE TIME WAS 32, RANGING FROM 4K TO 48K MODEL 1'S WITH DISKS, WITH A NUMBER OF 16K VIDEO GENIES AND MODEL 3'S ALSO IN EVIDENCE. I HOPE THIS WILL ENCOURAGE NEW MEMBERS TO COME ALONG AND ENJOY THE UNIQUE ATMOSPHERE OF WORKSHOPS IN THE FUTURE.

THE LIBRARY WAS WITH US AGAIN AND MEMBERS EXPRESSED THEIR APPRECIATION OF LEIGHTON'S WORK, BUT JUDGING FROM HIS COMMENTS, HE ENJOYS HIS PRIVILEGED POSITION.

BRIAN PAIN



Lower Case With Descenders for the TRS80

Members may be interested in a lower case modification for the TRS80 which gives true descenders and an English pound sign. If you have a recent TRS80 with the new ROMS your character generator will probably have a lower case generator with descenders capability but if you have an early chip without descenders you have two options for obtaining descenders.

Option one is to purchase a recent character generator chip direct from Radio Shack in the US. This can be ordered from:
Radio Shack National Parts Department. 900 East Northside Drive, Fort Worth, Texas 76102. USA.
Order part number: AXX-3027.
Air Mail and shipping costs will bring the total charge to about £18 including duty. This chip is a plug in replacement and with the standard Electric Pencil lower case mod. will give good results.

If you are a 'do it yourself' type you may prefer option two. This is to use the 74S262 Teletext character generator. This is available from Technomatic Ltd. for about £8. Unfortunately it is not a direct plug in replacement. There are also a few characters which are different from the TRS80 chip but most of these can be corrected by the neccesary software.

The 262 chip will give a 5x9 character matrix with automatic blanking of the lower 3 lines. This was done by hardware in the TRS80.The (#) sign will produce an English pound sign (£), using this chip unless corrected by software to read the new (#) sign location in the chip.There is no underline in the 262 chip therefore the cursor character will normally be the (#) sign in BASIC. Using a suitable software patch any of the unused characters could be used as a cursor. The solid block, the last character in the table provided is a good alternative to the underline.The left arrow used by Pencil for line feed/carraige return indication is slightly different. This should cause no problems in use. Probably the only real disadvantage with the chip is the lack of a slash line through the zero. A zero is identical to the capital letter O. The full stop and colon signs are produced by a single dot instead of the groups of four dots used in the Tandy character generator. This makes them slightly less legible on a small screen.

A method of fitting the new chip is to use a 18 pin IC header to replace the original character generator. Thin wires can lead from this header to an additional 20 pin holder fitted to a small piece of Vero board to allow the cross connections required by the new chip. A small modification to the TRS80 board is required to allow the extraction of bit 6 from the data lines for use by the new chip. This was not required on the original chip.The copper track can be cut as shown in the diagram and the signal linked to the new chip as shown.The 262 chip in its holder and board can be fixed to the TRS80 board by the use of silicon rubber. This is freely available in the form

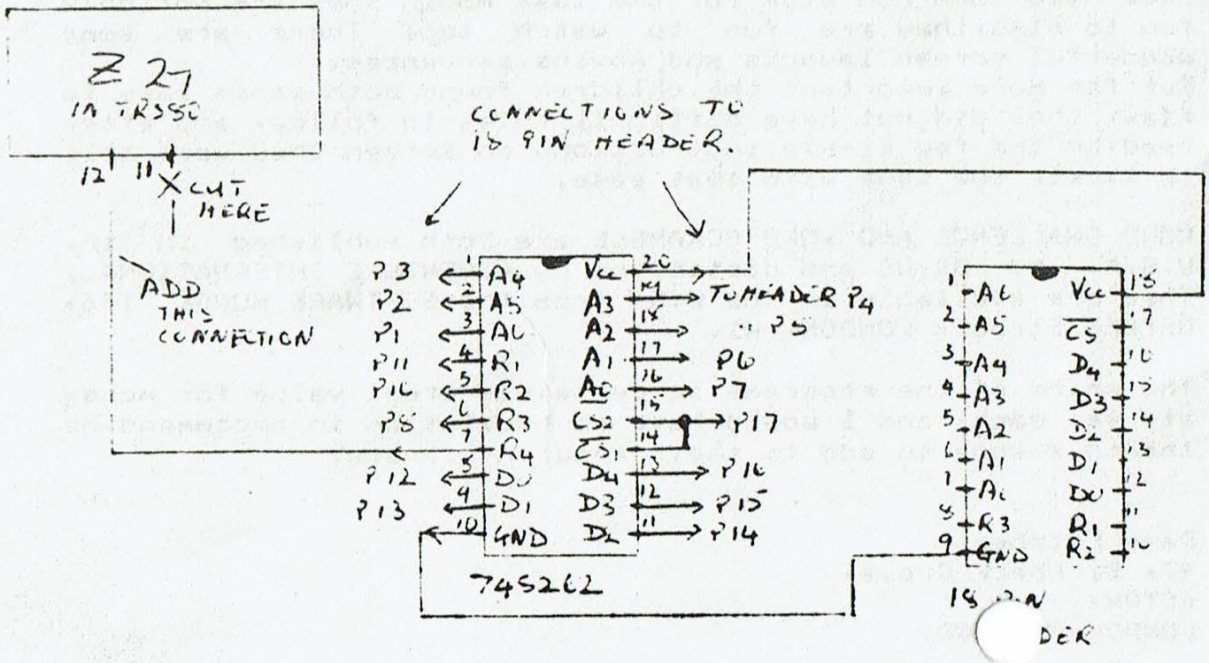
sold in hardware shops for sealing round sinks and baths. It is necessary to locate the new chip and its board directly over an existing IC but the ones used do not run hot and no problems have been encountered in my installation. There is room for the board and the new chip in its holder if the board is located about one inch to the left of the old character generator holder. This will allow the board to fit in the recess forming the foot of the case.

A further improvement can be affected if required by wiring a 74LS00 Nand gate on the Veroboard to allow automatic selection of the correct combination of bits 5 and 6. This will make the need for a switch as used in the Electric Pencil mod. redundant. This is probably similar to the modification fitted by Tandy when they instal their own lower case mod. for Scripsit.The only disadvantage with this mod. that I have found is that the left arrow prompt used by Pencil to signify a C/R L/F is no longer available. A letter M replaces it. This is a little disconcerting at first but as it is only a video prompt and does not appear in print it is only a slight inconvenience.

The circuit below shows the additional Nand gate incorporated to remove the need for a switch. Scripsit may use some other of the special characters of the original chip and cause a different display. As I do not have Scripsit I can not test this.

Given below is a table of characters in the Tandy character generator chip and the equivalant 262 device for comparison. I cannot show the exact dot locations here but the 262 places the characters one dot higher to allow for better looking descenders. This is not noticable unless the two chips are directly compared.

Replacement of Original Character Generator by the 74S262 :



Additional Nand Gate to Eliminate Need for a Lower Case Switch :

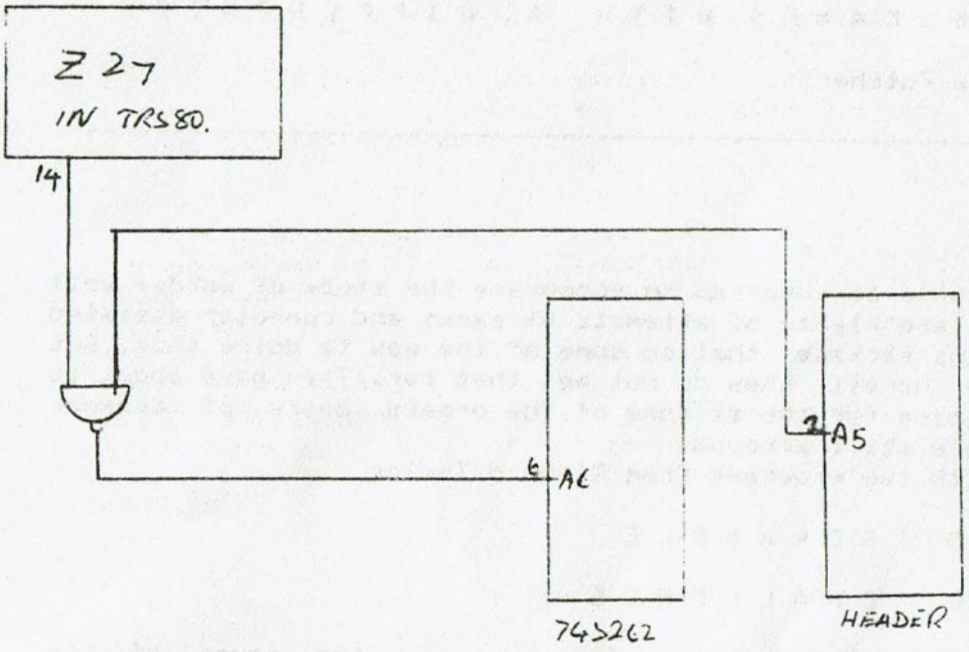
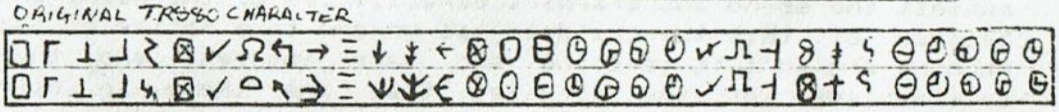
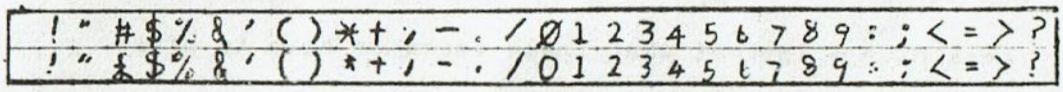


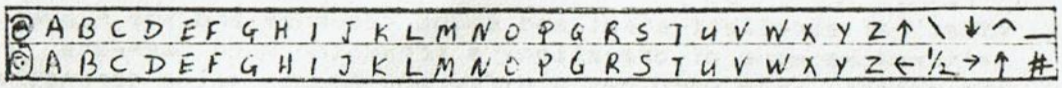
Table of Characters in the two Character Generators :

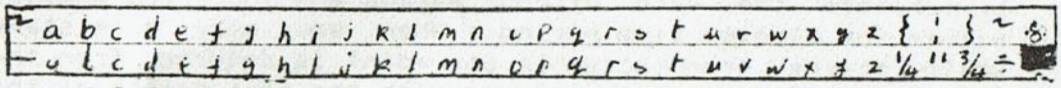
ORIGINAL TRS80 CHARACTER



74S262 CHARACTER







J.R.Barber,

SOUTHERN SOFTWARE FULL SCREEN EDITOR

A review by Edward Rashbrooke

Tandy owners with access to mainframes or other microprocessors may be surprised that the TRS80 does not include full screen editing of programs. On the other hand, users with only experience of the TRS80 probably do not know what they are missing.

A full screen editor allows viewing of any section of a program text, and amendment of anything displayed on the screen. Additional facilities to move or copy text are usually provided and rapid paging backwards or forwards through the text is also helpful.

The Southern Software EDIT program(£17.95) is one of the better screen editors around, giving many useful facilities to assist speedy program development. Once used it becomes quite a shock when doing editing again with the "I"s, "D"s and "X"s of the standard ROM editor. The program is supplied on a tape and can be copied to disk for use with TRSDOS or NEWDOS if required. It is also relocatable, requiring a target address to be specified when first invoked, although relocated versions can be saved for later use. This relocatability is a great advantage because users can then shunt different utilities together in protected memory as required(in fact all Southern Software is suplied as relocatable). Together with the tape comes a 6 sided A4 dscription of the use of the program although after about 2-3 hours, use of the editor becomes second nature. Helpfully, the document does include a single sided reference summary.

Once loaded, the editor is invoked by typing /EDIT or /EDIT nnnn where nnnn is to be the first line displayed. Fifteen lines are then displayed with a vertical bar in the left and right most columns, and 5 digit line numbers in the left margin. Lines longer than 62 characters are wrapped around to the next line, and full statement lengths are allowed(not the case with all screen editors). During an edit session the left and right margins are used to display information about a particular line, and the bottom line is used for messages and commands. The whole display only takes a few minutes to get used to and is quite comprehensive in the information provided.

Once invoked, the editor provides full cursor control by the arrow keys, and automatic scrolling up and down of the program text. Moving the cursor to the top or bottom line and keeping the key depressed causes fairly rapid scrolling of the text. All keys have automatic repeat on them and the cursor is displayed as a solid block for clarity.

A number of special function keys are provided using BREAK, ENTER, CLEAR and the @ symbol in conjuction with other keys. In fact, for systems with a numeric keypad, each key of the keypad has been redefined in conjunction with the @ key. This gives a group of PF (programmed function) keys similar to those on many expensive mainframe systems.

These PF keys can be summarised as:

@/1 jump to start of line
@/2 jump to start of next line?@/3 down arrow
@/4 left arrow
@/5 right arrow
@/6 up arrow
@/7 insert character
@/8 delete character
@/9 delete rest of line

Although some of these functions may sound simple, the increase in editing speed brought about by their use makes them very worthwhile.

Another type of special function is brought about by line commands. These can operate on single lines or blocks of lines. These can be summarised as follows:

@/D delete line
@/I insert line
@/R replace line
@/C copy line
@/M move line
@/A after (used with move/copy)
@/B before (used with move/copy)
@/P reposition line
@/Q start of block
@/F repeat find
@/G global change

These commands are a pretty comprehensive set of facilities for speedily changing text, and normal text revisions are accomplished by simple overtyping of the required section of the current display. ENTER can be pressed to fix all of the changes, and unlike a number of text editors, these changes can be to all lines displayed at any time and not to just one line at a time.

Use of the block commands is particularly powerful when repetitive sections of code must be created, and the improvement in the formatting of source code can be remarkable(albeit at the expense of space). During development, however, there is much to be said for tidy source code, even if the production version is compressed to save space etc.

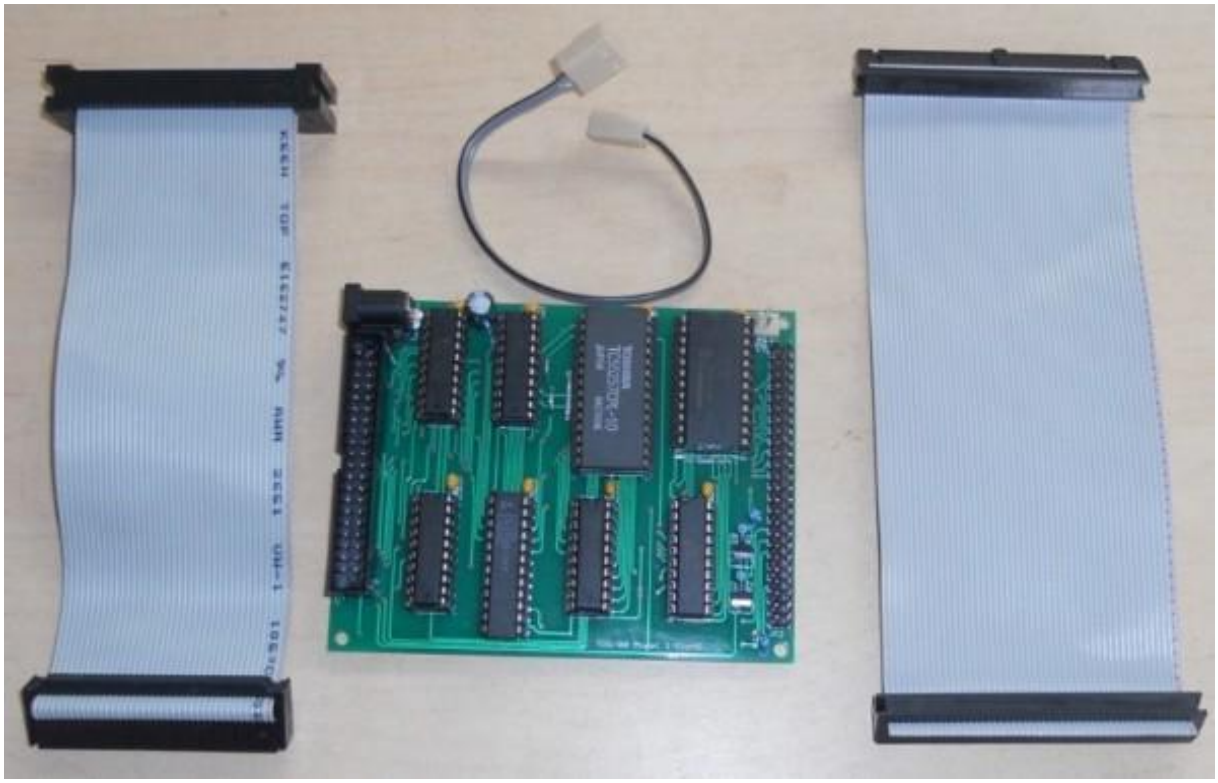
Having now used the EDIT for some time I have only two criticisms of it, neither of which are particularly severe. The first problem is that block moves and copies require manual renumbering. There are no facilities to use a RENUM function on the text, although for blocks of text this would be quite useful. I suppose the actual renumbering would need to be quite sophisticated to cope with the problems of repeated text in a module, and may be disproportionally large compared with the modest 3072 bytes of the existing module. The other criticism concerns the FIND/CHANGE commands, which would be nicer if they allowed longer text strings than at present.

On the positive side, once loaded into the top of memory (I have a 48k system) the program remains there most of the time that I am working in BASIC, allowing RUNs of the BASIC and then further editing without problems. On a 16k system the 3k size of the module may cause a few more problems when in use with larger programs. Editing has been greatly simplified and speeded up and the resultant source looks tidier now than my previous attempts using the ROM editor. In conclusion I would say that the editor is well worth the money for anyone involved in significant amounts of BASIC program development.

The program was originally written for TRS80 Model 1 and Video Genie, although it also works with my Model 3 with slight changes to the loading procedure.

“Quinnterface” Mini Expansion Interface for 16K Model 1 FreHD users.

- COMPLETLY ASSEMBLED AND TESTED -



This is THE perfect device for all Model 1 users who own a 16K Level II unit, (which is most of us!), but no Expansion Interface or disk drives, especially if you don't want to modify your M1 with upgraded boot ROM or memory upgrade.

The ‘Quinnterface’, developed by J. Andrew Quinn from New Zealand, adds 32K RAM and auto-boot functionality to you FreHD.

U.K. And Europe, contact Bas. at BetaGamma Computing.
U.S.A. And rest of the world, contact Mav. At “The Right Stuff”

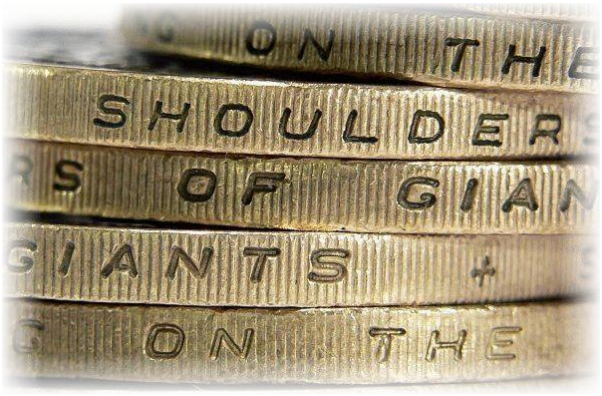
LOOKING FOR FAST, INEXPENSIVE, UNLIMITED MASS STORAGE FOR YOUR TRS-80 MODEL I/III/4/4P/4D?

The amazing

"FreHD"

-Emulates a TRS-80 hard drive, but faster than any hard drive!
-Works with your favourite DOS (LS-DOS, LDOS, CP/M, Newdos/80 2.5)
-Uses SD card for storage medium
-Bonus free Real Time Clock function!
-Designed in Belgium and proudly built and shipped from Australia
-Kit form or fully assembled

Order yours today:
http://ianmav.customer.netspace.net.au/trs80/emulator



USING NEWTON'S QUOTE "STANDING ON THE SHOULDERS OF GIANTS" AS INSPIRATION, I OFFER ANOTHER IN THE SERIES OF SOME OF THE KNOWN, AND LESSER-KNOWN NAMES WHO INSPIRED MUCH OF WHAT WE, TODAY, TAKE FOR GRANTED.

Charles Babbage

Though never fully implemented, Charles Babbage's machines were the forefathers of the modern computer

'I wish to God these calculations had been executed by steam!' Charles Babbage exclaimed as he laboured over the tables of the Nautical Almanac. The 19th century had developed steam power but accurate navigation at sea was still a problem. A ship's position was found by observing the moon and then using mathematical tables that were often inaccurate.

It was in 1812 that Babbage first had the idea of building a machine, which he called the Difference Engine, that could perform the laborious calculations needed for the nautical tables. By 1823 he had completed a small model and approached the government for a grant to build a working machine. The Chancellor of the Exchequer gave him £1,500 and he set about building an engine that would eliminate errors by automatically printing the results of its own computations.

Babbage's life's work was thereby determined. The project consumed vast sums of money, for he was working at the frontiers of the engineering skills available at that time. He obtained the money through the assistance of the Prime Minister, his friend the Duke of Wellington. Despite Babbage's confidence that 'whatever the engine did it would do truly', the government eventually withdrew, having sunk £17,000 into the venture. Babbage's engineer, Joseph Clement, resigned shortly afterwards, following an argument, and took with him all the tools that had been specially machined for the engine.

Babbage moved quickly onto a more ambitious project, the Analytical Engine, which was intended to accomplish all that the Difference Engine had been designed for and much more besides. Its design in many ways resembled that of the modern computer. It contained a memory store and an arithmetic 'mill' (equivalent to the CPU), provided printed output, and could even be programmed, using conditional branching.

At first the instructions were controlled by spikes as in a barrel organ; later the punched card system that Joseph Jacquard had introduced into the weaving industry was adopted. Babbage also experimented with different number bases but as all his machines were mechanical, there was no advantage to be gained from using the binary system.

He was joined in his project by his companion Countess Ada Lovelace, who was a gifted mathematician. They were dogged by problems, not least money. She lost much of her wealth gambling on an 'infallible' horse-racing betting system. After her death at the age of 36, Babbage continued alone.

A man of prodigious energy, he also invented the doctor's ophthalmoscope for looking into the eye, choreographed a ballet, devised a system of stage lighting and invented a technique for signalling at sea.

In his last years he grew irascible. Expecting a peerage, he turned down the baronetcy offered as recognition for his work.

Babbage's work anticipated the structure of the modern electronic computer but he failed to realise his vision fully. His Analytical Engine remained unfinished, its completion thwarted by the technical limitations of 19th-century engineering science.

1792
Born in Totnes, Devon, on 26 December

1810
Goes to Trinity College, Cambridge, to study mathematics

1814
Marries Georgina Whitemore

1822
Publishes a paper entitled 'Observations on the Applications of Machinery to the Computation of Mathematical Tables'. Receives the first Gold Medal of the Astronomical Society, which he helped to found

1827
Cambridge appoints him Lucasian Professor, the chair once held by Newton, at £80 a year, though he never lectures or takes up residence

1833
Parliamentary candidate at Finsbury

1834
Work on Difference Engine suspended after the engineer Joseph Clement resigns

1862
Partially complete Difference Engine exhibited at South Kensington, London

1871
Dies on 18 October

IN MAV'S WORKSHOP
"THE LONG WHITE CASED COLOR COMPUTERS"

by

Ian Mavric

In The Beginning: Everything Looked Grey

By 1983 the Color Computer 1 and it's specs had gone a little stale. Released in 1980 and sold as essentially the same computer until the beginning of 1983, Radio Shack looked at ways to keep the computer relevant. Only one external change in the Coco 1 lifeline was the "cleaning up" of the case in 1982. This involved placing the TRS-80 emblem in the middle of the case, removing the black border around the keyboard along with the Ram-size badge. At the same time the 4K Coco 1 was dropped. Under the bonnet there had been three motherboard designs. Most 1980 computers had what was known as a "D" motherboard. Designed around 4K and 16K operation, it was difficult to upgrade to 32K and even more difficult to upgrade to 64K (an important number to remember, but bear in mind 64K in the Coco hadn't even been thought of in 1980-81). 1981 saw the "E" motherboard which was designed to work with 4K, 16K and could be easily upgraded to 32K. 32K became important because also in 1981 the Coco disk drive became available and while it did run on 16K systems, very little memory was left over to run programs once a disk drive was added. These "D" and "E" systems had the original case which included the Ram-size badge.


Pictures: 1980-81 Color Computer (left) and 1982-early'83 re-styled Color Computer (right)

Things began to stir in the Coco world in 1982. As well as the disk drive, a port of the 6809 multitasking Operating System OS/9 was being debugged and pitched to Tandy for possible sale in Radio Shack stores. Around this time as well some clever individuals at Frank Hogg Laboratories found they could upgrade the Coco to 64K and run a ported version of Flex, an alternative Operating System to OS/9 in the 6809 world. With 64K memory on tap OS/9 becomes a much more viable Operating System and 64K puts the Coco in the same "class" as the newly released Commodore 64 and market leading Apple IIe. It's also as much memory as was going to be in the soon-to-be designed successor to the Model III, what we call the Model 4 TRS-80. So Tandy engineers design yet another Coco motherboard, this time called an "F" board. In fact it doesn't have "F" on it anywhere but since it came after the "E" board that's what the Coco community called it. "F" boards are designed to run either 16K or 64K memory. At this time Tandy changed the case as explained above. The chicklet keyboard was retained, and like all 1982 Radio Shack computers, it was grey.

An interesting thing from this time is that clearly there was an advertising anomaly in which this 64K "F" board Coco 1 was marketed as a 32K system. I think the reasons for that are threefold. 1. This new system carried the same catalog number as the old 32K "E" systems: 26-3003. It's possible that the entry for the 32K Coco had been locked into the Radio Shack 1982 Catalog in September 1981 before the design for the 64K Coco 1 had been finalised. 2. PRINTMEM command on the 32K "E" or 64K "F" systems returns the same number of bytes free: 24871. I can see where it might be confusing where someone thought they were buying a computer with extra memory which they couldn't actually use in the BASIC language. 3. The official release date of OS/9 was still some time away and an upgrade to the DOS version (from 1.0 to 1.1) was needed to easily boot the DOS (via the "DOS" command). Best practice would have been to try to sell out as many of the grey disk drives as possible, before upgrading the whole Coco line-up in 1983 with new devices and software to bring it properly into the mid-80s.

You can have your TRS-80 any colour you like, as long as it's white

By 1982 the marketing people at Tandy realised something very concerning. Every popular computer system on the market was either white, off-white, beige or some other variation of a theme. This includes the IBM PC(white and beige), Apple][(off-white), C= Pet (white), C= VIC-20 (cream), C64 (beige), and even some popular mainframes like the IBM/360 is off-white.

Grey was looking so last decade and most of the consistent sellers in the TRS-80 lineup were based on designs from the late 70s (I'm talking about the Model II and Model III here) that a decision was made that 1983 would be the year TRS-80 became serious looking and joined the rest of its data processing compatriots.

By the end of 1983 the last Model II, III, and grey Coco 1 had been sold, Tandy printers had underwent a name change (from Line Printer to DMP) and most carried over accessories had been treated to a nice new white paint job (this includes the CCR-81 tape deck, Multi-Pak Interface, and Modem I).
In place of the Model II was the new Model 12 and 16 (released late 1982 and the first TRS-80 to show off the new white look), in place of the Model III was the Model 4, and in place of the 32K grey Coco 1 was the flashy new white Color Computer, and hallelujah, it finally has a decent keyboard!

Picture: the US 64K Color Computer cat. no. 26-3003B

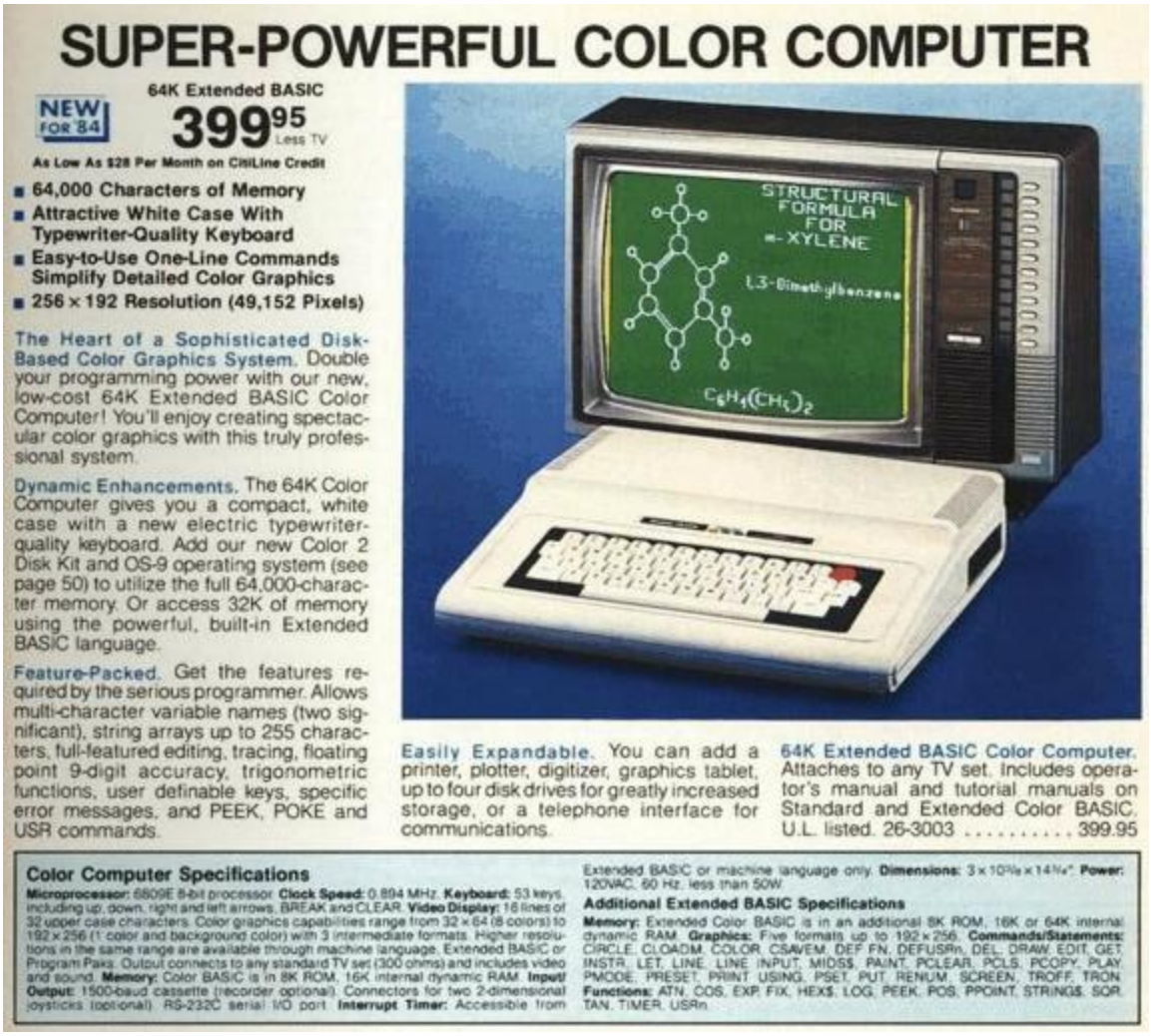
Duh, so what do we call this here new 64K Color Computer?

Not being shy of criticism, in 1980 when the Coco 1 was released Tandy was accused of being unimaginative when naming its new small home computer, capable of being connected to a colour TV set, the "TRS-80 Color Computer". Many other brands had released colour TV connectable home computers (the Apple][and the Atari 400/800 spring quickly to mind) that naming the new TRS-80 the Color Computer was taking the bleeding obvious too far. Tandy's rationale was that it was their first computer which supported colour graphics. This was true and the public bought it, both figuratively and literally. Since the new system had 64K memory this seemed to be a good marketing point, and the 64K Color Computer was born.

Picture: inside the US 64K Color Computer features the "285" aka "F" motherboard

Externally its the same case as the Coco 1 "F" system, this time moulded out of textured white plastic, so familiar to those who use the Model 4P. The keyboard is now also white, its not full-travel but it is nicer to type on. Referred to as a "melted" keyboard, it helped to rectify probably the biggest criticism of the grey Coco 1, it's chicklet keyboard. This one looks better and works better for fast typing. You could actually write a novel on it and be happy with its performance which is something that could not be said about the chicklet. It still wasn't up there with the VIC-20/C64 keyboard but the press at the time praised it nonetheless.

Under the bonnet was a complete anti-climax. It contains the same "F" motherboard as the previous grey Coco 1. This is how all smart companies who are onto a good thing operate.
Take a proven product, and re-package it to keep up with the current styling trends of the time. Car makers do it all the time. The catalog number was upgraded at this time to 26-3003B, and production was done in Fort Worth, Texas where all US TRS-80s are built.



SUPER-POWERFUL COLOR COMPUTER

NEW FOR '84

64K Extended BASIC
399.95 Less TV

As Low As $28 Per Month on CitiLine Credit

- 64,000 Characters of Memory
- Attractive White Case With Typewriter-Quality Keyboard
- Easy-to-Use One-Line Commands Simplify Detailed Color Graphics
- 256 × 192 Resolution (49,152 Pixels)

The Heart of a Sophisticated Disk-Based Color Graphics System. Double your programming power with our new, low-cost 64K Extended BASIC Color Computer! You'll enjoy creating spectacular color graphics with this truly professional system.

Dynamic Enhancements. The 64K Color Computer gives you a compact, white case with a new electric typewriter-quality keyboard. Add our new Color 2 Disk Kit and OS-9 operating system (see page 50) to utilize the full 64,000-character memory. Or access 32K of memory using the powerful, built-in Extended BASIC language.

Feature-Packed. Get the features required by the serious programmer. Allows multi-character variable names (two significant), string arrays up to 255 characters, full-featured editing, tracing, floating point 9-digit accuracy, trigonometric functions, user definable keys, specific error messages, and PEEK, POKE and USR commands.

Easily Expandable. You can add a printer, plotter, digitizer, graphics tablet, up to four disk drives for greatly increased storage, or a telephone interface for communications.

64K Extended BASIC Color Computer. Attaches to any TV set. Includes operator's manual and tutorial manuals on Standard and Extended Color BASIC. U.L. listed. 26-3003 ..... 399.95

Color Computer Specifications

Microprocessor: 6809E 8-bit processor. **Clock Speed:** 0.894 MHz. **Keyboard:** 53 keys, including up, down, right and left arrows, BREAK and CLEAR. **Video Display:** 16 lines of 32 upper case characters. Color graphics capabilities range from 32 × 64 (8 colors) to 192 × 256 (1 color and background color) with 3 intermediate formats. Higher resolutions in the same range are available through machine language. Extended BASIC or Program Paks. Output connects to any standard TV set (300 ohms) and includes video and sound. **Memory:** Color BASIC is in 8K ROM. 16K internal dynamic RAM. **Input/Output:** 1500-baud cassette (recorder optional). Connectors for two 2-dimensional joysticks (optional). RS-232C serial I/O port. **Interrupt Timer:** Accessible from Extended BASIC or machine language only. **Dimensions:** 3 × 10¹⁵⁄₁₆ × 14¹¹⁄₁₆". **Power:** 120VAC, 60 Hz, less than 50W.

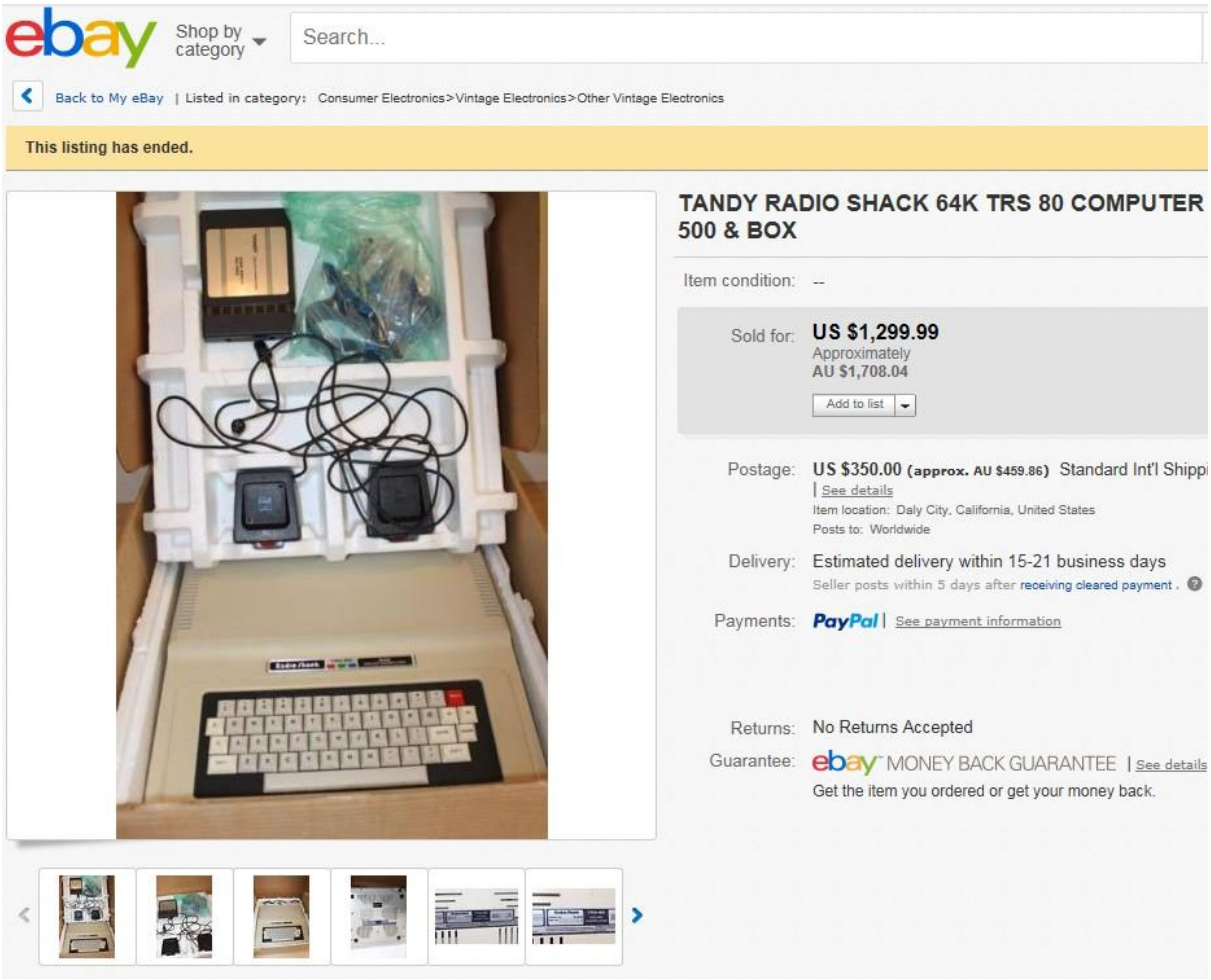
Additional Extended BASIC Specifications

Memory: Extended Color BASIC is in an additional 8K ROM, 16K or 64K internal dynamic RAM. **Graphics:** Five formats up to 192 × 256. **Commands/Statements:** CIRCLE, CLOADM, COLOR, CSAVEM, DEF FN, DEFUSRn, DEL, DRAW, EDIT, GET, INSTR, LET, LINE, LINE INPUT, MID$, PAINT, PCLEAR, PCLS, PCOPY, PLAY, PMODE, PRESET, PRINT USING, PSET, PUT, RENUM, SCREEN, TROFF, TRON. **Functions:** ATN, COS, EXP, FIX, HEX$, LOG, PEEK, POS, PPOINT, STRING$, SQR, TAN, TIMER, USRn.

Picture: one of the only advertisements for the US 64K Coco 1

The 64K Coco 1 also holds the distinction of being the rarest production Coco model. Production numbers from Tandy always have been hard to obtain but if the serial numbers of those which have sold on eBay are any indication, they made only a couple thousand of these and they all sold in March-April 1983 in the USA and Canada only.

When the US Coco 2 was released in June it was obvious that the 64K Coco 1 was aimed to plug a hole in the product line until the 64K Coco 2 was ready to release. This makes the 64K Coco 1, in good condition, extremely collectable. Regularly fetching over $US150 on eBay, it makes the eBay prices of the grey Coco 1 ($20-$50) and Coco 2 ($15-$40) laughable. So rare is the 64K Coco 1 that early on in internet forums there was even debate that the 26-3003B didn't even exist as it was very difficult to find any RS advertising which featured it. It has also been known to fetch crazy prices like in this eBay listing below:



Meanwhile, in Australia....

....and other markets like UK, EU, and New Zealand, a variation on a theme appeared. It looked for all the world like the US 64K Coco 1 but close inspection externally saw it named the Colour Computer 2 (note the 'U' in Colour) and it was made in Korea.

The case was subtly different, being textured painted white plastic, similar to a Model 4. So while the US 64K Coco 1 turns yellow over time, like a 4P does, the Colour Computer 2 case *doesn't* (which is the same as the Model 4). However the keyboard does go yellow over time on either system when subject to sunlight. The downside of this is that while it is possible to find the odd Colour Computer 2 kept in its box it's whole like and it's white and pristine, finding a US 64K Coco 1 is virtually impossible. (Even that expensive one pictured above has yellowed considerably.)

Picture: a US 64K Coco 1 sitting next to a Korean 64K Coco 2. This never happens unless a collector imports one or the other because these two computers were never sold in each others country/market.

Picture: the Korean 64K Coco 2

Picture: subtle differences between the two. US Coco 1 has black cartridge door. Korean 64K Coco 2 has white cartridge door and flat nameplate.

Why is the US 64K Coco 1 not a Coco 2 or vice-versa?

It seems logical that in countries where the Korean Colour Computer 2 was released a clean break from the grey Coco 1 machines heralded the new white Colour Computer 2, rather than added a white version to an existing product line.
But there is a good reason behind this, and that is that 64K Coco 1s (and in fact all Coco 1s) have 12V available at the cartridge slot. All Colour Computer 2s, including all the short-case Coco 2s and the Coco 3 don't have 12V at the cartridge slot.

12V used to be important internally as 12V was needed to power 4K and 16K RAM chips, and also the first Coco 1 disk drive controller (cat. no. 26 -3022) needed 12V to power the FDC chip.
A subsequent redesign of the disk drive controller (cat. no. 26-3029 and it's derivatives 26-3129, -3131, and -3133) no longer required 12V so 12V could be left out of the computers design. Therefore all Coco 2s, even the long-white case one not sold in the USA, has no 12V inside it.

Picture: inside the Korean 64K Coco 2

Around the same time, early on in 1983, a new product was released called the Multi-Pak Interface (or MPI). It was available initially in two colour schemes to match the grey Coco 1s or the white US 64K Coco 1 and long white case Coco 2. It was not properly marketed for a long time; for the first year or two of sales it was pushed as a device allowing 4 game carts to be plugged in at once and changed between them at the flick of a switch.

Yes, it can be used for that purpose, but that is sort of using the MPI for the simplest use. It's price-tag of $350 in Australia didn't help sales either. The MPI has 12V available at each cartridge slot so Coco 1 devices could be used with the Coco 2 and 3, but this isn't it's only function. When coupled with OS/9, the MPI allows connection of disk drive and ability to address the other three slots, which can contain other devices like an RS232 Pak or (later on down line) a hard disk controller.

After the R&D budget had been recovered for the MPI it's price was reduced to a more palatable $179 in Australia. The last word on the MPI was that when the short case Coco 2 came along the styling of the MPI was updated to match those systems and their successor, the Coco 3. For those who have ever wondered, all MPI versions run with all Coco versions. You can plug a grey Coco 1 MPI into a Coco 3 and it works, though is rather odd looking.

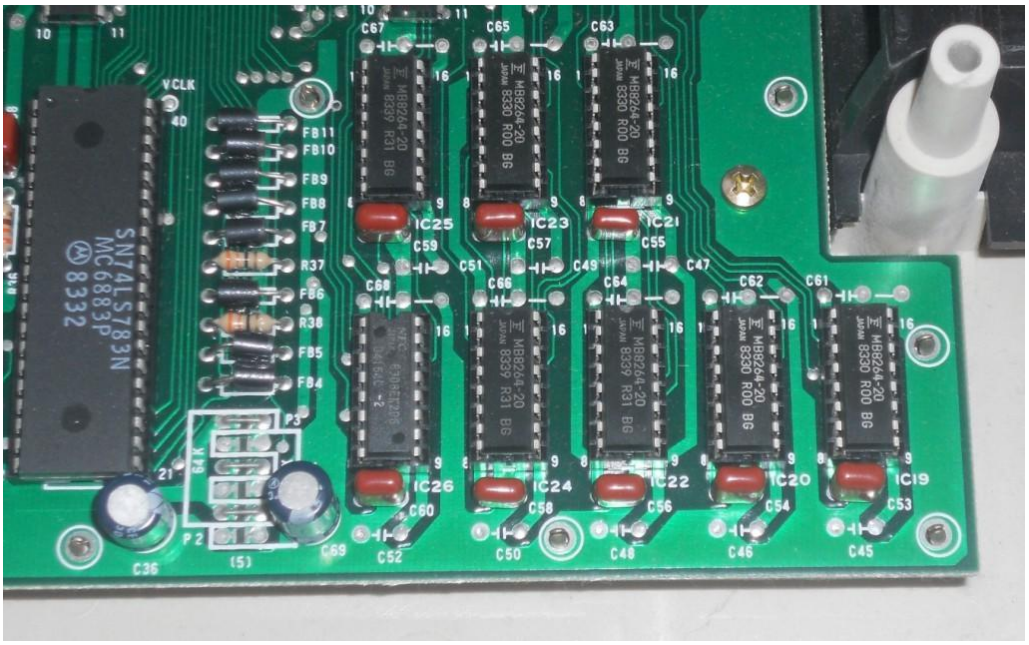
Picture: Multi-Pak Interfaces (MPIs) were available in two paint schemes to match your Coco model.

Pictures: MPIs connected to their respective Coco models look right together

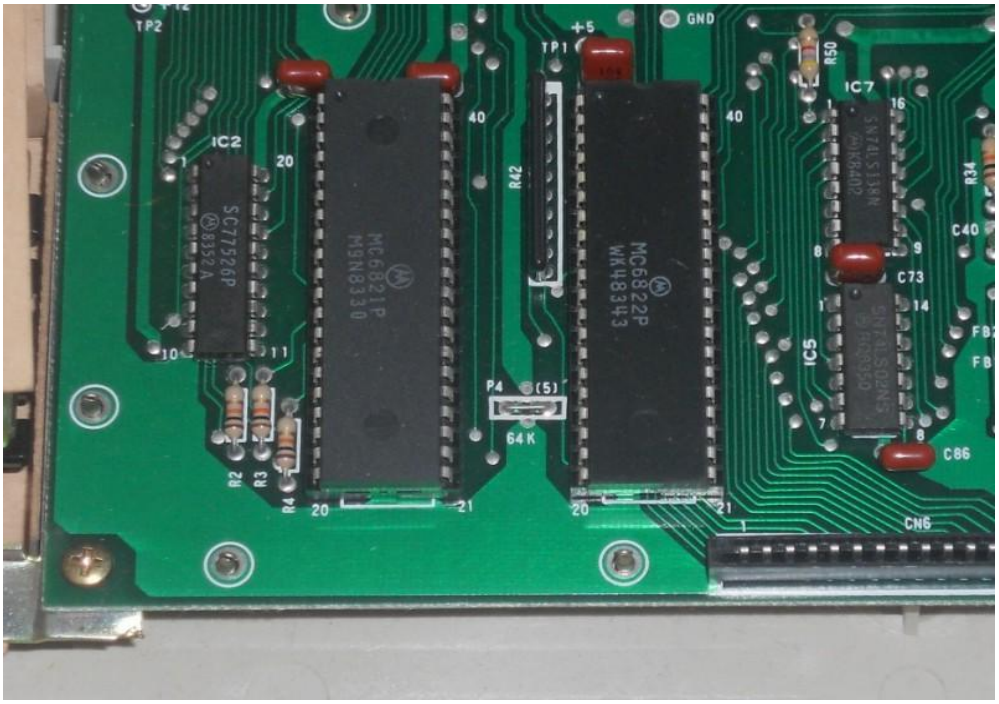
Upgrading Long White Case Colour Computer 2s

Notice I didn't say Coco 1s and 2s, just the Twos. The US 64K Coco 1 came maxed out with 64K memory and Extended Color Basic 1.1 No upgraded required. The Korean Colour Computer 2 came in three flavours: 16K Color Basic (cat. no. 26-3026), 16K Extended Colour Basic (cat. no. 26-3027), and the 64K Extended Color Basic (cat. no. 26-3127).

When you open it you will notice the eight memory chips are re-arranged into a group of five and a group of three. They are special 16K chips which don't need 12V. To upgrade these systems to 64K you need to remove these 8 chips and replace them with 4164s of your choice, my preference being Motorola MC8040665s. Four jumpers need to be attended to. Grouped together between C36 and C69 is a group of three solder links marked 16K. Clean the solder up with a solder wick and jumper the ones next to them marked 64K. There is one more jumper MC6821 and the MC6822 marked 64K. It will be unlinked on 16K systems. Solder a link over it. Finally you need to remove some capacitors which are no longer needed: C45.46,47,48,49,50,51,52,61,62,63,64,65,66,67, and 68.



Picture: memory banks (3 chips and 5 chips) and 16K/64K memory jumper block



Picture: the other 64K jumper which needs to be added for the 16K->64K upgrade

To upgrade the Extended Basic Rom: obtain an Extended Color Basic 1.1 Rom and install it in the IC12 position where there is a vacant socket. If your system came with Extended basic there will be no socket (RS deciding to save 10c on each machine by soldering the chip directly onto the motherboard) and no reason to remove the Extended Basic Rom.

Rarity and Collectability in 2016?

As mentioned previously the US 64K Coco 1 was made only for a very short time, in small numbers, and only sold in the US and Canada. Functionally it's nothing special so the desirability to own one really is if you feel you need one in your collection. In the last couple of years there have been reasonable numbers showing up on US eBay, and collectors are snapping them up.

I have imported two of them in my travels: the one pictured above next to my 64K Coco 2, and one which is in the hands of an Australian collector. I'd be surprised if there were any others in Australia. A note to would-be collectors: Being an NTSC computer, getting it to run on most television sets can be a frustrating experience. You might find your expensive import its best admired in a display case rather than being put into actual use.

The Korean Coco 2 did sell in respectable numbers in Australia being able to be purchased from Tandy Electronics stores for all of 1984. To my eye these are the best looking of the Color Computers and it is how it should have looked in the first place.
It also is the one most people hang onto and so they come up for sale relatively infrequently compared to the grey Coco 1s and the short-cased Coco 2s. I know when I get one of these they sell pretty quickly once I announce that I have one for sale, whereas the short-cased Coco 2s are generally slow movers.

For US collectors these Korean Coco 2s would be very rare indeed but the same warning applies with regards to television standards and Australian computers, being PAL broadcast standard, will have a hard time syncing with most TVs in the United States.

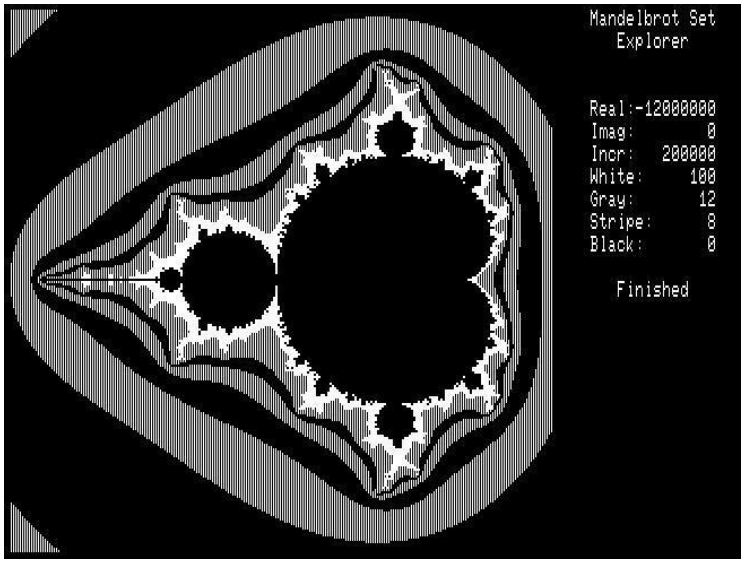
Ian Mavric

ianm@trs-80.com

LOOKING FOR HIGH-RESOLUTION GRAPHICS FOR YOUR TRS-80 MODEL 4/4D?

Then you simply must have the

"Improved Grafyx Resolution Solution!"



-Improved version of the established Grafyx Solution GX 4.2 board
-640x240 resolution for a total of 153,600 pixels, High Bandwidth
-Easy install in minutes inside your Model 4, Gate-Array Model 4, or 4D
-4P version is just around the corner - stay tuned!
-Compatible with most M3 and M4 Hi-Res software for Radio Shack HRG
-Designed in Belgium and proudly built and shipped from Australia or UK

Order yours today exclusively from:

UK and EU: BetaGamma Computing: bgialopsos@aol.com

Rest of the world: http://ianmav.customer.netspace.net.au/hires/hires.htm

NEW REDUCED PRICE USD$165
(DUE TO AU$ EXCHANGE RATE CHANGES)
WITH ADDITIONAL DISCOUNT FOR FURTHER PURCHASES.
CHECK WITH IAN FOR FULL DETAILS

Character generation with the Motorola MCM667x

Larry Kraemer

If you are interested in knowing more about the TRS-80 Characters that are contained in the Motorola MCM667x. Character Generator ROM.
The following BASIC Programs will display all the available characters on the TRS-80.

Basic Programs to display:

1. Special Characters

```
10 CLS
20 FOR I=0 TO 31
30 POKE, 15360+I*16,I
40 NEXT
50 PRINT@ 640,"";
```

2. Text Characters

```
10 CLS
20 FOR I=32 TO 127
30 PRINT@ (I-32)*8,I; CHR$(I);
40 NEXT
50 END
```

3. Graphic Characters

```
10 CLS
20 FOR I=128 TO 191
30 PRINT@ (I-128)*8,I; CHR$(I);
40 NEXT
50 END
```

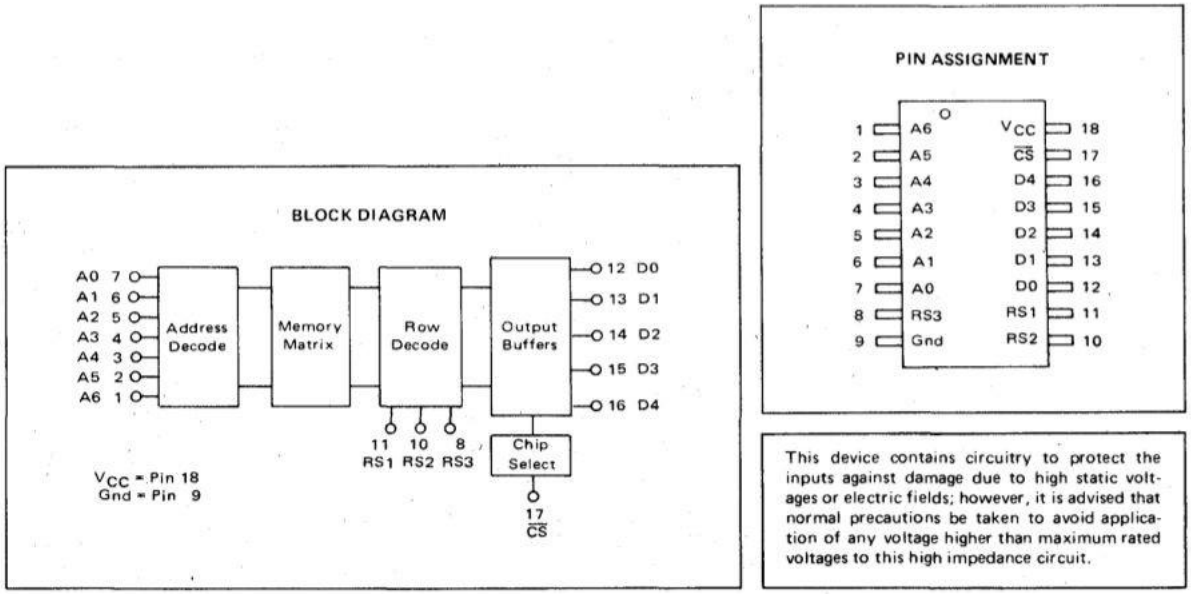
Special Characters

```
5  CLS
10 POKE 16526,105
20 POKE 16527,0
30 X=USR(O)
40 PRINT CHR$(21);
50 INPUT "PRESS ENTER TO SEE SPECIAL CHARACTERS";X
60 FOR I=192 TO 255
70 PRINT CHR$(I);
80 NEXT
90 PRINT
100 INPUT "PRESS ENTER TO SWITCH TO ALTERNATE SET";X
110 PRINT CHR$(22);
120 INPUT "PRESS ENTER TO RETURN TO NORMAL AND END";X
130 PRINT CHR$(22); CHR$(21)
140 END
```

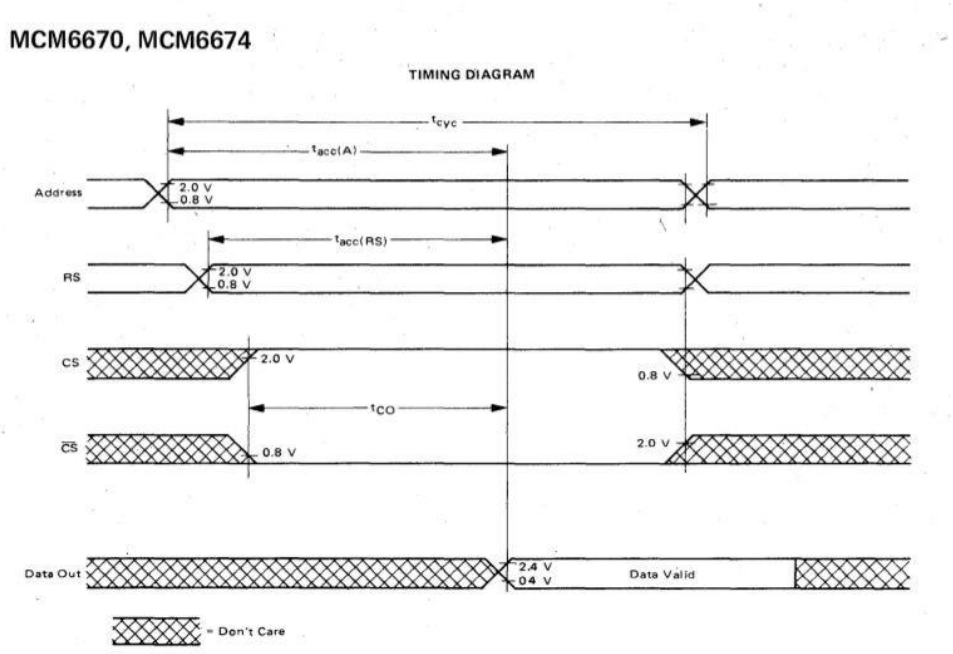
The MCM6670 is a mask-programmable horizontal-scan (row select) character generator containing 128 Characters in a 5x7 matrix, with a 350ns access time.

A 7-bit address code is used to select one of the 128 available characters, and a 3 row select chooses the appropriate row to appear at the outputs. The rows are sequentially displayed, providing an 8-word sequence of parallel bits per word for each character selected by the address inputs.

The MCM6674 is a reprogrammed version of the MCM6670.

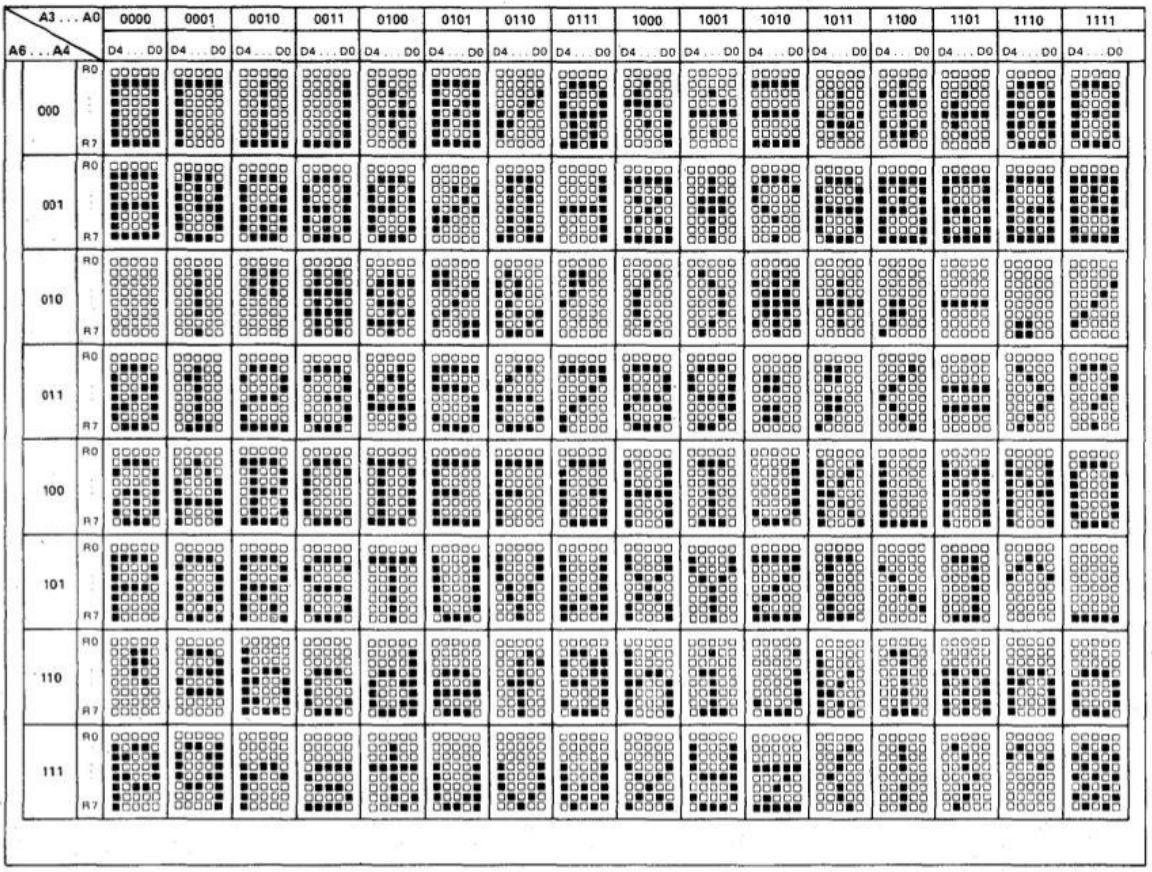


Here is the timing diagram for the Character Generator's access time.



For each address {A0..A3} (2^4 = 16 = {0x0..0xF}), there are eight rows {A4..A6} (2^3 = 8 = {R0..R7}) of bits that are read for the actual character.

The complete table is shown below:



I used an Arduino Mega 2560 to read my TRS-80 Model 1's, G with Ver 1.3 ROMS, Character generator into hex bytes.

The Arduino has PULLUP resistors enabled on the Data lines which makes the MSB (xxx11111) incorrect.

Here is an example of the data:
E0FFF1F1F1F1F1FFE0FFF0F0F0F0F0F0
E0E4E4E4E4E4E4FFE0E1E1E1E1E1E1FF
E0E8E4E2EFE4E2E1E0FFF1FBF5FBF1FF
E0E0E1E2F4F8F0E0E0EEF1F1FFEAEAFB

This requires each data byte to be ANDED with 0x1F to mask the PULLUP's incorrect data.

Since I use Linux, I knew that this was a prefect job for srec_cat. srec_cat is an invaluable tool that can manipulate the ROM data accordingly.

For more information in Linux, use:
man srec-info
man srec_cat

The URL's of:
http://srecord.sourceforge.net/
http://srecord.sourceforge.net/man/man1/srec_examples.html

give more information on the data types, and the command specifics. There are also lots of examples.

I started by letting srec_cat & srec_info try to detect the file's data type. I knew the data was HEX bytes.

I tried: srec_info 6670tst.rom, but, this assumes the default of Motorola S-record format, which is incorrect.

Next, I tried: srec_info 6670tst.rom -guess but, srec_info's -guess is incorrect again.

srec_info 6670tst.rom -nh -ignore_checksums finds the correct data:
Format: Needham
Data: 0000 - 03FF

The first four lines of data from the MCM6670.txt:
E0FFF1F1F1F1F1FFE0FFF0F0F0F0F0F0
E0E4E4E4E4E4E4FFE0E1E1E1E1E1E1FF
E0E8E4E2EFE4E2E1E0FFF1FBF5FBF1FF
E0E0E1E2F4F8F0E0E0EEF1F1FFEAEAFB

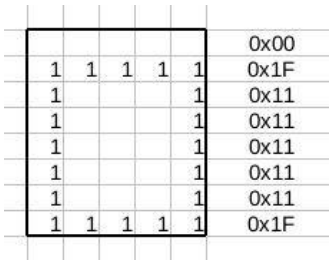
shows the data has pull up resistors on the three MSD bits ({5..7}) and needs to be ANDED with 0x1F.

This command will do exactly what I need:
srec_cat infile -nh –and 0x1F –o outfile -nh -ignore_checksums -data_only

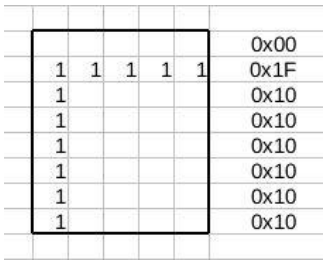
Now the actual data is:
00 1F 11 11 11 11 11 1F 00 1F 10 10 10 10 10 10
00 04 04 04 04 04 04 1F 00 01 01 01 01 01 01 1F
00 08 04 02 0F 04 02 01 00 1F 11 1B 15 1B 11 1F
00 00 01 02 14 18 10 00 00 0E 11 11 1F 0A 0A 1B

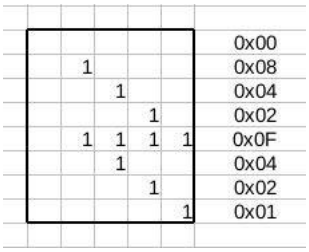
The first eight bytes are Character #0 and the next eight bytes are Character #1.

0x00
0x1F
0x11
0x11
0x11
0x11
0x11
0x11
0x1F

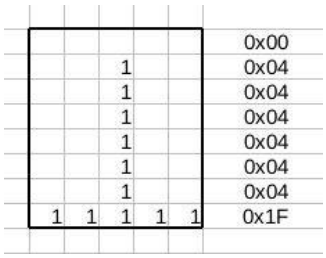


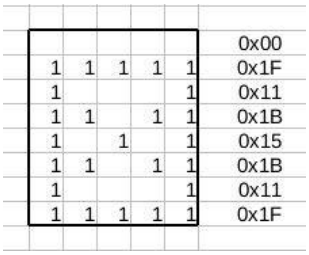
0x00
0x1F
0x10
0x10
0x10
0x10
0x10
0x10
0x10



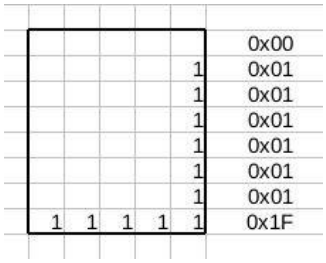


0x00
0x04
0x04
0x04
0x04
0x04
0x04
0x04
0x1F





0x00
0x01
0x01
0x01
0x01
0x01
0x01
0x1F



I copied the data for all 128 Characters into Excel, and looked at each of the 5x7 Characters.

Larry Kraemer

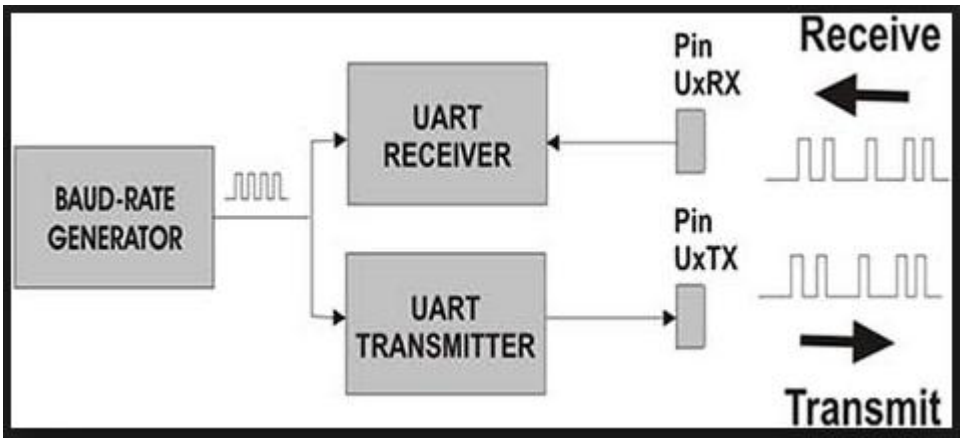
Larry Kraemer is an ex-Military USAF Officer, and a retired Electronics & Instrumentation Technician, that has a hobby of Amateur Radio (Advanced Class), and Computers, with Computer Repair experience.

IN MAV'S WORKSHOP

FIXING YOUR MODEL III / 4 / 4P / 4D SERIAL PORT

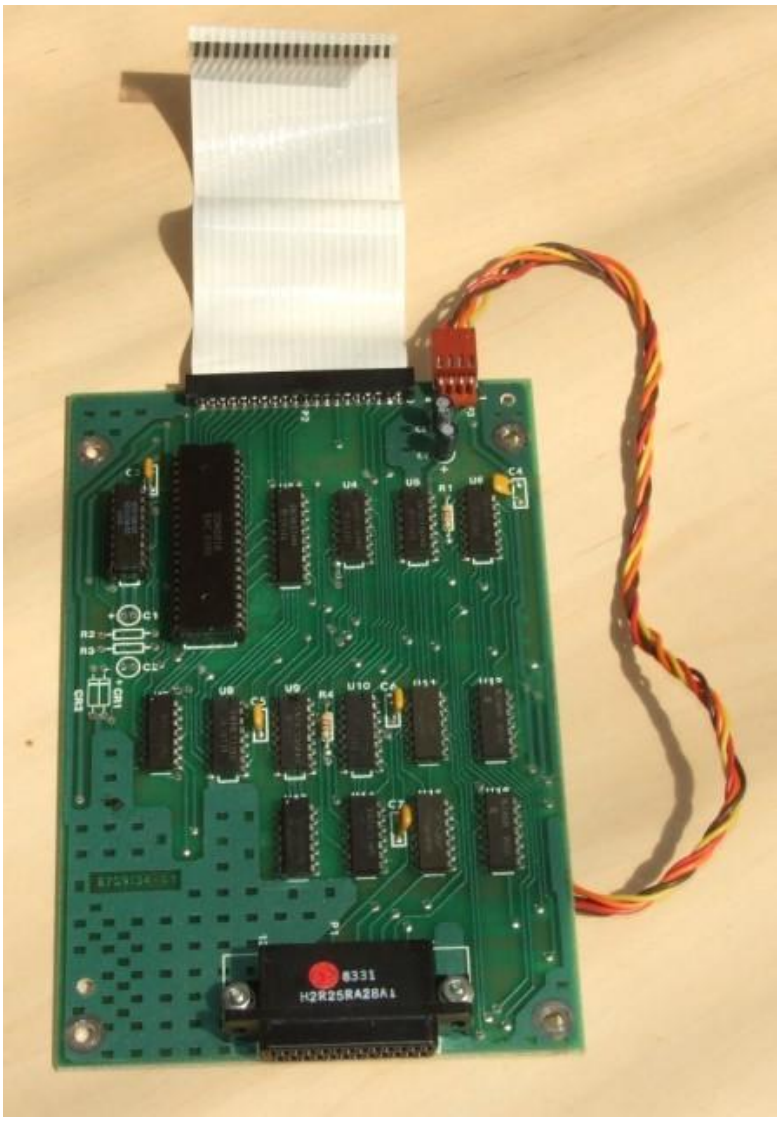
Ian Mavric

Back in the September 2013 issue of TRS8Bit I wrote a two-part article about repairing the Model I RS232 serial board. With the imminent release of Peter Cetinski's TRS-Box Serial program around the corner, now is a good time to look at the serial port in your TRS-80 to make sure it works properly. A failed serial port will not prevent your TRS-80 from running. In fact it may have failed back in the 1980s and if you've never needed to use it you would never know. The rest of the TRS-80 doesn't care. Fortunately the RS232 is easy to understand and diagnose if you have basic electronics skills, the serial ports in the Model I/III/4/4P/4D are of the same basic design:



Model III and Model 4 Non-Gate Array

The Model III and non-gate array Model 4 have a separate RS232 serial board connected to the motherboard by a 20-way flat cable, and a power cable. These cables rarely give troubles but if you damage the 20-way flat cable it can be difficult to find a replacement. Installing a pin header and running an old IDE hard drive cable between the motherboard and the serial board is a good fix. The power cable rarely gives problems, but remember the serial board uses 4 wires: 5V, Gnd, 12V and -12V. All 4 are required for the serial board to work. If you plug the disk controller power cable (which only uses 3 wires) into the serial board the system will not work. When testing, look for the above voltages at the 4 pins before proceeding.



Picture: a Model III or 4 serial board - the big chip is the UART

The serial board itself has 16 chips but of those only three of them tend to give problems. 10 of the chips are 74LS types which are extremely robust and in fact I have never seen one fail in all my years working on TRS-80s. We will look at the UART, the BRG, and one of the line drivers.

A serial port takes parallel data and coverts it into a stream of 0s and 1s. It does this in both directions, so data can be "sent" from a computer in which case parallel 8-bit "words" are converted to a serial stream of 8-bits and out the serial port. Since the serial port is designed around communication, it also has to have the ability to "receive" 8-bits of serial data and covert that back into 8-bit parallel "words" the computer can work with.
This is achieved by a 40-pin chip called a UART which stands for Universal Asynchronous Receiver/Transmitter. It's the big socketed chip on the serial board. It's one of the first chips I check and roughly 1/3rd of the time it's responsible for the serial port not working.

The UART needs to work at a set speed to receive its serial data incoming and outgoing so it can be matched to that of the serial device it's trying to communicate with. Speed/timing is critical here as if the split second timing of receiving or sending serial bits goes out of sync by only 1/100th of a second you can tell bits will be missed or misinterpreted. The speed at which serial communications occur are set rates known as Baud rates, and because different serial devices, depending on their sophistication and distance from the terminal they are communicating with, have a choice of a number of speeds.

To accomplish this we use a chip called a Baud Rate Generator (BRG) which lives in a socket next to the UART. It can be set for any baud rate between 110 and 19200, however the most common rates used are 300, 1200, 2400 and 9600. These chips work together, the BRG tells the UART how fast to run and keeps it in perfect time. The BRG, like the UART fails about 1/3rd of the time in my experience.

Finally we come to the line drivers, these are gateways between the TTL logic of the computer and the generally higher voltage expectations of the connections to the outside world. Basically the TTL signals inside the computer are not suited to long distance wires due to their characteristics so they need to be amplified up to meet EIA Standard RS-232D and CCITT Recommendation V.24.

Obviously the reverse is also true, the higher voltage/amperage signals coming into the serial port need to be reduced to TTL levels palatable with the computer. Output is accomplished with a pair of MC1488 line drivers and input is accomplished with a pair of MC1489 line receivers. In my experience the MC1489s never give any problems but the MC1488s regularly fail. Why would this be? As best I can tell when a malfunctioning or badly designed serial device is connected to the serial port, the MC1488s are the first chips in the firing line. They also have a very large operating range of ±30V, and are powered by 12V. They just have a hard life. They fail the last 1/3rd of the time in my experience.

"Isn't there a simple easy way to fix these?"

Well yes, this is how I do it... since I have a soldering rework/de-soldering station and many spare parts on hand, I change out the UART and BRG for known good ones, and if it doesn't work then I de-solder U12 and U16 (both MC1488s) put in a couple of sockets and new MC1488s.
99.9% of the time this will fix the serial board. However not everyone has spare UARTs and BRGs lying around, or the de-soldering station to change a chip in 1-2 minutes so most people need to diagnose the problem properly.

Diagnosing Serial Port Problems

So you've connected up a serial device or done a simple loop-back of pins 2 and 3 at the RS232 connector and nada... nothing happens or the systems hangs. To get started you need a terminal program such as MAL/CMD, or better still the RSCC Technician Series Diagnostic Disk. Both can be downloaded from my downloads page: http://ianmav.customer.netspace.net.au/trs80/downloads.htm

Rig up a loop of wire connecting pins 2 and 3 on the RS232 connector, this joins the transmit pin to the receive pin so anything you type on a terminal program is echoed back to the computer. Connect a logic probe or a CRO to the exposed wire running between pins 2 and 3. Load your terminal program and type some letters. Watch for activity on the logic probe or CRT screen.

If you see activity then the line drivers are OK but one of the line receivers is not working, so replace one or both MC1489s. However like I stated earlier, the line receivers rarely give problems so you are much more likely to see no activity on the looped wire. If the line drivers are not working nothing it output so nothing can be read into the line receivers. The rest of the article is written in jargon.

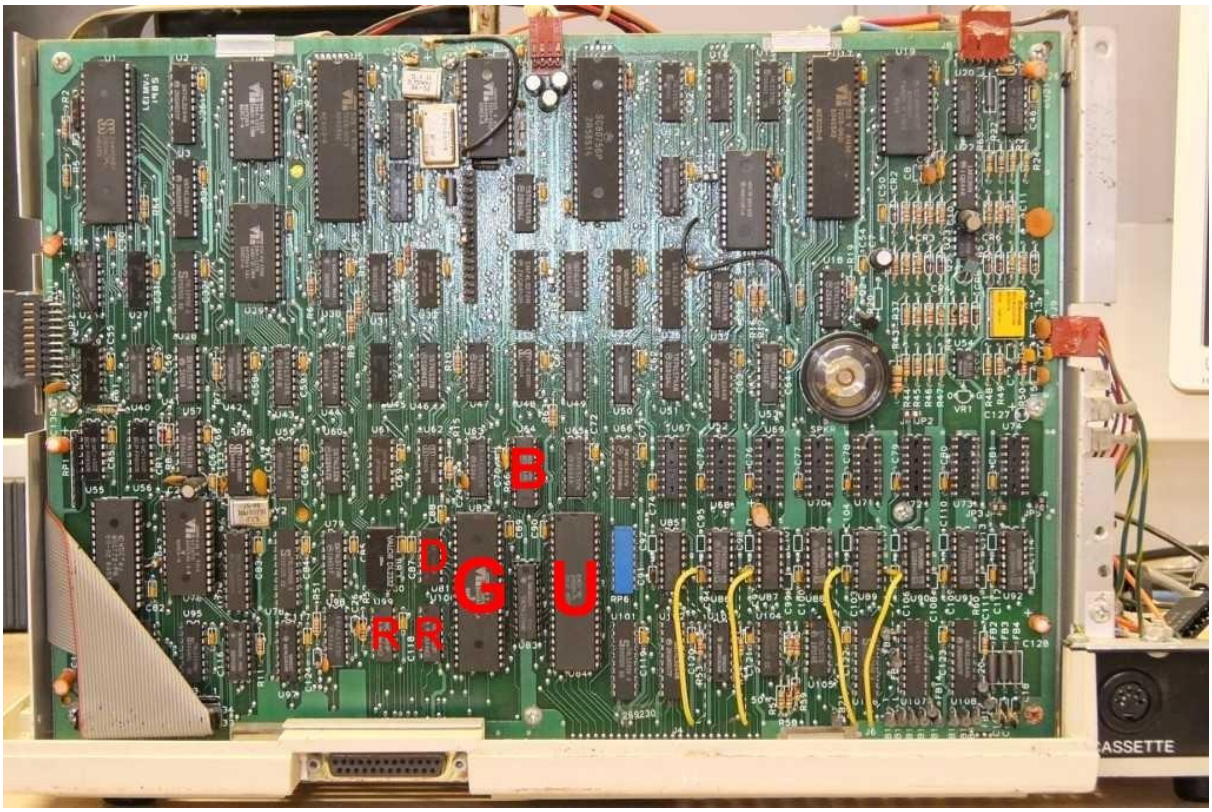
DB25 P2 is Rx and P3 is Tx. Rx connects to U16 P11. P11 is an output of an MC1488 and it's inputs are P12 and 13. Look for activity on P12 and 13, if there is activity then everything before it which includes the UART and the BRG are sending data to U16 but the output of the nand gate formed by P11,12,13 is busted. Replace U16.

If the above checks out, but data is still not echoed back, follow Tx to U11 P10. Look for activity on P8 and P9. If there is activity on P10 and 9 but not P8 then you have found your bad gate. U11 is an MC1489 - a line receiver - it sends the data back to the UART.

To check the UART look for activity on U2 P20 (Rx input) and P25 (Tx Output Register). Load a terminal program and look on these pins as you type a character, they should pulse. If the UART (usually a TR1865) is bad it can be replaced with a TR1602 or a COM8018. To check the BRG look for activity on U1 P8 and P12 for activity, these are the strobes to and from the UART.
Note: the BRG, as used on a Model III/4/4P/4D uses its own internal oscillator to derive the frequencies for each baud rate. As used in a Model I, a Xtal is connected to P1 and P18 so if diagnosing a bad BRG on a Model I, look for activity on those pins, though it's far more likely that the Xtal itself is bad on the Model I RS232 board than the BRG itself. In all cases, if the BRG is bad, replace it with a COM8116 which is a plug in substitute for the old WD BR194x series of 1970s BRGs.

Model 4 Gate-Array and 4D

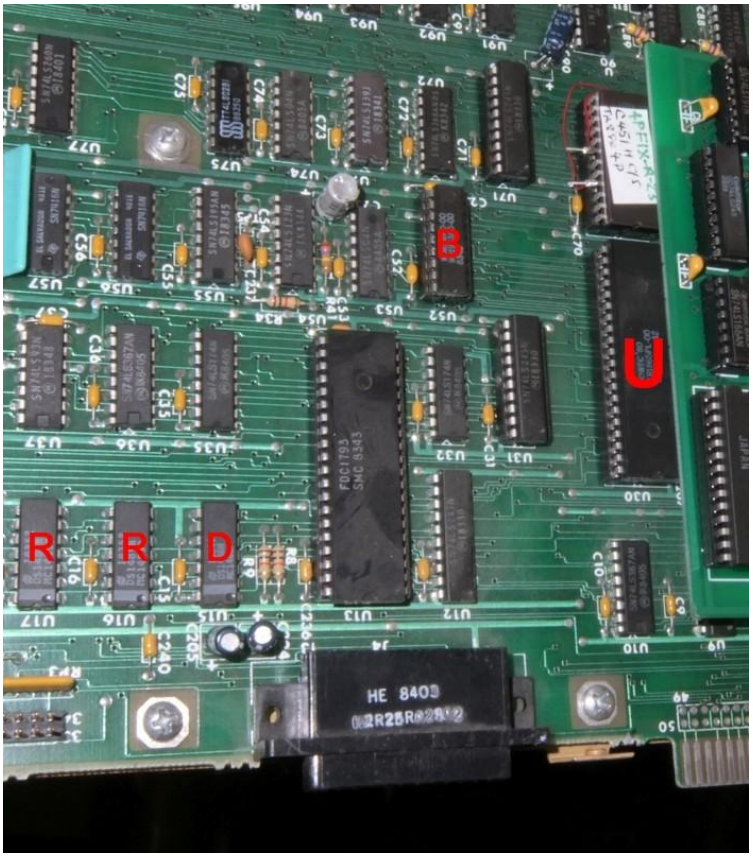
Cost cutting when designing the second generation of Model 4 (what we call the Model 4 Gate-Array or M4GA) meant that everything which used to be on three circuit boards is now put on a single board. On the downside it also means that whole serial board substitution is not possible and the only way to fix the serial port on a M4GA is to repair it. Using the same technique outlined above, you can solve the problem, however in the case of the M4GA the BRG is U64 and the UART is U84. The line driver is U81 and the line receivers are U99 and U100. Additionally, the M4GA has a Gate-Array chip (GA 4.5, U82, called the RS232 support GA), which cannot be easily tested, only by chip substitution. On the positive side I've never once seen this chip fail. The information pertaining to the M4GA also applies to the 4D because that is essentially a dressed up M4GA with double-sided disk drives.



Picture: M4GA/4D motherboard. Legend: U-UART, B-BRG, D-Line Driver, R-Line Receivers, G-4.5 support G-A

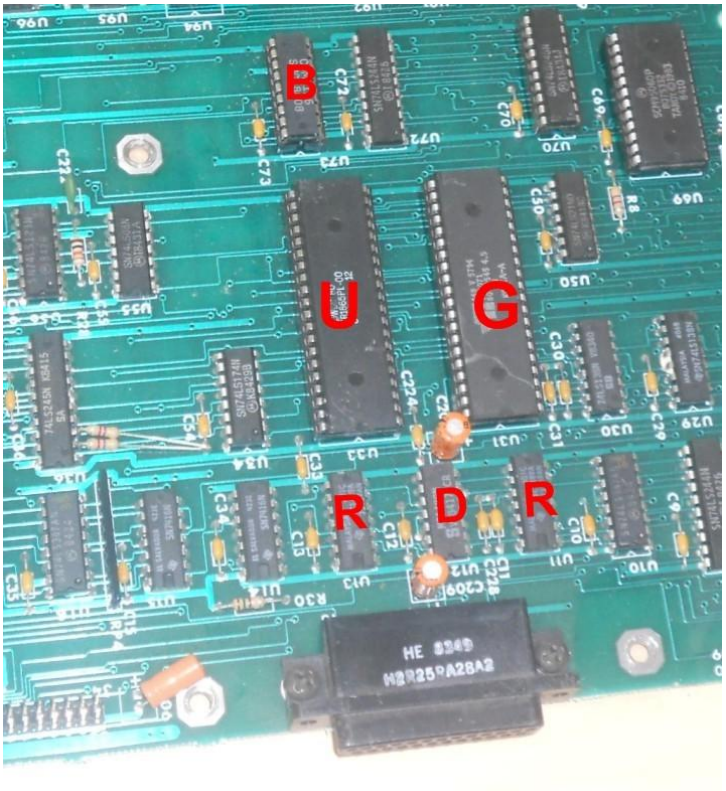
Model 4P

As we know they produced two 4P motherboards, the 4P Non-Gate and the 4P Gate-Array. On the 4PNGA the relevant parts are UART: U30, BRG: U52, Line driver: U15, Line Receivers: U16 and U17.



Picture: 4PNGA serial port. Legend: U-UART, B-BRG, D-Line Driver, R-Line Receivers

On the 4PGA the parts to look for are UART: U33, BRG: U73, Line Driver: U12, and line Receivers U11 and U13. Once again we have an RS232 support Gate-Array, and in this case it us U31.



Picture: MPGA motherboard. Legend: U-UART, B-BRG, D-Line Driver, R-Line Receivers, G-4.5 support G-A

"My Model III or 4 does not have a Serial Board?"

If your Model III started out as a 4K Level I, 16K Model III BASIC, or a 26-1065 single disk drive computer it may have no serial port as these were sold as an option for these machines. The same goes for the Model 4 16K and 26-1068 single disk drive computer.

Tandy sold an upgrade option (cat. no. 26-1148) which comprised of the serial board and is necessary internal connection cables. I have a few of these NOS if people need them, I also have some refurbished 2nd hand serial boards which I also sell with the necessary interconnection cables. Lastly I also offer a repair service if you can't get your Model III or 4 serial board to work simply send it to me and I'll repair it. Normal computer repair rates apply.

"A video on how you fix these would be handy?"

A few years ago I made a video about how I repaired a Line Driver on a Model 4 gate-array motherboard, you can find it on YouTube: https://youtu.be/aBLaQIrwQLU

It also shows usage of the RSCC Diagnostics suite to test the RS232 functioning. Check it out.

Next time: Serial cable for the Model I

Ian Mavric

ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

I KNOW THE GUYS AT NIGHTFALLCREW.COM HAVE BEEN BUSY CLEARING THE 'SNOW' FROM M1 VIDEOS TO ASSIST ALL YOU GAME PLAYERS OUT THERE. HERE IS THE ORIGINAL ARTICLE BY BRIAN SMITH, WHICH FIRST APPEARED IN THE MARCH 1982 EDITION OF 80-MICRO.

BAS, AT BETAGAMMA COMPUTING VERY KINDLY DID THE MOD ON MY OLD, BUT JUST AS WONDERFUL AS EVER, M1. IT STILL WORKS OK WITH MY 'QUINNTERFACE' AND FREHD AND THERE IS A NOTICEABLE IMPROVEMENT IN THE SCREEN-REFRESH WHEN I PLAYED 'ATTACK FORCE'

Get rid of those white streaks.

Video Snow Shovel

Brian Smith
6770 Halifax Street
Burnaby, BC Canada
V5B 2R4

Video snow, the black streaks against white characters, is most apparent with large, rapidly changing areas of white graphics.

The root of the problem is that the TRS-80 display is memory mapped by the CPU. In addition, the display memory is continually scanned by the video circuitry. This circuitry reads the display memory and translates it into characters on the screen. The CPU and video circuitry cannot access the display memory at the same time: The CPU has higher priority. When the CPU reads or writes part of the display memory, the video circuitry is locked out. The result is a black streak, or glitch, on the screen.

For a more complete description of display circuits' operations read Radio Shack's *TRS-80 Microcomputer Technical Handbook.*

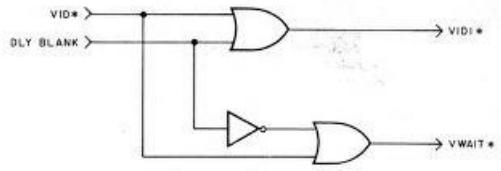


Fig. 1. Conceptual Circuit.

The Solution

My solution is to give the video circuitry higher priority than the CPU.

The display is made up of 192 lines of 384 dots per line. Each character is 12 rows of 6 dots. The maximum display is 16 lines of 64 characters. After displaying a full line of dots the electron beam in the picture tube turns off and moves back to the left side of the screen to begin a new line. When the bottom line is complete the electron beam returns to the top left corner. The TRS-80 maintains black borders along all four sides. I will call these periods when the electron beam is not displaying anything blanking periods.

When the CPU accesses display memory during a blanking period we do not see any black streaks. The snow removal solution is simple: Let the CPU read and write the display memory only during blanking periods. In other words, reverse the priorities of the CPU and video circuitry for reading the display memory.

By using the WAIT* input to the CPU, access is blocked and the CPU is put into a wait state until the video circuitry enters a blanking period. Then the CPU is released and allowed to continue.

This does not slow down faster real-time graphics games.

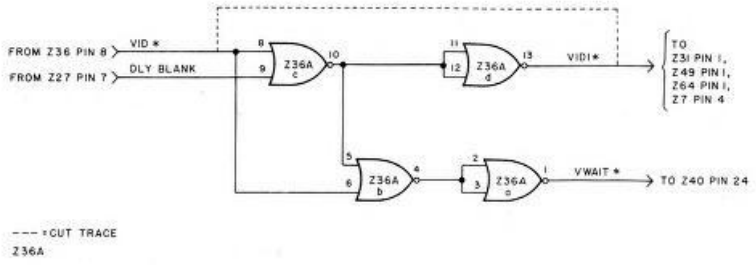


Fig. 2. Version 1 Schematic.

Even the most intensive graphics program spends little time accessing the display memory. Games that combine intensive graphics with simultaneous sound generation suffer slightly. The sound quality deteriorates because the wait states change timing loops.

The Circuit

Fig. 1 shows a conceptual version of the video de-glitcher circuit. It functions identically to the actual circuit and is easier to understand. DLY BLANK is high when characters are displayed and low during blanking periods. VID* goes low when the CPU addresses the display memory. Normally, VID* goes directly to the video RAM selectors, allowing the CPU to access the display memory. With the de-glitcher installed, the new signal (VID1*) decides whether the CPU or video circuitry can access the display memory. VID1* is low only when both VID* and DLY BLANK go low. The CPU can access the display memory only during blanking periods.

VWAIT*, when low, puts the CPU into a wait state. It goes low when VID* is low (the CPU is trying to access the display memory), and DLY BLANK is high (the video circuitry is displaying characters). With VWAIT* high the CPU operates normally.

The actual circuit, (Fig. 2 and 3) is less obvious than Fig. 1. Fig. 2 is the simple version and Fig. 3 includes a switch to turn the de-glitcher off. The second version is for those who want no sound distortion in their graphics and sound programs.

To use other modifications or peripherals with the WAIT* input, add a tristate buffer to either of the circuits in Fig. 2 and 3. The buffer removes the VWAIT* signal from the WAIT* line until VWAIT* goes low. If this is not done, the de-glitcher circuit will fight with other devices over whether the WAIT* signal should be low or high. Any other devices using the WAIT* input should also be isolated with a tristate buffer. (See Fig. 4).

"Let the CPU read and write the display memory only during blanking periods."

On the simple version (Fig. 2), the buffer is added between Z36A pin 1 and Z40 pin 24. On the switchable version (Fig. 3) it is installed between Z36A pin 1 and S1 point F.

Installation

The following are step by step instructions for installation of the video de-glitcher. Since only one or two integrated circuits are added, I use piggy-back installation rather than constructing a circuit board.

- ● Open the case and place the main circuit board, component side up, on a flat surface.
- ● Locate Z36 on the board.
- ● Bend all pins except 7, 8, and 14 on the new 74LS02 (Z36A) into a horizontal plane.
- ● Install Z36A on top of Z36, soldering pins 7, 8, and 14 of Z36A to the corresponding pins on Z36 (see Fig. 5).
- ● Solder a wire between pins 5 and 10 of Z36A.
- ● Solder a wire between pins 10 and 11 of Z36A.
- ● Solder a wire between pins 11 and 12 of Z36A.
- ● Solder a wire between pins 8 and 6 of Z36A.
- ● Solder a wire between pins 2 and 3 of Z36A.
- ● Solder a wire between pins 3 and 4 of Z36A.
- ● Solder a wire from Z27 pin 7 to Z36A pin 9.
- ● Cut the trace between Z36 pin 8 and Z31 pin 1 (see Fig. 7).

Depending on which version of the circuit you use follow the indicated steps in Fig. 9.

A1. Solder a wire from pin 13 of Z36A to the point indicated in Fig. 8.
A2. Solder a wire from pin 1 of Z36A to the point indicated in Fig. 6.

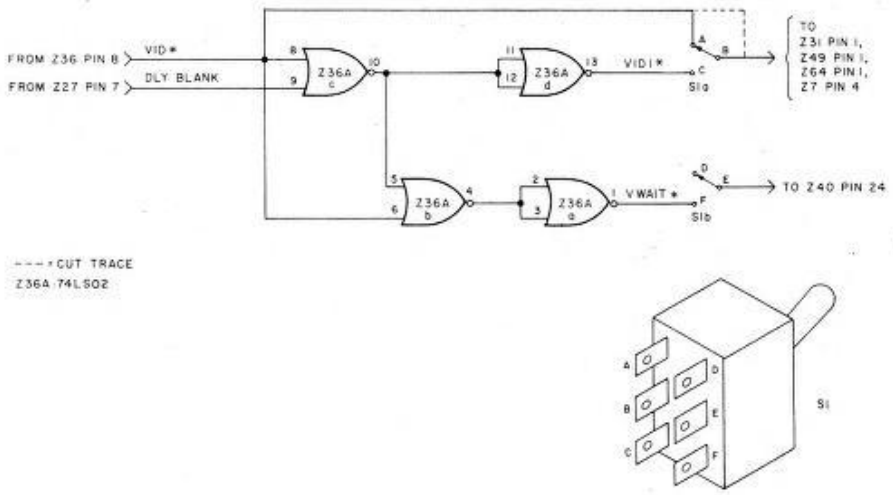


Fig. 3. Version 2 Schematic.

B1. Solder a wire from pin 13 of Z36A to the point indicated in Fig. 8.
B2. Locate Z75 on the circuit board.
B3. Bend all pins except 7 and 14 of the new 74LS125 (Z75A) into a horizontal plane.
B4. Install Z75A on top of Z75, soldering pins 7 and 14 to the corresponding pins on Z75.
B5. Solder a wire between pins 1 and 2 of Z75A.
B6. Solder a wire from pin 1 of Z36A to pin 2 of Z75A.
B7. Solder a wire from Z75A pin 3 to the point indicated in Fig. 6.

C1. Install the new switch in a suitable place on the TRS-80s case.
C2. Solder a wire from pin 8 of Z36A to point A on the switch.
C3. Solder a wire from pin 13 of Z36A to point C on the switch.
C4. Solder a wire from point B on the switch to the point indicated in Fig. 8.
C5. Solder a wire from pin 1 of

Z36A:	74LS02
275A:	74LS125 (optional)
S1:	DPDT toggle switch (optional)
	Thin-gauge insulated wire, such as wire wrap wire

Table 1. Parts List.

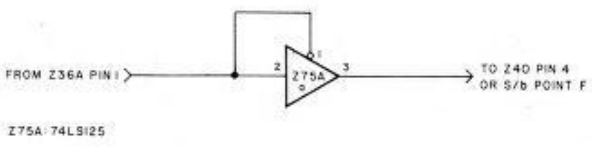


Fig. 4. Buffer Circuit.

Z36A to point F on the switch.
C6. Solder a wire from point E on the switch of the point indicated in Fig. 6.

D1. Install the new switch in a suitable place on the TRS-80s case.
D2. Solder a wire from pin 8 of Z36A to point A on the switch.
D3. Solder a wire from pin 13 of Z36A to point C on the switch.
D4. Solder a wire from point B on the switch to the point indicated in Fig. 8.
D5. Locate Z75 on the circuit board.
D6. Bend all pins except 7 and 14 of the new 74LS125 (Z75A) into a horizontal plane.
D7. Install Z75A on top of Z75, soldering pins 7 and 14 to the corresponding pins on Z75.
D8. Solder a wire between pins 1 and 2 of Z75A.
D9. Solder a wire from pin 1 of Z36A to pin 2 of Z75A.
D10. Solder a wire from pin 3 of Z75A to point F on the switch.
D11. Solder a wire from point E on the switch to the point indicated in Fig. 6.

The video de-glitcher is now installed. Double check your wiring and reassemble the computer.

Testing

Power up to begin testing. If you do not see a normal display, *immediately* turn the machine off and recheck your wiring. If everything looks fine run any program with video glitches or type in the following program:

```
10 'QUICK VIDEO DE-GLITCHER TEST
20 CH = 0
30 FOR I = 15360 TO 16383 'VIDEO
MEMORY ADDRESSES
40 POKE I,CH
50 CH = CH + 1
60 IF CH > 255 THEN CH = 0
70 NEXT I
80 GOTO 20
```

You should see no snow on the screen with this program running. If you installed Version 2, try switching the de-glitcher on and off. Note the difference in the displays. If you still see snow, the circuit is not working properly. Go back and check your wiring.

Postscript

After designing and installing the video de-glitcher, I looked at a Model III schematic. It uses a technique similar to mine to eliminate snow on the new machine.■

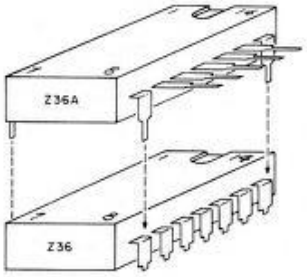


Fig. 5

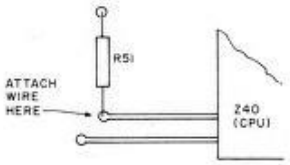


Fig. 6

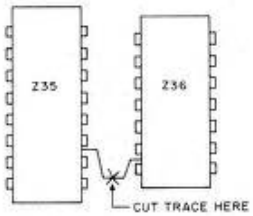


Fig. 7

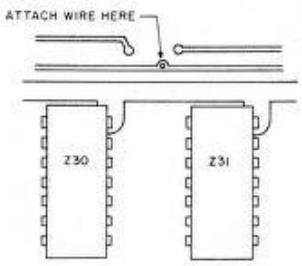


Fig. 8

Version 1 without buffer circuit:	Follow steps starting with A.
Version 1 with buffer circuit:	Follow steps starting with B.
Version 2 without buffer circuit:	Follow steps starting with C.
Version 2 with buffer circuit:	Follow steps starting with D.

Fig. 9

See List of Advertisers on page 402

IN MAV'S WORKSHOP

FreHD AUTO BOOT EPROM PART 3: HOW TO INSTALL IT ON THE MODEL 4 G-A AND 4D

Ian Mavric

In previous issues of TRS8Bit I've discussed how to install the FreHD Auto-boot Eprom into a TRS-80 Model III and Model 4NGA (easy - just remove an old Rom chip and replace it with a newly programmed Eprom), and a Model I (difficult - requires soldering, and a switch as well as 6 wires).

This article explains how to add the FreHD auto boot Eprom to a Model 4 Gate-Array computer (cat. no. 26-1068A or 26-1069A) or a Tandy Model 4D (cat. no. 26-1070). It's more difficult that adding the Eprom to the Model III and 4NGA but less difficult than adding one to a Model I. Soldering is required so as I always say, if you are not confident in soldering, get someone else who is skilled to undertake this modification for you. Since no traces need to be cut the modification is easily reversible if things go wrong.

Items required:

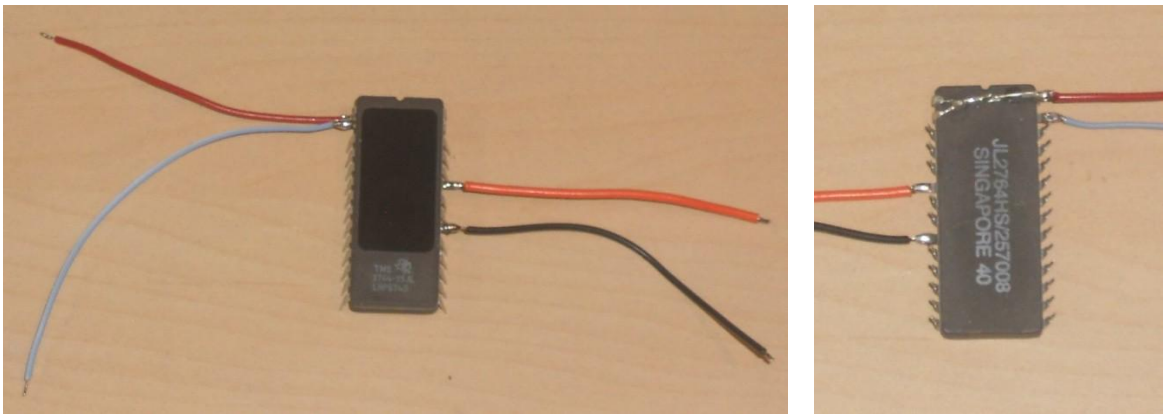
Eprom: 2764
Wires: Black (10cm long), Blue (10cm long), Orange (10cm long), Red (5cm long)

Firstly you need to program the Eprom with this BIN file from the Downloads page on my web site:
http://ianmav.customer.netspace.net.au/downloads/model4romgen2-BC.bin

Once programmed you need to do the following work to prepare the Eprom for installation in your computer:

Bend out pin 2 and solder the blue wire to it.
Bend out pin 20 and solder the back wire to it.
Bend out pin 23 and solder the orange wire to it.
Bend pins 1, 27 and 28 under the chip and solder the red wire in a manner which connects all three.

It should look like this:



Pictures: top (L) and under (R) of the 2764 Eprom. Notice pins 1,27 and 28 are joined to the red wire.

If you haven't noticed yet, it will quickly become clear that we are replacing a 24-pin Rom with a 28-pin Eprom. We will be replacing U4, which is up in the top left corner of the motherboard.

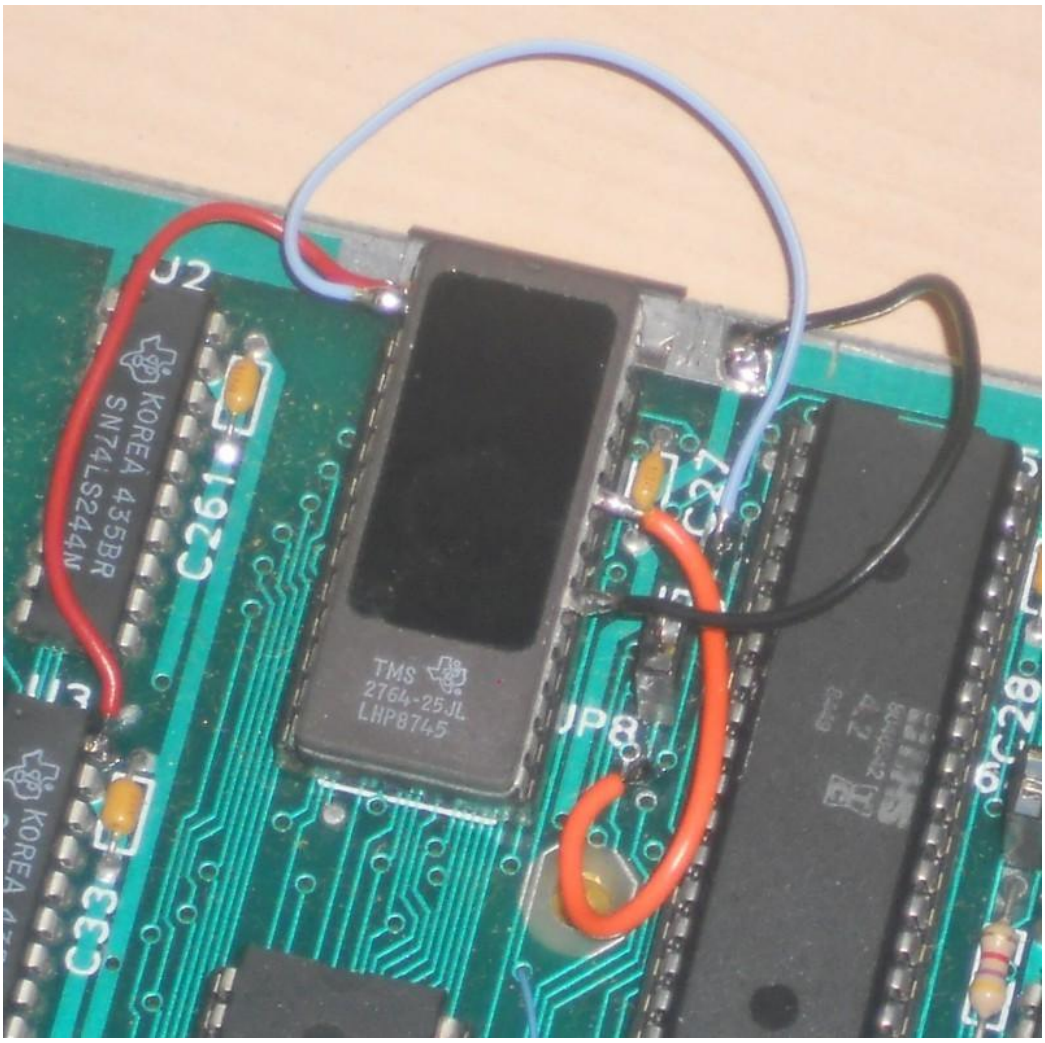
While it is possible to make this modification with the motherboard left in place, I prefer to remove it, which makes soldering the wires rather easier.

Picture: completed modification and motherboard ready to re-install in the computer.

Once U4 has been removed, the programmed and modified 2764 is inserted into the socket with the 4 pins at the front (pins 1,2,27,28) overhanging at the front.

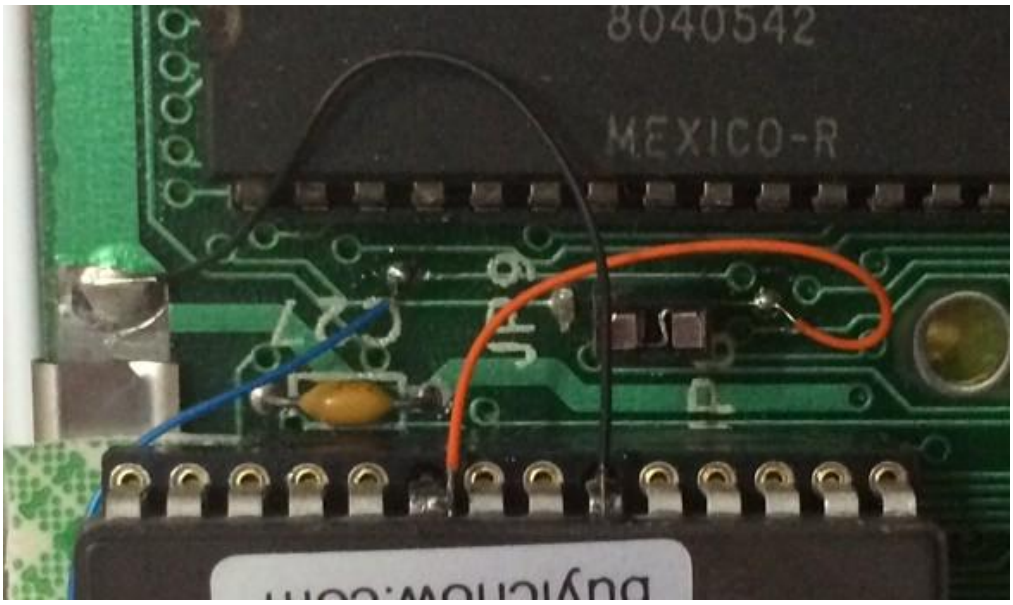
Make the following connections:
Solder the red wire U3 Pin 20 (this is the 5V source powering the Eprom)
Solder the black wire to the ground rail which runs across the top of the motherboard, near the clip which holds the RFI shield behind the motherboard. Note: the shield does not need to be removed.
Solder the orange wire to the pad directly below (and joined to) JP8*.
Solder the blue wire to the pad nearest the C in C29 silkscreen and directly above the 9 in JP9*.

* You may need to clean away some of the green mask covering the points needed to solder the orange and blue wires to. Also, I find it easier to remove U5 and the JP8 jumper when soldering these two wires. It should look like this:



Picture: shows the completed modification.

Here is a better close-up of the black, blue and orange wires from Dean Bear (thanks Dean) clearly shows where the wires need to go:



Picture: wire-routing close-up. (Photo courtesy of Dean Bear)

Re-assemble the motherboard back into your computer and test the system in the normal manner first to make sure it runs as a floppy-based system and boots properly. If everything seems right, connect your FreHD and download the starter image for the Auto Boot Model III and 4.

Pros / Cons

The main Pro of this modification is that since it doesn't protrude any higher than the Rom chip removed, there is still plenty of room to install a Hi-Res graphics board if needed.
Also the aluminium RFI shield can be re-installed without any special sheet-metal acrobatics needed to make it fit.

The main Con of this modification is that you lose the ability to drop to Model III Rom Basic (ie. Cass? and Memory Size?) which may or may not be important depending on how you use your computer.
You can still load a DOS, invoke Basic language and CLOAD tapes if needed however most software is available already online transferred from tapes and ready to use.

NF6X little adapter board

For those who wanted the ability to go back and boot Model III Rom Basic, Mark Blair (a.k.a NF6X) designed a neat little PCB which holds both the original ROM and the new Eprom, and allows you to switch between them with a switch accessible through the cooling vent on the top of the computer. You need to get your own PCB made (typically through Osh Park) and solder the sockets, pins and switch as needed. For more information check out Mark's web page on the subject:

http://www.nf6x.net/2013/11/version-2-of-trs-80-model-4-rom-adapter/

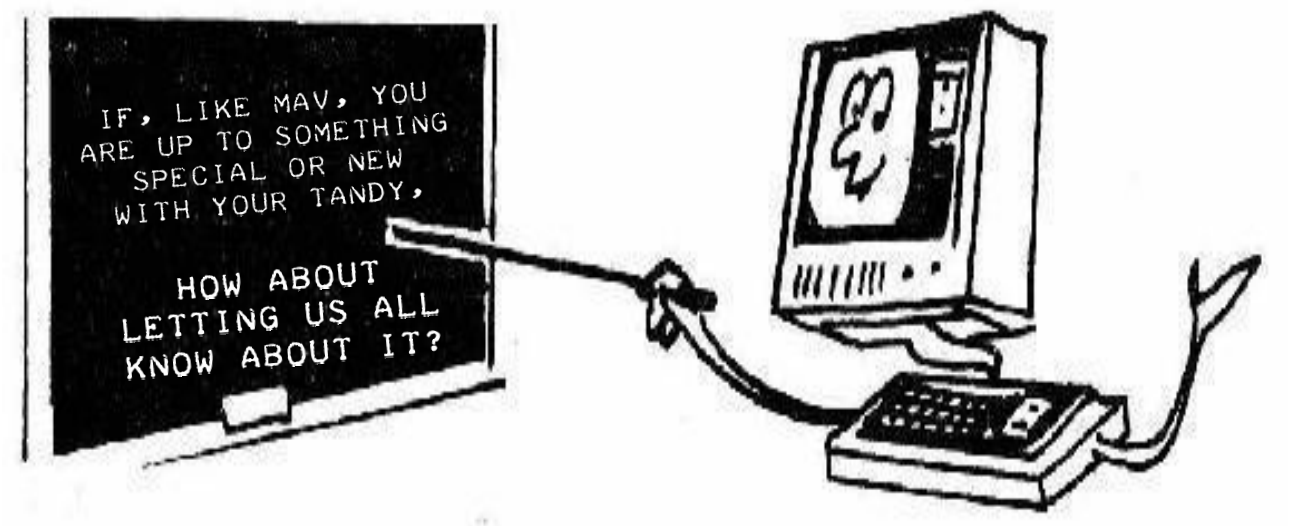
Please be aware that NF6X doesn't and won't make or sell these, and the project is posted for those who have the ability to assemble and install it themselves. Bugging NF6X about buying one will result in the same reply.

Next time: FreHD Auto Boot on the Model 4P

Ian Mavric

ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.



The *BEST* in TRS-80s

Call The Right Stuff

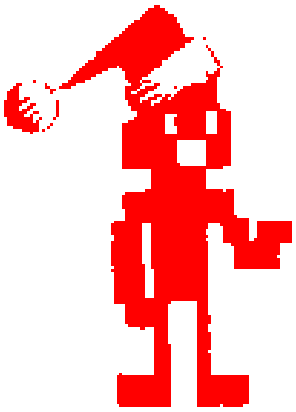
Ask for Ian
The number is +61 416 184 893

That's The Right Stuff
And he's in Melbourne

http://ianmav.customer.netspace.net.au/trs80/

TRS8BIT

PRODUCED BY TRS-80.ORG.UK



DEE AND I WISH YOU ALL A VERY MERRY CHRISTMAS AND A HAPPY NEW YEAR.

I CAN'T BELIEVE JUST HOW QUICKLY IT'S COME 'ROUND AGAIN!
(SOMETHING TO DO WITH GETTING OLDER I'M TOLD!)

ONCE AGAIN, I'M PLEASED TO REPORT THAT THERE IS A MASSIVE AMOUNT OF DEVELOPMENT HAPPENING ON THE TRS-80 FRONT.

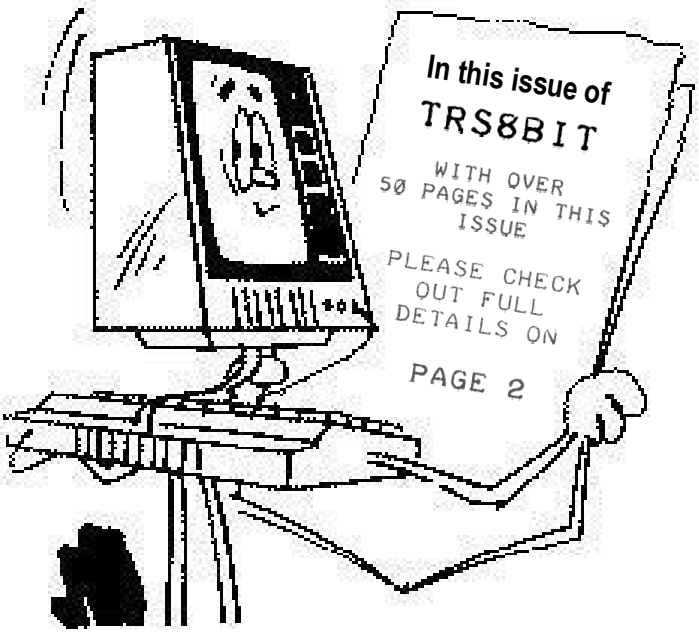
I'LL START WITH A SHORT & QUICK REVIEW OF EBAY ACTIVITY OVER THE LAST COUPLE OF MONTHS.
A VIDEO GENIE (3003) WAS SOLD FOR JUST £67.
A MODEL 4P, WHICH, ACCORDING TO THE TRS-FORUM IS BEING TLC'D, LOVINGLY, INTO THE 21ST CENTURY!
A SUPER LOOKING M2 SOLD FOR A MERE £132. IF THE LUCKY BUYER IS READING THIS, HOW'S IT RUNNING? IF YOU GET A CHANCE, PLEASE PUT A FEW WORDS ON THE FORUM WITH, IF POSSIBLE, A COUPLE OF PHOTOS.
A UK M2, COMPLETE WITH DESK AND ADDITIONAL DRIVES IS QUITE A RARITY!
AND FINALLY, JUST AS INTEREST, AN 'OSBORNE EXECUTIVE' FAILED TO GET A BID. AT £125 I WOULD HAVE LOVED IT BUT IT ALL COMES DOWN TO AVAILABLE SPACE, OR, AS IN MY CASE, LACK OF IT.

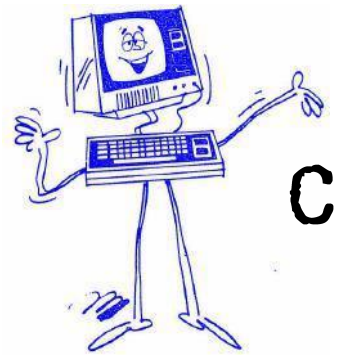
PETER CETINSKI IS STARTING A M2 'DEPOSITORY'. SO IF YOU HAVE ANY ITEMS, SOFT/HARDWARE, ADVERTS, EPHEMERA ETC., WHICH YOU MIGHT THINK HE WOULD FIND OF INTEREST PLEASE CONTACT HIM.

HAVE YOU CAUGHT PETER'S 10TH EDITION OF TRASHTALK? HE INTERVIEWS TRS-80 PROGRAMMER LANCE MICKLUS. LANCE TOLD OF HIS EXPERIENCE IN HIS ROLE IN THE HISTORY OF THE TRS-80 INCLUDING WRITING SEVERAL NOTABLE GAMES AND RUNNING A SMALL SOFTWARE COMPANY IN THE HEYDAY OF TRS-80'S.
EPISODE 11 IS JUST BEING PUBLISHED AS I PREPARE THIS EDITION OF TRS8BIT. ALSO CHECK OUT PAGE 51 FOR DETAILS OF LATEST DEVELOPMENTS.

THERE'S BEEN QUITE A BIT OF INTEREST IN CP/M SHOWN RECENTLY BOTH WITH HARDWARE DEVELOPMENTS (MAV'S MAPPER-1 AND PASCAL'S REVAMPED OMIKRON BOARD) AND ON THE FORUM.
JUST IN CASE IT'S ALL NEW TO YOU, OR YOU JUST NEED TO REFRESH THOSE 'GREY CELLS' I'VE INCLUDED A COUPLE OF

(CONTINUED ON PAGE 43)





CONTENTS

Configuring a Floppy SONY MPF 920 M1 TRS-80/3 / IV and 4P

Pascal Holdry

This floppy is the latest generation and is compatible IBM PC and SUN
It therefore operates under three fashion

1) Mode 1: 720KB (speed disk 300 TR / min) IBM PC and TRS-80
2) Mode 2: 1.2MB (disk rotational speed TR 360 / min) and SUN (TRS-80 floppy 8 ")
3) Mode 3: 1.44MB (speed disk 300 TR / min) IBM PC in HD

The strap SH1 can change the disk rotation speed via the sheet 34 strands.

Floppy SONY MPF 920

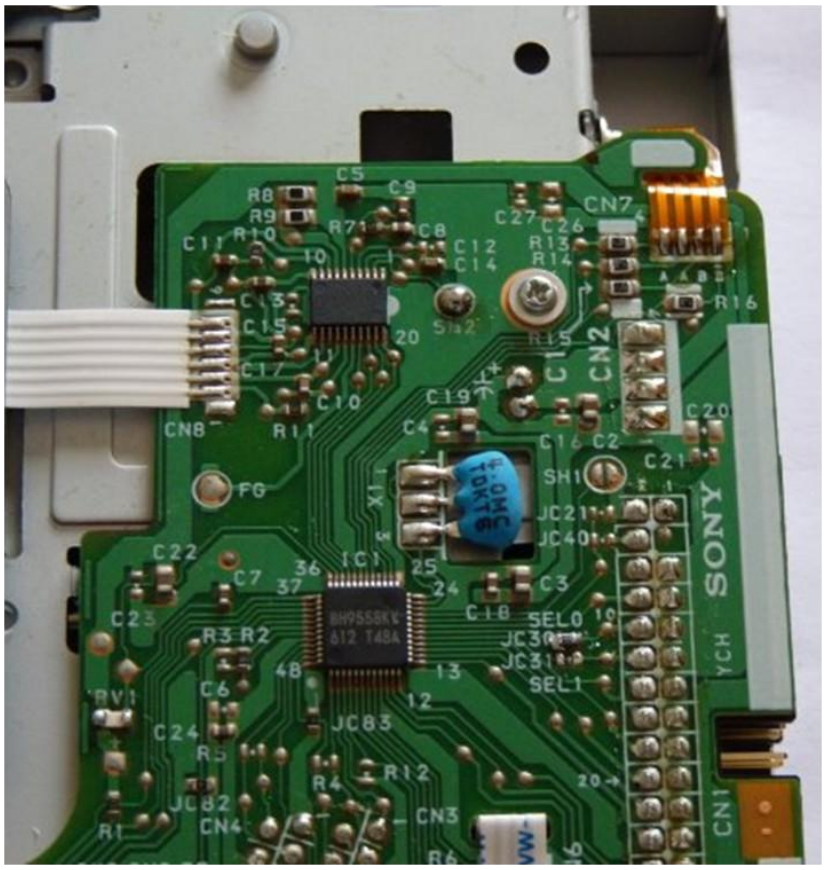
Top view

Bottom view

There are several versions of "PCB" for this floppy drive. I will describe only two

Overview of the "PCB" version 1

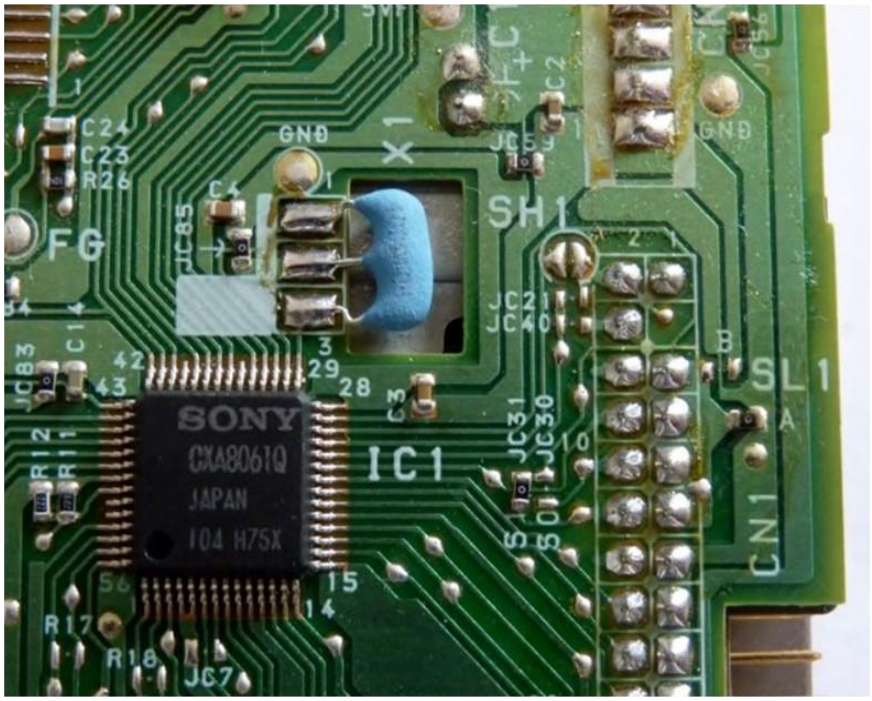
Overview of the "PCB" version 2



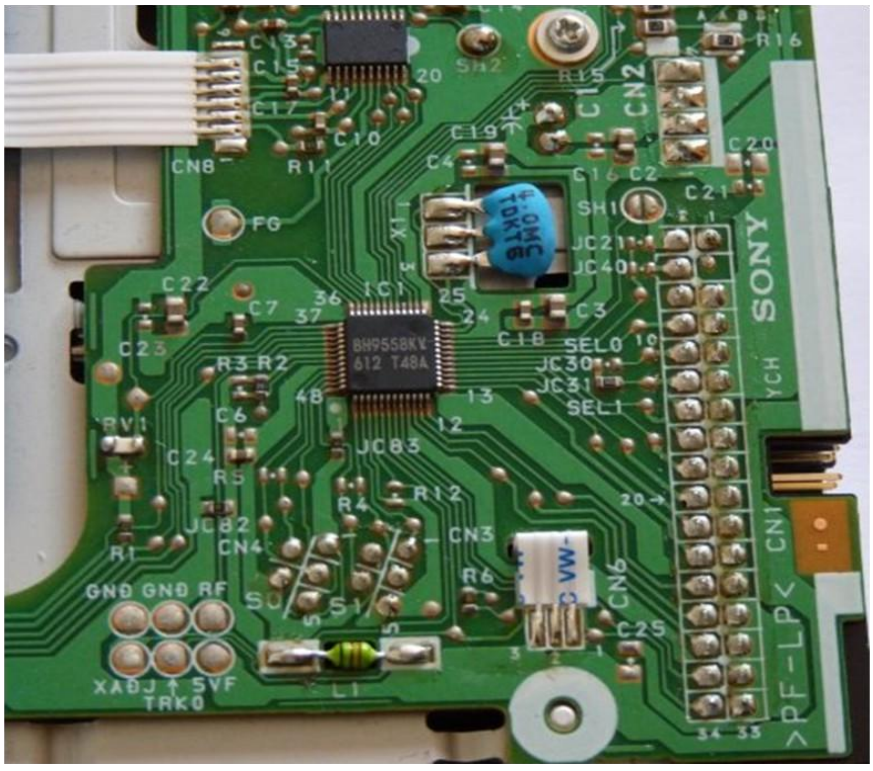
SONY 920 MPF configuration for a PC

The strap S1 or SEL1 selects the Floppy No. 1. This strap is made by a "CMS" 0 Ohm resistance

SONY 920 MPF setup for a TRS-80 floppy # 1
The strap S1 or SEL1 selects the Floppy No. 1. This strap is made by a "CMS" 0 Ohm resistance
There is no change to be made



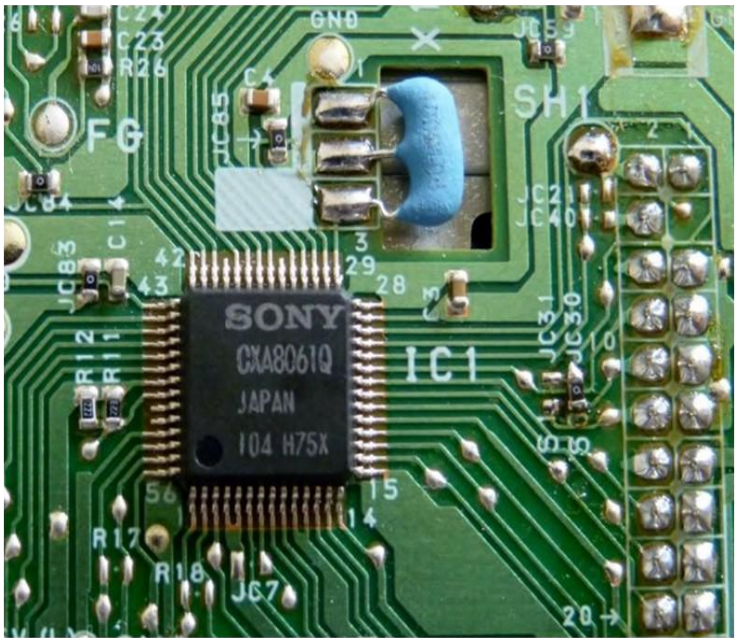
PCB v1



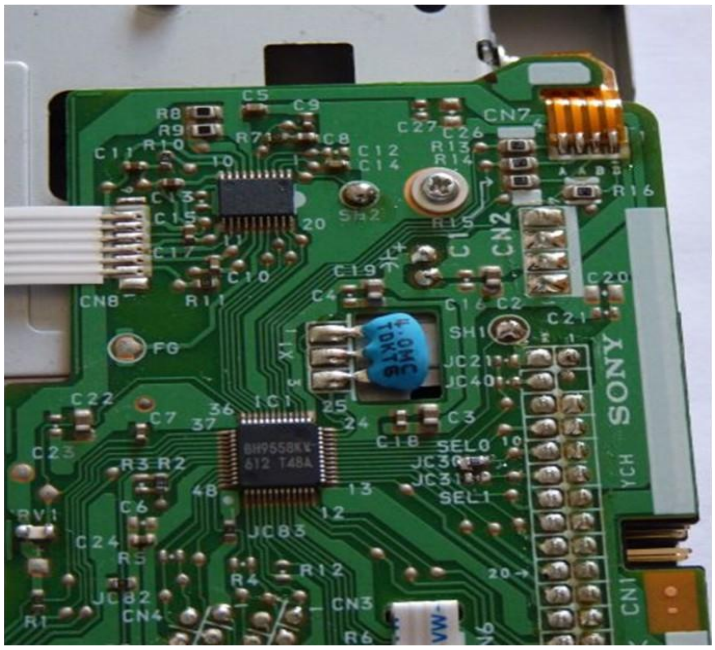
PCB v2

SONY 920 MPF setup for a TRS-80 floppy No. 0

Two changes are made.
Remove the resistance "CMS" Ohm position of S1 or SEL1
Place this resistance "CMS" in the position where S0 strap SEL0



PCB v1



PCB v2

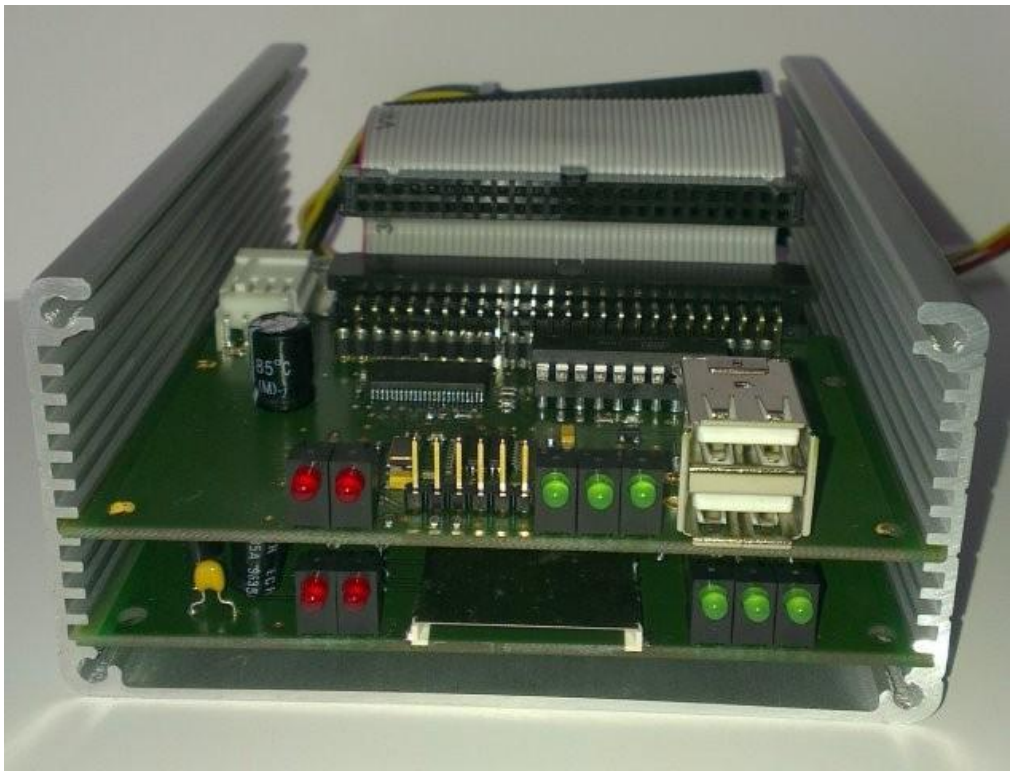
After these changes, it remains for you to make a flat cable of 34 strands with an HE10 34 pin connector to connect to floppy disk edge connector on the Expansion Interface

NOTE: It is not necessary to place a network (DIL) of 150 ohm load resistors for floppy 3.5 with a 5.25 floppy when/ if using mixed drives.

It may be necessary to add a resistor network (DIL) 150 Ohm load in the DIL support provided for this purpose on the Floppy 5.25 .

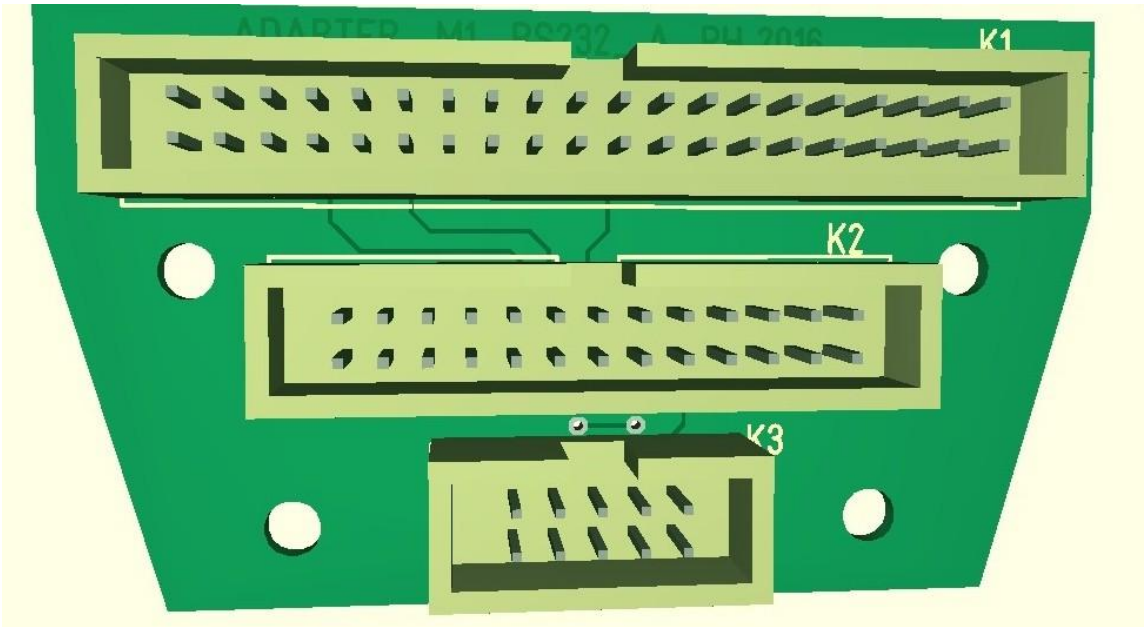
REGUSB + + FREDHD REGBT

The RegUSB and FredHD maps are redrawn to fit in a box distributed by FARNELL



COMING SOON

RS232 cable 9 / 25P TRS-80 M1



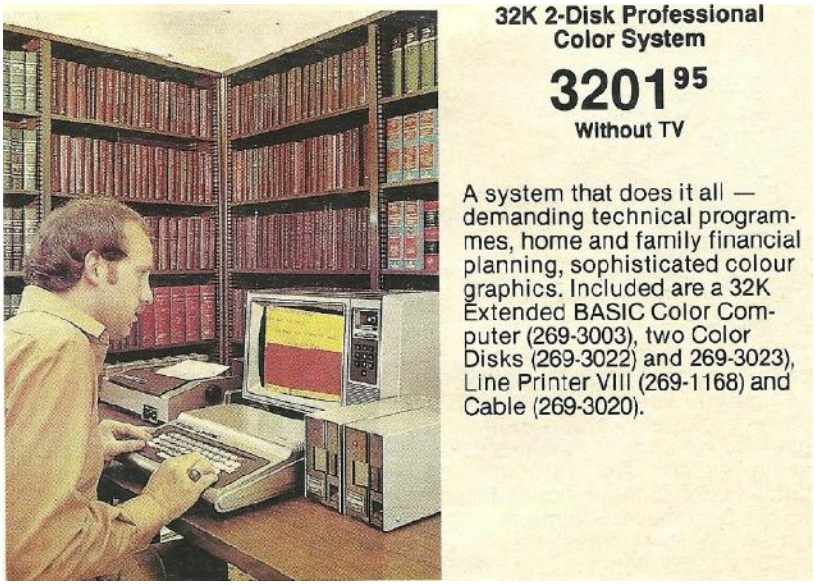
In Mav's Workshop

COLOR COMPUTER DISK DRIVES

By Ian Mavric

History: In 1980 Tandy released it's Color Computer, its small home and educational computer system designed to be connected to a colour or B&W TV set. It was priced and marketed as a competitor to similar machines of the time which include the Apple][plus and the Atari 400/800. Early Tandy advertising and press releases implied it was also a great way to get the kids off your "real" TRS-80 Model I, II or III so you could get on with work and your kids played games and learned elementary programming techniques. Techniques seen as invaluable in the new computerised future looming just around the corner as the 1980s progressed.

While the Coco, as it became known, was released with fairly low specifications (4K Color Basic (26-3001) and 16K Extended Basic (26-3002)) in order to keep the costs down, the smarts inside of the machine were quite sophisticated for the time and it was not long before talented programmers and indeed hardware hackers saw the potential of the machine and upgrades appeared. High resolution graphics was becoming a big thing, as well as a reasonable amount of memory for complex programs and floppy disk storage are all part of what was considered de-regur for a serious computer system in the early 1980s. Surprised by the interest and uptake in their Coco platform, in 1981 Tandy upgraded the memory to 32K and released a disk drive for the machine in early '82. The Color Computer as a serious alternative to the Atari 400, Apple][plus and even the Model III had arrived.



Picture: an academic uses his 32K Coco 1 for writing research papers

Disk Drives on the Coco: The first disk drive was the 26-3022 which these days we call the "grey" disk drive as its paintwork matched the computer it was designed to be used with. It was impressive looking but was not the most reliable piece of engineering to come from Radio Shack. Very few still work these days and they are best avoided because the TEC mechanisms are archaic, even by 1982 standards, only able to read 35 tracks and step at 30ms. This set the standard for all subsequent Radio Shack disk drives, as this was the lowest common denominator. The 26-3022 controller is also the most complicated and least reliable version, its a variation on the Model III disk controller and requires 12V in addition to the usual 5V and for this reason it only works on the Coco 1, or later models only with a Multi-Pak Interface. Having said that many are still in active use, and with clean edge connectors run reasonably well. Just don't go buying one for a Coco 2 or 3 and expect it to work without an MPI.

Picture: early Coco disk drives - best avoided if at all possible

The next disk drive was better, it carried the catalog number 26-3029 and this time it was painted white to match the newly released 64K Color Computer and the soon-to-be-released Coco 2. The disk drive mechanism was this time a Texas Peripherals disk drive as was in use in the Model III/4 computer, and while it was far from the last word in disk drive technology, it could step at 6ms and read up to 40 tracks. However, to maintain compatibility with the earlier drive it stepped at 30ms and still used only 35 tracks. The disk controller was re-designed at this time to use only a 5V supply so this controller worked with the Coco 2, as well as the Coco 1. This and all subsequent controllers work with all Cocos, 1,2 and 3.

FD-Series disk drives: By 1985 the Coco disk drive was looking rather dated, being a full height disk drive mounted in an upright case. The basic design of a Tandy disk drive had remained basically unchanged since they released the first disk drive for the Model I in 1978. Half-height disk drives had been perfected by 1985 and Tandy had been using them successfully in the TRS-80 Model 4P so decided to design a new disk drive unit which mounted up to two half height disk drives horizontally on top of each other, in a neat little unit called the FD-500. The Radio Shack TRS-80 name was dropped at this time, replaced with "Tandy" and the controller was unchanged except for its label it was still a 26-3029 controller. The FD-500 (catalog number 26-3129) had Tandon TM65 single-sided disk drives (one or two) installed and reliability was about equivalent to that of the 26-3029 disk drive. The main problem with the TM65 was its inability to properly latch a diskette if it was even slightly off-centre inside its jacket. It made a terrible sound and was in no way good for the diskette. The best way to avoid this problem was to have the drive spinning as you close the latch. This clamping problem seemed to get worse the older the drives got, so when they were new it wasn't too bad but once it had been in service for 2 or 3 years the clamping was a real headache.

In 1986 Tandy wised up to the shortcomings mentioned above and released the re-designed replacement, the FD-501. This was a major upgrade which included a newly-design disk controller in a much smaller case, this time with gold-plated connectors for increased reliability. The whole unit was designed by TEC in Japan and featured a very rugged TEC FB501 disk drive. No latching or other notable problems with this setup, and though the drive could easily operate at 40 tracks and step at 6ms, it still stuck with the old 35 track and 30ms stepping like all previous drives. It's worthwhile noting that by now OS/9 had been released and enterprising users found you could set the operating system up to take advantage of the newer drives extra tracks and speed, so most OS/9 users did.

Increase Data Storage

599.00 156,672 Characters of User Storage Per Diskette

FD-501 Color Thinline Disk #0. Turn any Color Computer with Extended BASIC into a complete disk system and store over 156,000 characters of data. Write your own programs or use ready-to-run software. Includes 13.3cm double-density, 35-track floppy disk drive; Program Pak cartridge containing the Disk Operating System; cable; blank diskette; manual and operator's instructions. 26-3129 ..... 599.00

Second Drive for FD-501. Adds 156K of storage. Requires installation (not included). 26-3130 ..... 519.95

Picture: advertisement for the FD-501 disk drive

Finally in late 1987 one last disk drive was released, called the FD-502 it was again made by TEC and it was aimed squarely at the Coco 3, though it did work with all previous versions of the Coco. Selling a single-sided drive in 1987-88 would be a joke so the FD-502 gets a double-sided disk drive by TEC called the FB-502 which has a usable capacity under OS/9 Level II of 360K. In native mode it's still restricted to 35 tracks/156K of storage as with all previous models. The controller was redesigned again, this time the expensive gold-plated connectors were taken away but the ROM socket was replaced with a 28-pin Intel 2764 8K compatible one, instead of the previous 24-pin Motorola 8K compatible one. This gives rise to the potential for the average hacker to modify their operating system by programming a 2764 with any cheap Eprom programmer.

Picture: advertisement for the FD-502 disk drive

Disk Drives in 2016: So you want to add a disk drive to your Coco or a Dragon 32/64 and you want to do it right and not waste money? Using the guide above you can easily see that the FD-501 and FD-502 are the ones to look for. If in good working condition the FD-500 or even the 26-3029 will do the job in most cases, leaving only the earliest 26-3022 to avoid. If you are only after a controller and have a Coco 1 or an MPI then even a 26-3022 controller in good condition should suffice. All of these controllers can support normal 360K PC compatible drives and will also work with the HxC floppy drive emulator once you work out the foibles to getting the disk images into the correct format to work on the HxC.

Dragon owners who are having trouble locating an original DragonDos controller have the option to use a Coco controller (any version will do) which need an Eprom upgrade but will work on the Dragon and read/ write both DragonDOS and SuperDOS disks. The Eprom upgrade is more complicated than simply removing an old one and installing a new one; a certain amount of re-wiring is needed so it's best left to the experts who know what they are doing. On the other hand this does lend itself to installing a switchable DOS Eprom with two DOS versions on it (one for the Coco and one for the Dragon) so you only need one disk controller and it can be used on two computers at the flick of a switch.

RE-FD502 and RE-FD502-KIT: The sad fact is that although there are reasonable numbers of controllers out there on eBay and other places for sale, most are sold AS-IS untested, and in my experience 1 in 3 either don't work or have developed some amount of flakiness. Prices on eBay vary from around $USD60.00 to over $USD100.00, which is a fair price if the device works properly, but really annoying if you have to buy two or three of them at this price to get a good one. It would be much better if someone made these new still from new components. And so with this in mind, Pascal from www.prof-80.fr assisted me with readying a reproduction of the FD-502 controller, which I now sell as a fully built controller for $USD69.00 or in a kit form for $USD40.00

New!!

RE-FD502

Coco 1-2-3-Dragon Disk Controller

(Released 12th November, 2016)

Who is it for? **It's for Coco 1, 2, 3 and Dragon owners who want to add a disk drive to their system. For people with a broken or unreliable disk controller, this will get you back up and running. Choice of three operating systems: RS-DOS 1.1 (for the Coco 1-2-3), ADOS 3 (for the Coco 3), and SuperDos E7T (for the Dragon 32/64).**

ONLY $USD69.00 + shipping

Based on the old FD-502 controller design it features the handy Intel compatible 2764 socket so people can easily upgrade their DOS as they see fit. It has so far been tested with the Coco 1, 2 and 3 as well as the Dragon 64. It has proven to be 100% compatible with the old FD-502 controller, and even fits inside the old FD-502 plastic enclosure if your unit has died you can drop a RE-FD502 straight in. The RE-FD502-KIT for true hardware hackers who like to solder their own Heathkit-style computer accessory projects together themselves saves some money over the pre-built one and comes with all the parts needed to get it completed. If you can solder competently and read a BOM then you can complete this kit. I may ask about the soldering ability of prospective buyers to make sure in my mind the builder will have the best possible chance of success in completing the kit.

New!!

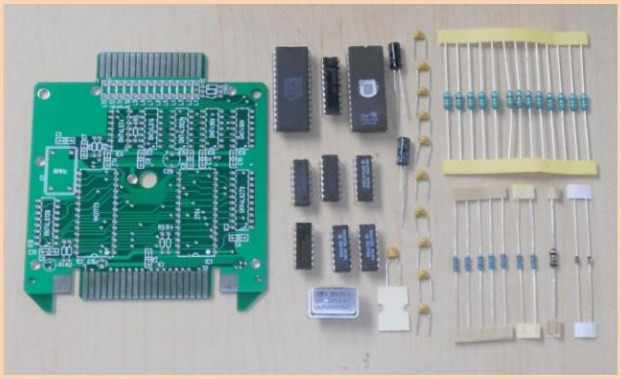
RE-FD502-KIT

Coco 1-2-3-Dragon Disk Controller KIT

(Released 12th November, 2016)

Who is it for? **It's for Coco 1, 2, 3 and Dragon owners who want to save a few dollars and enjoy soldering together their own electronic devices. It's the same as above, but you build it yourself.**

"I may ask simple questions about your ability to build this prior to selling it, but if you have made electronics projects successfully in the past then you will likely succeed with this one. You also need to tell me which DOS you prefer." - Ian.



ONLY $USD40.00 + shipping

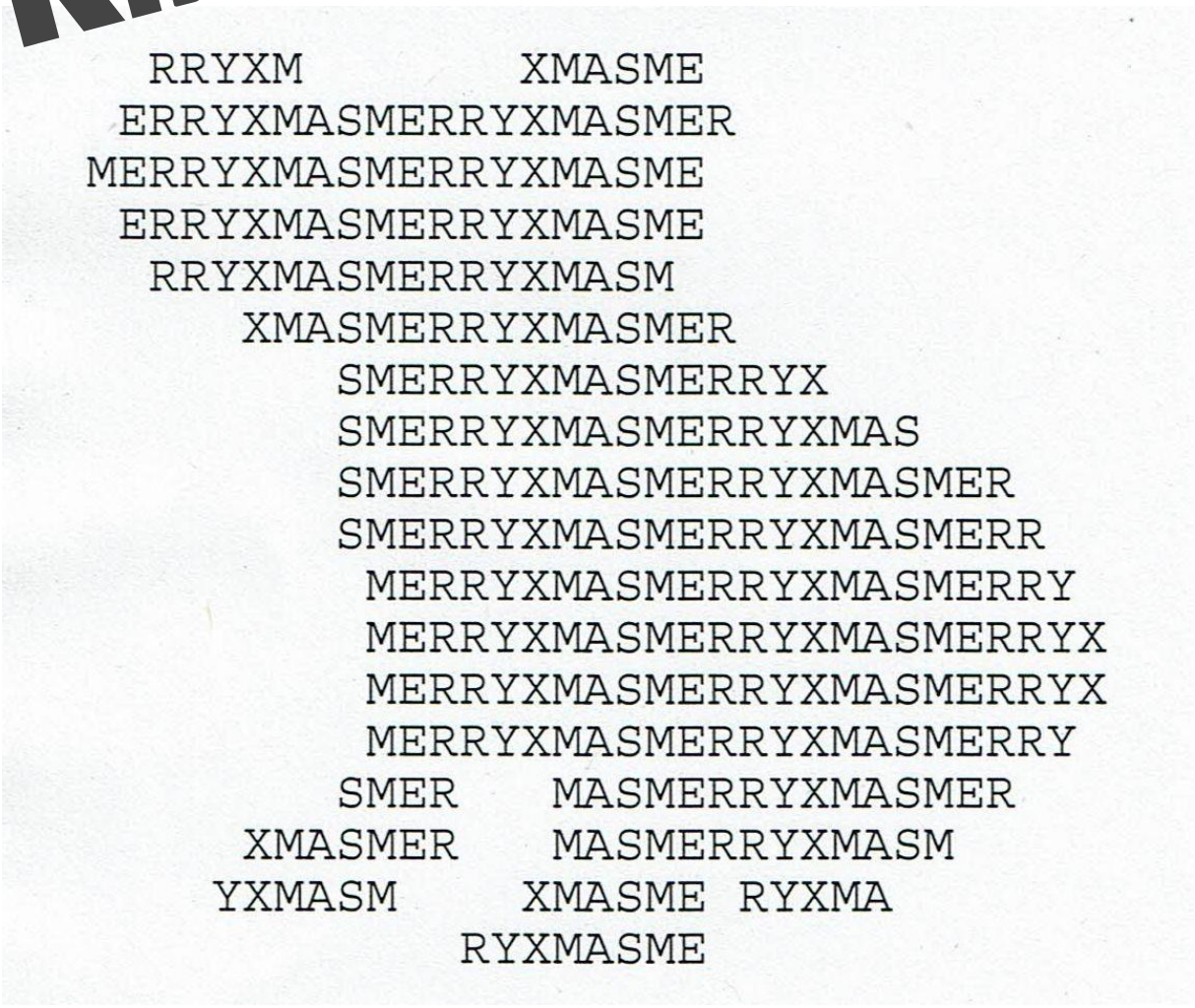
for a limited time

Ian Mavric

ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

KRISTMAS KOALA



OUR USUAL CHRISTMAS CARD FROM DEE AND MYSELF.
THIS YEAR FEATURING A KOALA BEAR!

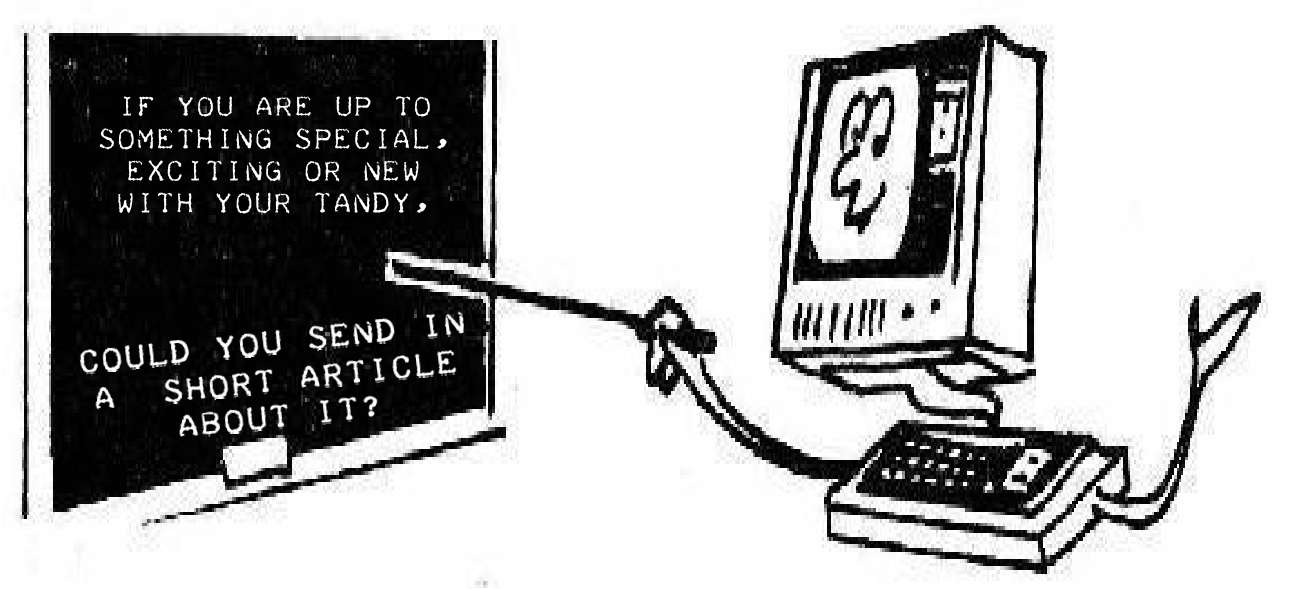
WISHING ALL VISITORS TO TRS-80.ORG.UK
TRS8BIT READER AND CONTRIBUTORS,

A VERY MERRY CHRISTMAS AND A HAPPY NEW YEAR

```
10 REM XMAS1004/BAS
20 CLEAR500
30 M$="MERRYXMASMERRYXMASMERRYXMASMERRYXMASMERRYXMASMERRYXMAS"
40 GOSUB 220
50 READ S
60 IF S=0 GOSUB 160
70 READ L
80 T$=MID$(M$,S,L)
90      X=PEEK(VARPTR(P$)+1)+PEEK(VARPTR(P$)+2)*256

100 REM THIS AND THE NEXT LINE ARE NOT NEEDED ON 16K SYSTEMS
110      IF X>32767 THEN X=X-65536

120          FOR Y=1TOL
130          POKE X+S+Y,ASC(MID$(T$,Y,1))
140          NEXT Y
150 GOTO 50
160 LPRINT P$
170 GOSUB220
180 READ S
190 IFS=0THENLPRINTP$:READS
200 IF S=99 THEN END
210 RETURN
220 P$=STRING$(80,32)
230 RETURN
240 DATA 3,5,15,6,0,2,20,0,1,20,0,2,19,0,3,17,0,6,16,0
250 DATA 9,16,0,9,19,0,9,22,0,9,23,0,10,23,0,10,24,0
260 DATA 10,24,0,10,23,0,9,4,16,15,0,6,7,16,13,0
270 DATA 5,6,15,6,22,5,0,13,8,0
280 DATA 99
```



ONELINER COMPETITION RESULTS

MANY THANKS TO E.T., MAV PETER AND GEORGE FOR THEIR COMPETITION ENTRIES.

THERE WAS A BIT OF A DISPUTE AT MILLER TOWERS THE OTHER SUNDAY AFTERNOON WHEN DEE HAD HER JUDGES HAT ON. MY 7 YEAR OLD GRANDDAUGHTER THOUGH THAT E.T.'S MUSIKEY SHOULD WIN, BUT AS THE PROGRAM WAS MORE THAN ONE LINE, IT RATHER DEFEATED THE IDEA OF A ONE-LINER PROGRAM :). HOWEVER, I FELT THAT IT WAS JUST SO GOOD AND MORE THAN WORTHY ENOUGH TO FEATURE IT IN THE NEWSLETTER. (AFTER ALL, ACCORDING TO THE TV, 7 YEAR OLDS ARE IN CHARGE AND KNOW EVERYTHING)
WELL DONE E.T., I SHALL SEND ALL THE EMAILS OF COMPLAINTS AND MOANS ABOUT THE NOISE ON XMAS AFTERNOON, DIRECTLY TO YOU :)

MAV'S ENTRY, A VERY CLEVER AND NEAT LITTLE PROGRAM MAKING FULL USE OF THE M4 '\' INTEGER DIVISION AND 'MOD' OPERATORS (WHICH ARE NOT AVAILABLE ON THE M1 OR M3) CALCULATES THE DAY OF THE WEEK FROM A GIVEN DATE.

GEORGE'S ENTRY, DRAWS A GRAPHIC REPRESENTATION OF A PLANET AND MOON WHICH PUSHES A STANDARD M1 OR M3 L2 DISPLAY TO THE LIMITS.

PETER'S ENTRY WAS, I FOUND, QUITE FASCINATING. COLLATZ CONJECTURE IS, ON THE FACE OF IT, A SIMPLE PROBLEM, BUT AS WITH MOST THINGS MATHEMATICAL, IT AINT!. THERE'S LOADS OF INTERESTING FACTS & FIGURES BOTH ON 'WIKI' AND 'NUMBERFILE'. WHEN YOU RUN THE PROGRAM, TRY THE NUMBER 27, IT'S A GOOD ONE :)

AFTER CAREFUL CONSIDERATION, DEE MADE HER DECISION AND I AM HAPPY TO OFFICIALLY REPORT THAT GEORGE PHILLIPS, WITH HIS "PLANET & MOON" ONE-LINER, IS THE WINNER.
I LOVE THE WAY HE TAKES ADVANTAGE OF GOTO WHEN USING LINE 0.
IT MAKES A GREAT SCREEN-SAVER TOO!

MAV'S PROGRAM WAS JUST SO GOOD, IT INSPIRED ME TO GET A M1 L2 VERSION UP AND RUNNING, (JUST FOR ALL US M1 FANS).

BELOW IS A M1 L2 VERSION WHICH IS BASED ON LEWIS ROSENFELDER'S BOOK, "BASIC FASTER AND BETTER". THE VERSION IN THE BOOK IS FOR DISK BASIC, BUT IT CAN WORK JUST FINE IN L2 BASIC WITH SIMPLE AMENDMENTS, TO AVOID USING THE DEFFN VERB. IT ONLY WORKS BETWEEN THE YEAR 1901 - 2099, BUT I THINK THAT WOULD SEE ME OUT QUITE NICELY!

```
1 INPUT"Y,M,D";Y,M,D:X=Y*365+INT((Y-1)/4)+(M-1)*28+VAL(MID$("000
303060811131619212426",(M-1)*2+1,2))-((M)2)AND((YANDNOT-4)=0))+D
:PRINTMID$("FRISATSUNMONTUEWEDTHU",(X-INT(X/7)*7)*3+1,3)
READY
>RUN
Y,M,D? 2017,1,1
SUN
```

MUSIKEY

E.T. FONEHUME

HI DUSTY -
HERE IS MY ENTRY TO THIS YEARS ONE-LINER COMPETITION. IT'S JUST SLIGHTLY OVER (BY ONE LINE!) OF A ONE-LINER BUT I HOPE, AT LEAST, YOU WILL STILL ENJOY IT !!

TURN NUMBERS 1 TO 8, ON YOUR NUMERIC KEYPAD INTO A PIANO. (WELL NEARLY EH?). THERE IS EVEN A COUPLE OF XMAS SONGS PROVIDED TO SING ALONG WITH, SHOULD YOU CATCH THE CHRISTMAS SPIRIT. (ME, MINE COMES FROM A BOTTLE IF SANTA REMEMBERS!)

THANKS FOR ANOTHER SMASHING YEAR WITH TRS8BIT, I'M LOOKING FORWARD TO ENJOYING THE TANDY SCENE IN 2017.

```
1 DATA 205,127,10,76,69,62,1,211,255,16,254,69,62,16,211,255,16,
254,13,32,239,201,142,128,114,106,96,84,74,70
2 A$="                              ":A=VARPTR(A$):B=PEEK(A+1)+PEEK(A+2)
*256:DEFUSR=B:FORC=BTOB+21:D=C:READE:POKED,E:NEXT:FORZ=1TO8:READ
T:T(Z)=T:NEXT:FORY=0TO1STEP0:N$=INKEY$:IFN$=""THENNEXTELSEV=VAL(
N$):Z=USR(T(V)):NEXT
```

JINGLE BELLS -
333-333-35123 44444333 2233-5

RUDOLPH THE RED NOSED REINDEER -
5653865 - 5655587 - 4542765

AND JUST TO DRIVE EVERYONE MAD, CHRISTMAS AFTERNOON
TWINKLE, TWINKLE LITTLE STAR -
1155665 - 4433221 ETC ETC

DAYS OF THE WEEK

IAN MAVRIC

Hi Dusty, here is my competition entry, it's a one line program which computes what day it is given the date.

In the 1st example, today's date 28th November, 2016, returns Monday, which it is.
In the 2nd example 1st Jan 2017, new years day, falls on a Sunday.
In the last example the date 3rd August, 6977 shows us the date the Model I turns 5000 years old is on a Sunday.
For those interested it's written in Model 4 Basic.

Cheers,
Ian.

```
Ready
list
10 INPUT"D,M,Y";D,M,Y:E=M+10:F=Y+(M-14)\12:P=((13*(E-(E\13)*12)-1)\5+D+77+5*(F-(
F\100)*100)\4+F\400-(F\100)*2)MOD 7:PRINT MID$("SunMonTueWedThrFriSat",P*3+1,3)
Ready
run
D,M,Y? 28,11,2016
Mon
Ready
run
D,M,Y? 1,1,2017
Sun
Ready
run
D,M,Y? 3,8,6977
Sun
```

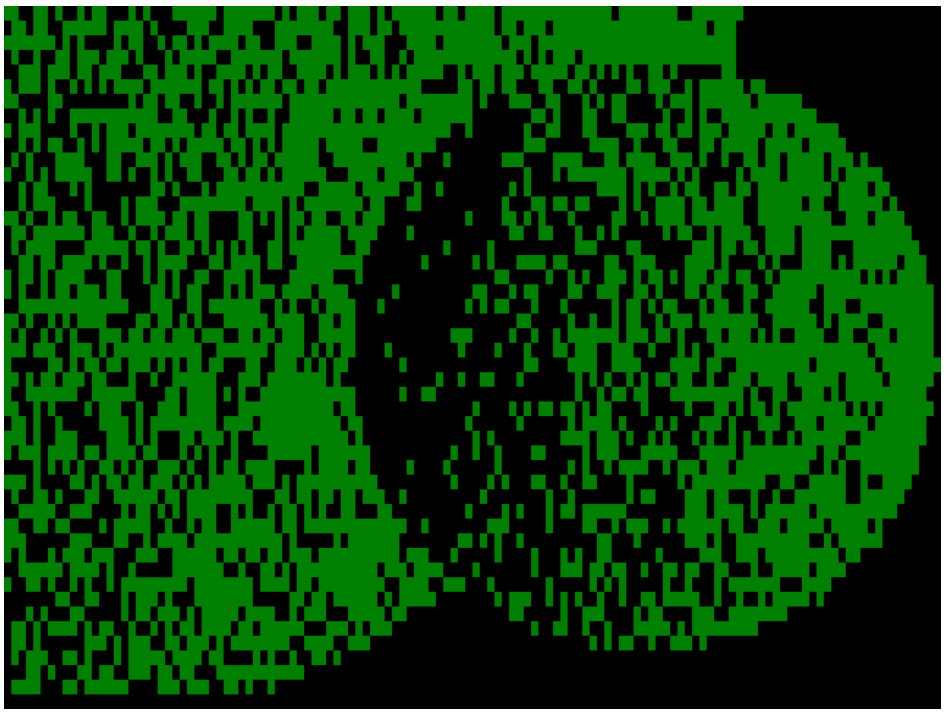
PLANET & MOON

WINNING ENTRY - GEORGE PHILLIPS - WINNING ENTRY

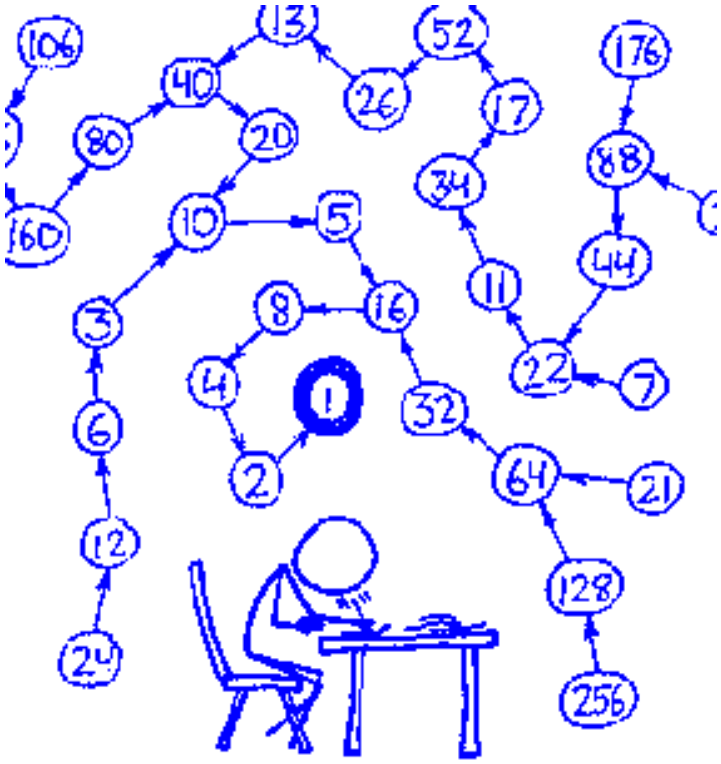
IT TAKES A WHILE TO RUN BUT ENDS UP DRAWING TWO SHADED SPHERES THAT WE MIGHT IMAGINE ARE A PLANET AND ITS MOON.

FOR A TRS-80 MODEL I OR MODEL III WITH LEVEL 2 BASIC. THE PROGRAM IS NEAR THE LIMIT SO YOU NEED TO USE EDIT TO MAKE IT FIT. IF YOU TYPE IT IN, DON'T BE DISTURBED BY THE VARIOUS APPEARANCES OF GOTO WITHOUT A LINE NUMBER. IT SO HAPPENS THAT BASIC HAPPILY ACCEPTS THAT AS EQUIVALENT TO "GOTO 0".

GEORGE



```
0 IFZ=0CLS:Z=1:GOTOELSEREADR,Q,U,V:FORI=0TO1STEPR:NEXT:FORX=QTOR
:A=SQR(R*R-X*X)/2:C=V-A:D=V+A:FORY=CTOD:J=X+U:K=Y:IFJ<0ORK<0ORJ>
127ORK>47 NEXT:NEXT:GOTOELSEIFRND(2*R)<R+X SET(J,K):NEXT:NEXT:GO
TOELSERESET(J,K):NEXT:NEXT:GOTO:DATA100,,,,40,-40,87,24,,,,
```



COLLATZ
SLOTS

PETER PHILLIPS

```
1 DEFDBLN,X:IFN=0THENPRINT"COLLATZ SLOTS, ENTER INTEGER";:INPUTX
:N=INT(ABS(X)):GOTO1ELSES=S+1:PRINTN:IFN=1PRINT"YOUR SCORE IS";I
NT(S/LOG(ABS(X)+1)):RUN:ELSEIFINT(N/2)=N/2THENN=N/2:GOTO1ELSEN=3
*N+1:GOTO1
>RUN
COLLATZ SLOTS, ENTER INTEGER? 3
 3
 10
 5
 16
 8
 4
 2
 1
YOUR SCORE IS 5
COLLATZ SLOTS, ENTER INTEGER?
```

IN MAV'S WORKSHOP:

"SOLE" EXPLAINED Part 2 (New_PERC Doubler)

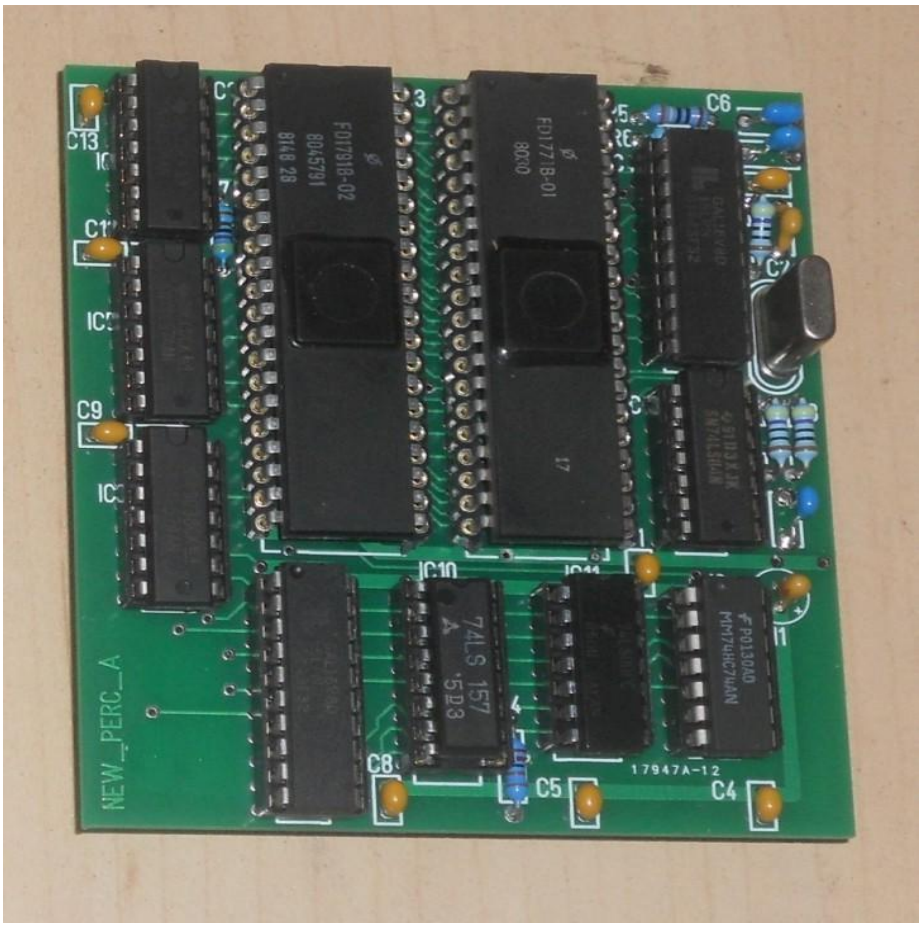
- BACKUP LDOS 5.3.1 DOUBLE DENSITY ON THE MODEL 1 -

by Ian Mavric

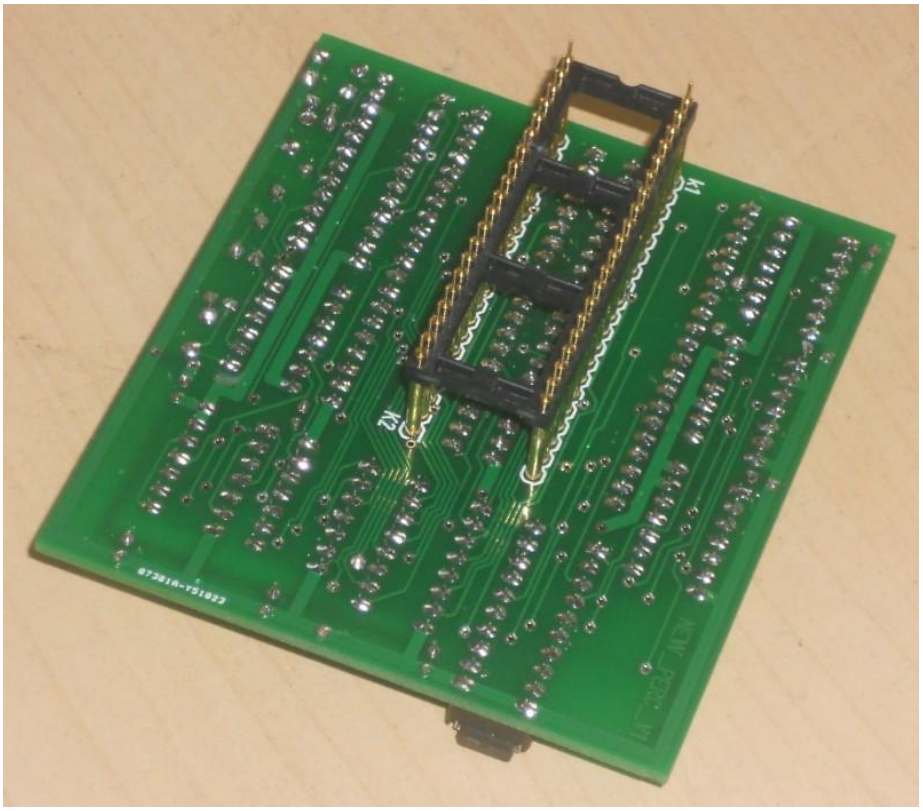
Last Time: A few issues ago I wrote about the history of double density adapters for the Model I and how LDOS utilised the device to get around 156K storage on a 35-track diskette, 184K on a 40-track single -sided diskette, 360K on a 40-track double sided diskette and 720K on an 80-track double-sided diskette. Using double density drivers PDUBL/CMD or RDUBL/CMD in LDOS 5.1.4 or FDUBL in LDOS 5.3.1 you can format and work with double density diskettes easily and as an added bonus the file structure is compatible with LDOS 5.x on the Model III and LS-DOS 6.x on the Model 4, making data transportability really easy.

Since the Model I can't boot a diskette with a double density track zero, Misosys released a program called SOLE which reformatted track zero in single density with some additional instructions to kick the system in double density operation, thus giving users a double density bootable diskette in drive 0. I showed step-by-step how to set up a double-density booting disk for your own systems.

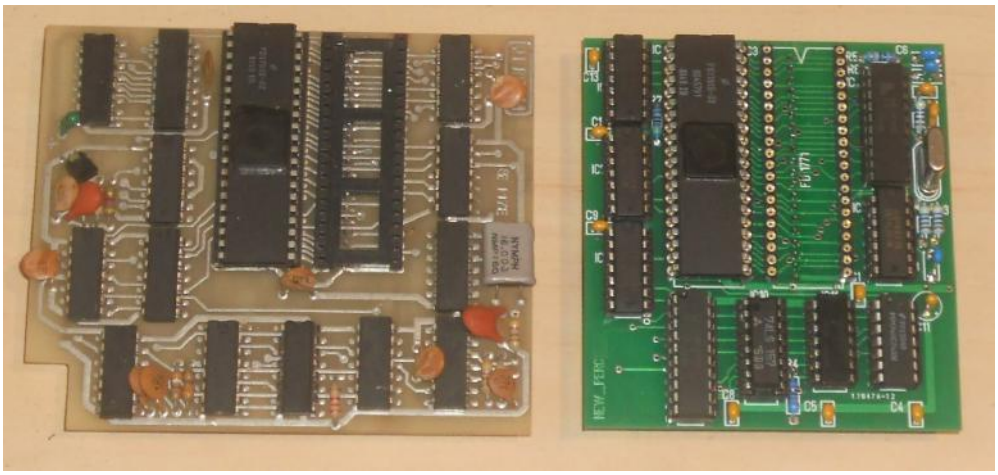
New_Perc: Since that last article I've started production of the New_Perc double density adapter for the Model I. It's the first new double density adapter for the TRS-80 since the late 1980s, and was a joint effort between Prof-80 and Régis Munoz, which takes the design of the old Percom Doubler II and reduces the parts count from 14 chips to 10 while maintaining compatibility with the old Percom. It's built up from all new parts and is not only compatible with LDOS, but also double density versions of other popular operating systems such as Newdos/80 v2.00, Dosplus 3.5 and Multidos 2.01



Picture: New_Perc with the WD1771 out of the EI installed on it



Picture: New_Perc has a gold-plated connector for reliability



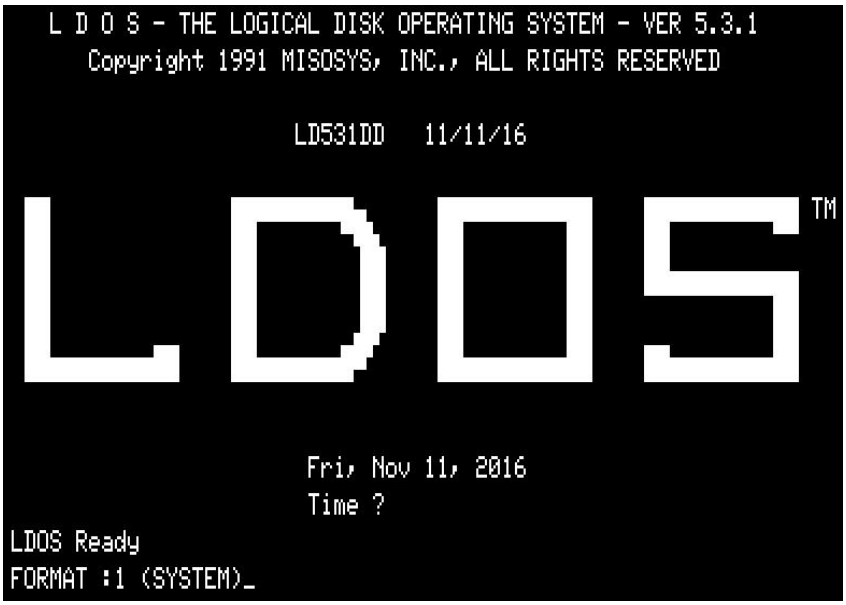
Picture: Percom Doubler II (left) sitting next to a New_Perc doubler

HOW TO BACKUP A DOUBLE DENSITY BOOTING DISK :

Previously: In the single density world of LDOS 5.3.1 to backup a diskette you would use the QFB utility. QFB stands for Quick Format and Backup, and simply reads a single density track from the source disk, formats a track on the destination disk, dumps the track and verifies it, and moves onto the next one. It starts at track zero and finishes with however many tracks it finds on the source diskette. QFB has one limitation which makes it unusable for backing up your mixed density disk is that QFB can only work with one density or the other, but can't handle a disk where track zero is single density and the rest of them are double density.

Two Step Process: There is a way to backup the diskette using the Format and Backup commands in the traditional manner. The only thing which is important here is that the destination disk is already formatted in the correct mixed density format with track zero single density and the rest in double density.

1: Format the destination disk with FORMAT :1 (SYSTEM)



It's important at this step that the cylinder count of the destination disk is the same as that of the source disk, which will normally be 40 cylinders, however some people use 35 cylinders and others push their drives to 41 or 42 cylinders if their drives are capable. For compatibilities' sake, I use 40 cylinders:

```
FORMAT - LDOS Disk Formatter Program - Version 5.3.1
Copyright 1991 MISOSYS, Inc., All rights reserved

Diskette name ? ANYTHING
Master password ?
Single or Double density <S,D> ? D
Enter number of sides <1,2> ? 1
Number of cylinders ? 40
Boot strap stepping rate <6, 12, 20, 30/40 msecs> ? 40
Disk contains data -- Name=LD531DD   Date=11/11/11
Are you sure you want to format it ? Y
Formatting cylinder 14
```

Once completed you can see that it has re-formatted track zero in single density:

```
Disk contains data -- Name=LD531DD   Date=11/11/11
Are you sure you want to format it ? Y
Formatting cylinder 39
Verifying  cylinder 39
 Re-format track 0...
Formatting cylinder 00
Verifying  cylinder 00
Directory will be placed on cylinder 20

Initializing SYSTEM information ....++.................
Formatting complete

Note: Realtime clock no longer accurate

LDOS Ready
_
```

2. Use the BACKUP command to copy the disk track-by-track to the destination disk. LDOS users habitually use the QFB command to backup disks and the BACKUP command to copy file-by-file because that is what both utilities are best designed for, however in this instance type BACKUP :0 :1 by itself, without qualifiers:

```
LDOS Ready
BACKUP :0 :1

BACKUP - LDOS Disk Backup Utility - Version 5.3.1
Copyright 1991 MISOSYS, Inc., All rights reserved

Different pack IDs! Abort backup ? N
Verifying cylinder  23

Backup complete

Note: Real time clock no longer accurate

LDOS Ready
-
```

It may say there is a Pack ID error and this is normal if it does, just reply N which will make it continue the backup, which this time will be track-by-track, including the single density track zero. Once completed your destination disk will boot and run without problems.

Ian Mavric

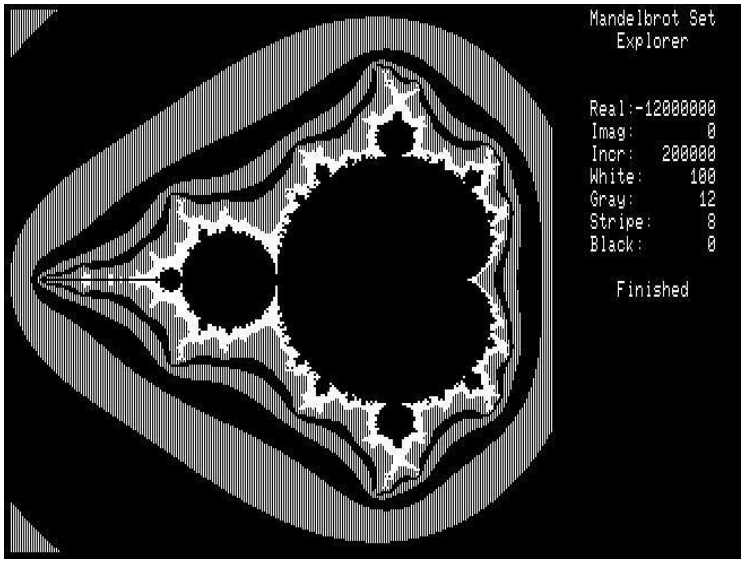
ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

LOOKING FOR HIGH-RESOLUTION GRAPHICS FOR YOUR TRS-80 MODEL 4/4D?

Then you simply must have the

"Improved Grafyx Resolution Solution!"



-Improved version of the established Grafyx Solution GX 4.2 board
-640x240 resolution for a total of 153,600 pixels, High Bandwidth
-Easy install in minutes inside your Model 4, Gate-Array Model 4, or 4D
-4P version is just around the corner - stay tuned!
-Compatible with most M3 and M4 Hi-Res software for Radio Shack HRG
-Designed in Belgium and proudly built and shipped from Australia or UK

Order yours today exclusively from:

UK and EU: BetaGamma Computing: bgialopsos@aol.com

Rest of the world: http://ianmav.customer.netspace.net.au/hires/hires.htm

NEW REDUCED PRICE USD$165
(DUE TO AU$ EXCHANGE RATE CHANGES)
WITH ADDITIONAL DISCOUNT FOR FURTHER PURCHASES.
CHECK WITH IAN FOR FULL DETAILS

First impressions of CP/M and the Omikron Mapper

I ordered the package described as "The Works" direct from Omikron in California, in response to an advertisement in "80-microcomputing", and some six weeks later it arrived, after I had paid about £50 in VAT and duty. The box contained a PCB, three diskettes (WORDSTAR is to follow) and no less than ten manuals, as follows:

The Omikron Mapper Owner's Manual
An Introduction to CP/M Features and Facilities
CP/M 2 User's Guide
CP/M 2 Interface Guide
CP/M 2 Alteration Guide
CP/M Assembler User's Guide
CP/M Editor (ED) User's Manual
CP/M Dynamic Debugging Tool (DDT) User's Guide
Microsoft Basic-80 (MBASIC)
CBASIC Language Manual

Installation instructions were clear and adequate. The Z-80 chip is removed from the keyboard and inserted in the Mapper board, which is then replaced in the Z-80 socket. Some modifications are suggested, and were necessary in my case, to achieve satisfactory operation. These are to improve the consistency of power supply to the Z-80 and the buffer chip Z75, and consist of replacing C47 and C56 by 4.7uf tantalum capacitors (supplied by Omikron) and adding a jumper lead from pin 11 of the Z-80 socket to pin 16 of Z75 and another from pin 29 of the Z-80 socket to pin 8 of Z75. Pressing RESET then gives you a choice between CP/M and TRS-80 modes; in the latter case the system behaves exactly as it did before.

In CP/M mode all address signals from the Z-80 are modified to be 4K lower, so that in a 48K system you have RAM from 0 to BFFF hex. The screen and other memory-mapped devices have C000 hex added to their addresses, and C000 to C7FF is occupied by a ROM containing part of the CP/M code. This means that data can be loaded in TRS-80 mode and then accessed (but at a different address) in CP/M mode, or vice versa.

CP/M comes adapted for the TRS-80 and Mapper, and with an OPTIONS program to tailor it for your particular configuration (type and number of drives, lower case,

printer, etc.). The resulting CP/M is sufficiently standard for all the standard manuals to apply, subject to some special uses of certain keys. There is sufficient information in the manuals to find your way round CP/M and make any changes which you want, though none should be necessary.

All CP/M systems come with the Editor (ED), an 8080 Assembler (ASM), the debugging aid DDT and the data transfer utility PIP. With this version a number of extra programs are provided: a fast copy program, memory and disk test programs and the options program mentioned above. In addition, a program is provided to transfer ASCII text files from TRS-80 disk format to CP/M.

The documentation is generally adequate, though there is the usual problem that the versions of manuals and disks are not quite in step. CP/M itself is not difficult to use, though of course not up to the standard of say Newdos/80; however the editor is somewhat primitive, designed for a teletype user, and the Assembler uses the old Intel mnemonics and does not support the additional Z-80 instructions. I expect WORDSTAR will solve the first problem, and plan to use ZEN in TRS-80 mode to solve the second, passing the object code across in memory. Alternatively, when funds permit there are of course many suitable products available for CP/M.

MBASIC-80 is similar to TRS-80 Disk BASIC, while CBASIC is widely used for commercial applications; I have verified that both work, but have no further comments on them as yet.

To sum up, this package is so far at least living up to my expectations, with the exception of the late delivery of WORDSTAR. After an hour or so of soldering it worked perfectly, and the documentation is certainly plentiful and generally quite readable. Omikron seem to have done a good job, and I look forward to developing software on my "new" CP/M computer.

Tim Bourne

MODEL 4

NOTES ON CP/M 3.0 PLUS by Roy Barber.

CP/M 3.0 Plus from Tandy comes on a set of five, 5 1/4inch disks. It is supplied boxed in a nice container in which the two, 3 ring binders containing the documentation fit. The five disks are in a plastic wallet inside one of the binders.

The main CP/M system is on one of two disks. Two versions of the system disk are provided as a different operating system is necessary for banked or unbanked memory. The unbanked disk is for 64k systems and the banked version is for 128k systems.

The remaining disks hold various utilities not normally necessary on the system disk.
A list of the contents of each disk is provided at the end of this article.

This CP/M is the standard version which has been adapted for the Model 4 by Digital Research. This is apparent from the documentation which is all clearly marked as supplied by D.R. The 3.0 Plus modifications for Model 4 appear to be mainly contained in one file called SETUP. This allows various temporary or permanent modifications to be made to the CP/M system to suit the user. A full list of all the options available is too long to list in detail but a summary is as follows:
Optional line wrap on the screen, block or underline cursor, clear screen on form feed and blinking or steady cursor (which does not work !!).

The disk choices are: Number of drives, step rate (not active, seems set at 6mSecs), number of hard drives up to 4, error messages, and number of error retries.

Printer options are: supress line feeds on carriage return, auto form feed, set lines/page, and set number of lines before form feed.

Keyboard options are: type ahead buffer on/off and program special function keys, (quite useful as any user choice can now be put on these keys).

Serial port options are: select Baud rate, set line protocol for xon/xoff, select CTS/RTS etc. Also word length and stop bits can be set up.

Auto boot options allow auto execution of a warm boot file. A better choice would have been a cold boot auto execution file !. A 50/60 Hertz line frequency choice can be made which allows the R.T. clock to maintain time correctly.

A file called COPYDISK allows formating and backing up disk operations. This does not work correctly. One of the options is

to allow system tracks to be copied. (The equivalent of SYSGEN in Vers.2.2). It does not put all the system on the destination disk as it should and the disk will not boot !. I have found this can be overcome by ignoring the instructions in the general user handbook provided. This says that the program COPYSYS is only for 8inch disks. If COPYSYS is used for 5inch disks it does put the system tracks on correctly. It will ask if CPM3.COM is to be installed. The user should answer Y for yes. This will copy this system file over to the sysgen'd disk. The job of sysgen is not yet complete however as although the disk will now boot it will look for another file called CCP.COM. This is also a system file and can be copied over to your sysgen'd disk using PIP with the (R) option which allows the copying of a system file. This is all a lot of unnecessary additional work to have to do when the COPYDISK program should do it in one operation. I understand Tandy is looking in to this problem. After sysgening a disk one has about 128k left to use.

One of my most serious criticisms of this version of CP/M is that there is no option for double sided drives. To my mind this is a major ommision. Also only four floppy drives are allowed so one cannot even reconfigure three double sided drives as six single sided drives using a hardware modification.

As CP/M users will/should know, it is necessary to install most programs to configure them to work from the particular terminal your CP/M supplier has emulated in his version of CP/M. Most CP/M suppliers make the computer emulate the popular ADM3A terminal and when installing a program (like Wordstar) all that is necessary is to select that terminal from the list the program provides and your software is usable on your computer. Unfortunately D.R. (or was it Tandy ?) chose to emulate the DEC VT52 terminal which does not appear on any of the lists of terminals supplied for installation with most CP/M programs. This means the newcomer to CP/M will experience considerable difficulty in running any program needing installation under this CP/M. To further add to the problem the list of terminal display functions provided in the manual is hopelessly incorrect and any attempt to patch the display functions in to a program is likely to cause the user to write off this version of CP/M in frustration.
A list of the correct codes (as far as I can establish) is given below. Only those incorrectly shown in the manual are given here (some were right !) .

Cursor right	27,67 decimal
Reverse tab	27,73
Home cursor	27,72
Restore cursor to saved position	27,107
Erase to end of page	27,74
Erase current line	27,76
Erase to end of line	27,75
Delete line	27,77
Delete character	27,78
E O L wrap-round	27,79
Discard char at E O L	27,68

Some of the hex codes given are also incorrect but if you use

the decimal codes shown you can work these out. You should also be warned that codes 27,110 and 27,111 for erase from start of screen to cursor and read cursor position hang the system and need a cold reboot to regain control.
I have been able to install most CP/M programs using the above codes.
You also need to know that the cursor lead-in characters required by most install programs are <esc>Y or 27,131.

Some of the key codes given for generating special characters are also not clear from the documentation, those most likely to cause trouble are: left and right brace or wiggly brackets. These are obtained by pressing left or right shift keys with the @ key. Clear screen is obtained by pressing clear and caps keys together followed by <enter>.

Regarding the performance of the CP/M this appears OK so far but the failure of the SETUP program to provide the steady cursor presents quite a problem. When using a program like Wordstar the cursor goes out while it is moved so if you want to zip along a line of text you can not tell where the cursor is. I trust Tandy will put this right soon.

The documentation other than that referred to is good and quite detailed. The provision of the MAC assembler with full documentation and the various library files with the link program is probably worth the cost of the whole CP/M. The assembler, although basically for 8080 has macros which allow its use as a Z80 linking assembler.

Hard disk users are well catered for with the ability to drive up to four such devices.

Although the banked memory version of CP/M should provide up to 64k of additional memory, because a lot of the CP/M operating system has to reside in each bank in parallel only about 11k extra is available to the user. This is according to the documentation which says there is 44k available in unbanked and 55k available in the banked version. From this I would think the Montezuma Micro CP/M vers 2.2 gives better value in terms of memory usage. It has to be used as a Ramdisk but 61k is available to the user in this form on top of the normal 48k in the TPA. Also in comparison with Vers 3.0 Plus the M.M. 2.2 is easier to use and very much easier to install to most CP/M programs such as Wordstar as it emulates the much more popular ADM3A terminal. The M.M. Vers 2.2 also allows the use of double sided drives and can read many other useful manufacturers formats.
Perhaps for once Tandy tried too hard to keep the price down and the product has suffered for this reason. The cost of CP/M 3 Plus is £79.95 against the M.M.Vers 2.2 available from Mollmerx at £171.00

If you would like to read more about the Version 3 Plus advantages over the usual 2.2 version get the book 'CP/M The Software Bus'. by Clarke, Eaton and Powys-Lybbe. Its published by Sigma Press and is as good a book on the subject as any. It

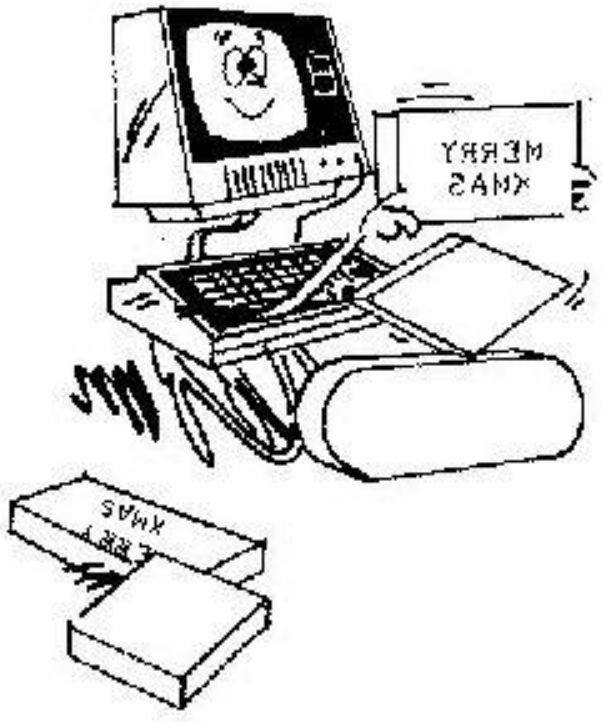
gives more information on Vers. 3.0 than most.

P.S. I have been using the Borland CP/M Pascal for a while now. At about £50 its quite first class. More on this later.

PPS. No one has yet given me any real help on how to access the Model 4 screen or keyboard matrix. Whats happened to all the Hackers ?..
Surely The Model 4 is not too much for them?. If they can find out how to copy SuperUtility Plus surely what I want is chicken feed....

CP/M 3.0 Plus disks:
Disk 1
CCP.COM COPYDISK.COM DATE.COM ERASE.COM PIP.COM
RENAME.COM SETDEF.COM SUBMIT.COM COPYDISK.COM CPM3.COM
DIR.COM GET.COM PUT.COM SET.COM SHOW.COM TYPE.COM
Disk: 2
,same as disk 1 but for banked memory.
Disk 3:
ASSIGN.COM HELP.COM INITDIR.COM GENCPM.COM HELP.HLP
Disk 4:
BACKREST.001 BACKREST.002 BACKREST.003 BACKREST.COM
BIOSKRNL.ASM CALLVERS.ASM COPYSYS.COM DRVTBL.ASM
ECHOVERS.ASM MOVE.ASM SCB.ASM BOOT.ASM COPYSYS.ASM
CPMINIT.COM DUMP.ASM FD1797SD.ASM RANDOM.ASM SETUP.COM

disk 5:
SID.COM GENCOM.COM PATCH.COM XREF.COM SAVE.COM CPM3.LIB
RMAC.COM Z80.LIB ED.COM LINK.COM MAC.COM CPMLDR.REL
DEVICE.COM HEXCOM.COM PORTS.LIB TRS4PORT.LIB DIRLBL.RSX
MODEBAUD.LIB LIB.COM



CPM - A NEW USERS TUTORIAL

A few years ago some of us may remember using TRSDOS for the Model I. It was noted for having few useful utilities and somewhat awkward and cumbersome to use. Since then many new versions of this Disk Operating System (DOS) have been produced. All, however, are along the same lines with similar directory files and BOOT sectors. This provides a degree of compatibility between the various operating systems.

Another system was devised in the late 1970's which bore no resemblance to TRSDOS, but provided a greater degree of compatibility between different types of machine. It was created by Digital Research in the USA and was called the Control Program for Microcomputers (CP/M). It was hailed as the "industry standard operating system", so a large number of programs were written to run under it. In those early days many machines contained only RAM and had no system or BASIC in low memory ROM, therefore CPM operated around the low addresses. When the earlier TRS 80's came out they had their ROM in the low memory area and so they could not run under CPM. With the advent of the Model II and the Model IV, which used all RAM, CPM could be successfully run on them.

The system was, as I have said, designed to provide "compatibility" between different types of machine, but in fact in reality only "transportability" has been maintained. That is to say programs may be transferred from machine type to machine type, but there is no guarantee that they will run. Most of these earlier programs were kept in their early format so that they would still run under the newer systems coming onto the market. For example Wordstar was an early and powerful word processing package, nowadays it is not as good in some aspects as AllWrite or LeScript, but these latter two are only for the TRS 80 and do not run elsewhere. CPM means that if a firm gets a new machine running CPM then the typist using the old Wordstar will not have to relearn new commands before operating the new package. As a result the keys used to delete, insert, and suchlike within Wordstar have changed little since it's inception, thus the commands are not easy to remember ie CTRL D and CTRL I for delete and insert respectively are simpler then those used in Wordstar. I am not trying to knock Wordstar but just to demonstrate why the commands within CPM have remained simple and unchanged over the years. Again the onward compatibilty has to be maintained not only for all users but also to keep all machines still compatible. In addition with the lack of Cursor keys on some computers and differing methods of addressing peripherals and suchlike the commands within CPM were also kept simple, in fact there are only 5 commands as standard within the operating system although a few more "standard" utilities have been added since then.

It is worth noting that the 5 commands are contained in the system tracks, normally just 2, and they are comparable to the Library Commands in TRSDOS, which by comparison takes many more than 2 tracks to hold the system.

Lets look at those 5 commands plus one other usually found within the operating system. They are

TYPE, DIR, ERA, REN, SAVE and USER

TYPE

This command is similar to the LIST command in the TRSDOS except that it has no

parameters. It is used to list text files to the screen, and if a printout is required then CTRL P must be pressed prior to the TYPE command, this control sequence will route all screen output to the printer, a sort of JKL function.
The format is as follows
A>TYPE DEMO.TXT
A>TYPE B:DEMO.TXT
The "A>" in the above example is the CPM prompt when using the drive called A. From the above it can be seen that the standard extension type can be used for ASCII text files, but you should note the the extension is not preceded by the "/" used in TRSDOS but by a full stop. Also the second example demonstrates the use of drivespecs in CPM.

All drives are refered to by a letter, this is it's logical drivespec although it's physical drivespec is still a number as in TRSDOS. The drivespec, in this case logical drive B, is put in FRONT of the filespec and is separated from it by a colon. Pressing CTRL P before the above commands will direct output to the printer, remember to type CTRL P afterwards to cancel output to the printer.

DIR

This is the directory command for CPM, and again is similar to TRSDOS DIR with except that it has fewer parameters.
A>DIR
This command will produce a list of all files on the current drive and produce a display like this.

```
A: MOVCPM     COM :  ASM      COM :  DDT      COM :  PIP      COM
A: ED         COM :  STAT     COM :  DUMP     COM :  DUMP     ASM
A: USERS      LIB :  SUBMIT   COM :  DEFAULT  SUB :  CAT-ONE  TXT
A: WS         COM :  WS       DOC :  BILL     HEX
```

The drivespec is to the left and the extension is offset and left justified; output to the printer is made by pressing CTRL P as for the TYPE command. At this point I will say a few words on filespecs, as you can see from above they may contain some characters not used in TRSDOS filespecs. In fact any character may be used in any position, including the first with the following exceptions.

< > . , ; : = ? * [ ]

There are no passwords allowed in CPM. Other allowable formats are
A>DIR B:
This will give the directory for drive B. Note that drives must be specified separately, there is no command for the directory on both drives, in fact if a program on drive B is called when using drive A, it will not be found unless B: is typed immediately in from of the filename. Another method is to move onto drive B and execute DIR from there, this takes longer but you remain in drive B as the default after the command. This is achieved as follows
A>B:
B>DIR
Note that the prompt character changes from A> to B> after the first command. Typing
A>B:DIR
will execute the DIR command from drive B's operating system but return a directory of drive A. The computer effectively runs a program called DIR on drive B whilst in the default drive A. A specific program or programs may be displayed by use of the following commands.
A>DIR B:ABC.COM
A>DIR B:A?C.COM
A>DIR B:A*.COM
A>DIR B:ABC.*
A>DIR *.COM
A>DIR FILE1.*
A>DIR *.*
In the above examples the first four will all display the program ABC.COM, on drive B, the ? is used to mask out specific characters and the * to mask out more than one character. From this it can be seen that the last three examples will display respectively,
STAT.COM
DDT.COM
WS.COM
FILE1.COM
etc in the first case.

FILE1.COM
FILE1.DAT
FILE1.TXT
etc in the second
All files on drive A in the last

Finally a word of warning. CPM does not check the disk for it's directory, but gets it from memory once it has been initially read in. This means that if you change disks, CPM will not realize this and either report a file not available or more disastrously, write new data all over the wrong places on your disk. To cure this you MUST type CTRL C, to force a directory read from disk, after all disk changes; this practice must be used from the first time you power up CPM so that it becomes second nature. On the Model 4 the BREAK key usually returns a CTRL C.

ERA

This is the erase command and is similar to the KILL and PURGE commands of TRSDOS, it uses extensions and wildcard masks in the same way as DIR does
A>ERA FILE2.COM
will erase the program FILE2.COM
A>ERA FILE2.*
will erase all FILE2 files irrespective of extension
A>ERA *.ASM
will erase all files with the extension ASM
A>ERA *.*
will erase all files from the disk having first prompted you to
ALL (Y/N)?
replying Y will delete all files, N will abort the command

ERA is a fairly powerful command and any experimenting on filespecs should be done using the DIR command.

REN

REN is the rename command and is used to change the names of CPM files. It takes the form "newspec=oldspec"
A>REN FILE2.TXT=FILE1.TXT
will rename FILE1.TXT TO FILE2.TXT, note that this is the opposite way round to TRSDOS. The drivespec is optional and
A>REN B:FILE2.TXT=FILE1.TXT
A>REN FILE2.TXT=B:FILE2.TXT
will have the same effect as the first example when using drive B.

SAVE

This command will save an area of memory to disk, it ALWAYS starts at the same point and saves files using "page" numbers. A page is 256 Bytes of memory and the SAVE command always starts it's save at 0100H, which is the start of the program area. You will need to work out the size of the program you are using prior to a save, and this is best done using the program called DDT.COM, which is on your CPM disk. For now all you should worry about is the format of the command: it's uses and exactly how to calculate program size will be covered later.
A>SAVE 4 FILE3.COM
will save a 1K program called FILE3.COM starting at 0100H in memory.

In my early days I would over-estimate the value for the pages and thus guarantee the program being entirely saved. This command is worth more discussion, and at this early stage it is unlikely that you will need to use it and so I will mention it later when discussing the DDT.COM program

USER

This command is not found in all versions of CPM and in some version may exist as a command file called USER.COM, note that the effect will be the same providing USER.COM is resident on your disk. "COM" files, I should explain at this point are the same as TRSDOS "CMD" files.

```
A>USER 1
```
will define a new user area called USER 1. This needs some explaining. The system initializes using user area 0, and on some versions of CPM this is indicated by the CPM prompt being
```
A0>
```
This is the lowest user area and usually most disks have all their files within this area. If a file is saved whilst in another user area, it will become invisible to users in the original area.
So if you type
```
A>USER 1
A>SAVE 12 NEWFILE.DAT
A>USER 0
A>DIR
```
Then the file NEWFILE.DAT will not be visible to people only using user area 0 typing DIR whilst in user area 1 will only display the file NEWFILE.DAT, unless other programs have been saved in it.

That covers the commands resident within the operating system, the commands usually found as COM files are called transient commands and may vary from system to system. I will cover the usual ones that you will encounter in the next article. Finally I would like to point out that I am no expert on CPM, and in order to produce this article I have had to "bone up" on CPM; a useful book to help me in this has been, " The CP/M Handbook with MP/M" by Rodney Zaks. It cost me £11.95, but has proved worthwhile, in helping me understand this new and strange operating system.

IN MAV'S WORKSHOP

FreHD AUTO BOOT EPROM PART 4: HOW TO INSTALL IT ON THE MODEL 4P PORTABLE COMPUTER

By Ian Mavric

In previous issues of TRS8Bit I've discussed how to install the FreHD Auto-boot Eprom into a TRS-80 Model III and Model 4NGA (easy - just remove an old Rom chip and replace it with a newly programmed Eprom), and a Model I (difficult - requires soldering, and a switch as well as 6 wires), and last time how to install it on the 4GA and 4D. This article explains how to add the FreHD auto boot Eprom to a Model 4P Portable Computer (cat. no. 26-1080 and 26-1080A). It's more difficult that adding the Eprom to the Model III and 4NGA but less difficult than adding one to a Model I, 4GA or 4D. Soldering is required so as I always say, if you are not confident in soldering, get someone else who is skilled to undertake this modification for you. Since no traces need to be cut the modification is easily reversible if things go wrong.

Items required:

Eprom: 2732
Wires: Grey 3cm and 15cm

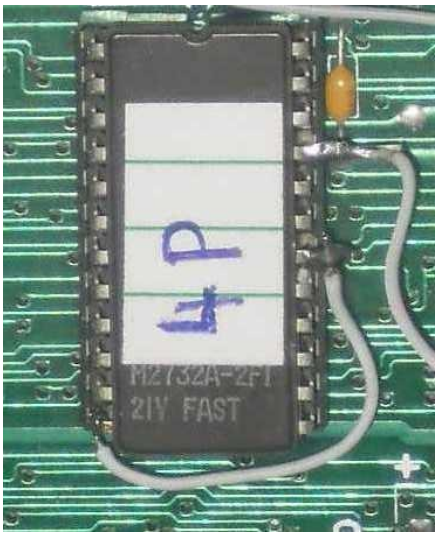
Firstly you need to program the Eprom with this BIN file from the Downloads page on my web site:

http://ianmav.customer.netspace.net.au/downloads/model4p_FIX-I_rom.bin

Once programmed you need to do the following work to prepare the Eprom for installation in your computer:
Bend out pin 18 and solder the short wire to it.
Solder the other end of that wire to pin 12 near the top where it enters the chip, that way the pin can still go into the IC socket.

Bend out pin 21 and solder the long wire to it.

It should look like this:

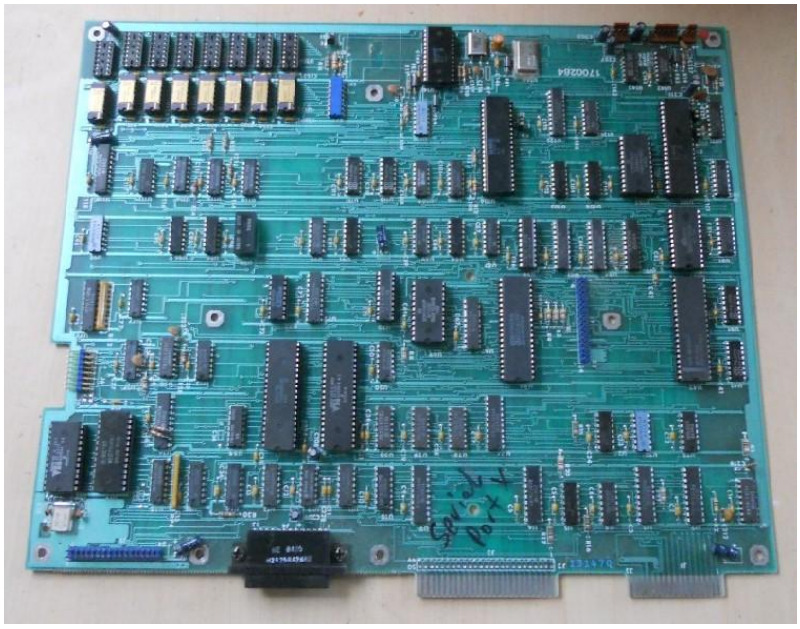


Picture: pin 18 connects to pin 12 and pin 18 goes through a PCB hole and is soldered under the motherboard

Installing the Eprom:

You will need to remove the motherboard from your 4P and this is quite a job due to the rugged nature of the design of the 4P, it's designed to be more sturdily than it's desk-bound brothers. The outer case it opened by removing the four white screws (2 either side) of the 4P case and then two more under the carry-handle. The case just slips off. Once inside you need to loosen the back black cover, and then the motherboard tray, which its held in place by eight screws (4 per side). After you unplug the disk drive cable you can fold the motherboard out.

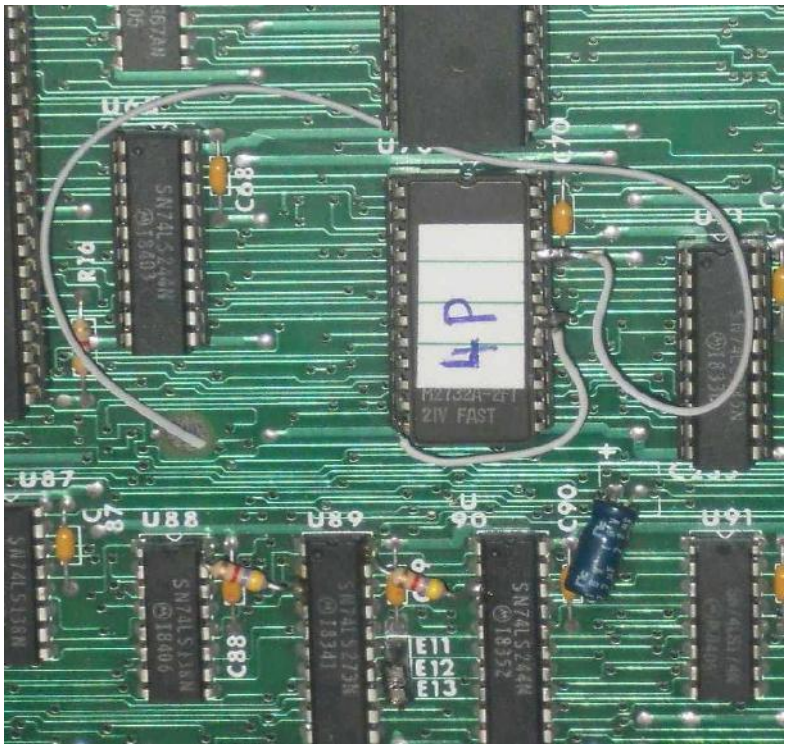
Since a wire needs to be soldered to the underside of the motherboard, you need to disconnect the motherboard connections (video, power, reset, ground) which will remove the motherboard from the rest of the system. Now you can undo the screws which hold the motherboard to the metal tray. When you remove the motherboard from the metal tray you can unplug the keyboard. Your motherboard looks like this:



Picture: 4P Gate-Array motherboard, 4P Non-Gate Array motherboard is similar.

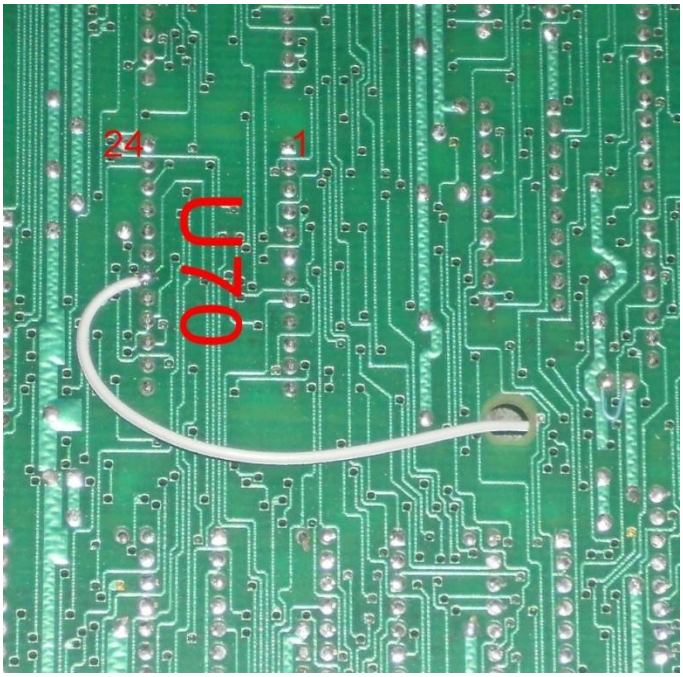
On the 4P Non-gate array (26-1080) the chip you are looking for is in the centre of the motherboard marked U70 and this may well look like the Eprom you just prepared with the wires. If so, life is easy, unsolder the wire which runs to pin 21 and install the Eprom into U70 socket. You are done.

If U70 has a regular 24-pin Rom, then you need to remove it and install the newly prepared Eprom into U70 socket. The wire attached to pin 21 needs to go through the hole in the motherboard right above U88 and it gets soldered to pin 18 of U70 socket. (Remember how pin 18 of the 2732 Eprom was bent out so it didn't go into the socket? The aim here is to connect pin 21 of the Eprom to where pin 18 would have gone.)



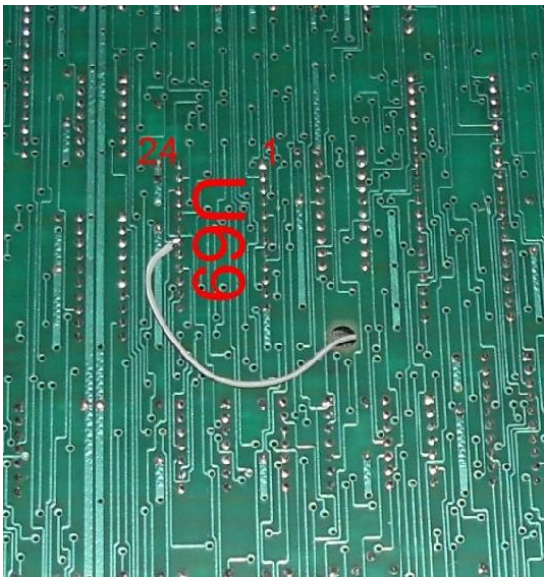
Pictures: wire goes through hole near U88, and comes out the under the motherboard

Since the motherboard picture above is a little hard to see, here is a close-up where the wire gets soldered (U70 pin 18):



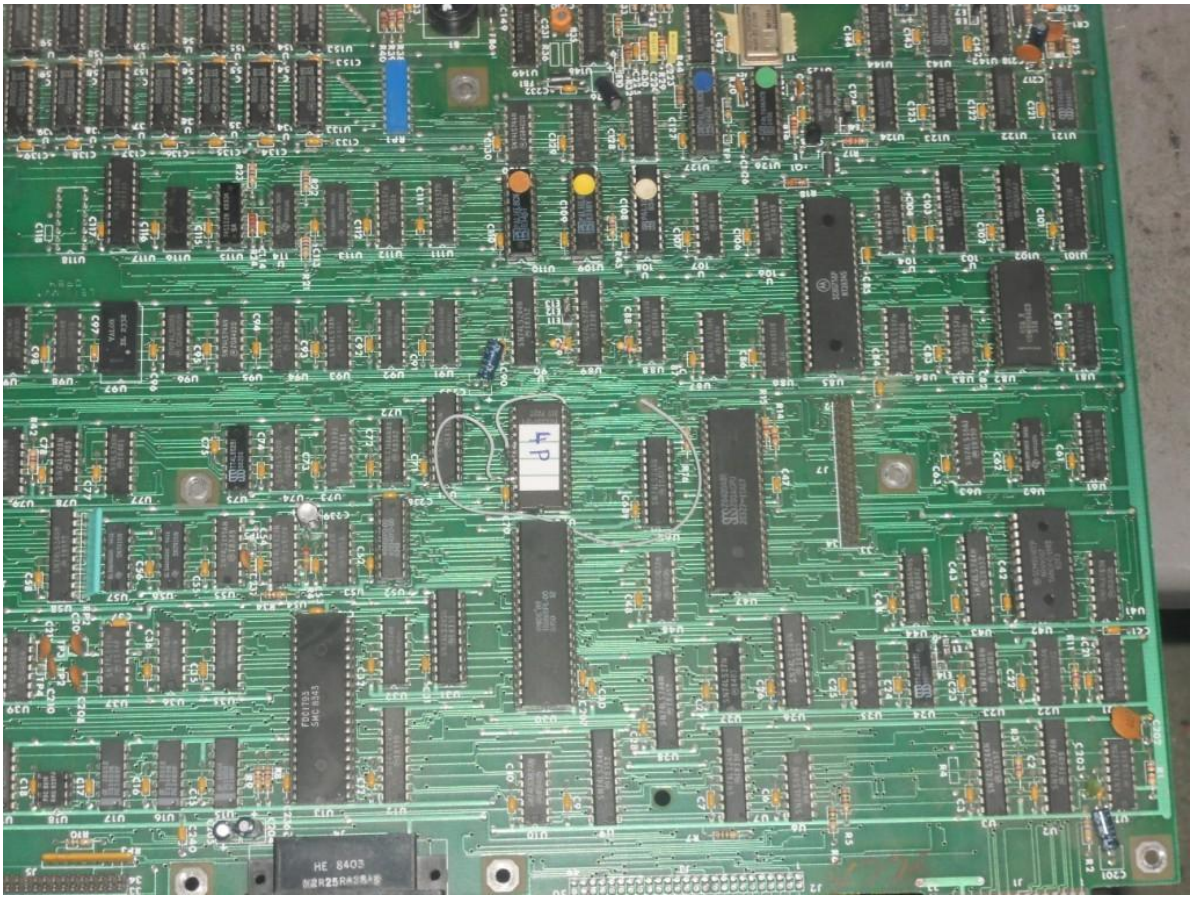
On the 4P Gate Array (26-1080A) is a similar process. By the time the 4PGA was released the Eprom with the wires was gone and a Rom was installed in it's place, this time in U69 but its still easy to find right in the middle of the motherboard.

U69 has a regular 24-pin Rom, and you need to remove it and install the newly prepared Eprom into U69 socket. The wire attached to pin 21 needs to go through the hole in the motherboard right above U88 and it gets soldered to pin 18 of U69 socket. (Remember how pin 18 of the 2732 Eprom was bent out so it didn't go into the socket? The aim here, again, is to connect pin 21 of the Eprom to where pin 18 would have gone.)



Picture: underside of the 4PGA motherboard shows where the wire goes (U69 pin 18)

Fitted it looks like this:



Reassembly and Testing:

Putting the computer back together is essentially a reversal of the steps used to disassemble it taking note of the following:

1. When you put the motherboard back on the metal holding tray, you need to plug in the keyboard before you install the screws which hold it to the tray.

2. The power, video, reset plugs and ground tag at the front of the motherboard can be difficult to get connected so take your time and use long-nose pliers.
3. Once the motherboard tray assembly is re-installed don't forget to plug in the disk drive data-cable.

Before connecting your FreHD, turn the 4P on and observe the three language boot error message comes up as normal, then test a few of your favourite floppy disks to make sure the disk drives work properly.

You can now connect your FreHD noting the following:

1. The data cable exits down from the 50-way I/O bus on the Model 4P. It doesn't exit upwards even if you plan to sit your FreHD on top of the 4P (which most people do). The red-stripe on the data cable is on the printer-port side not the RS232 port side.

2. Power your FreHD up before the 4P and wait for the single flash of the green LED before powering up your 4P.

3. You will find that the FreHD auto-boot images for the Model 4 work: LS-DOS 6.3.1 and CP/M 2.2, but Model III mode images don't and won't appear on the FreHD boot menu unless you have a copy of modela/iii ROM image in the root directory of the SD card. Once that image is in place all Model III mode auto-boot images become available. If you need assistance to get model iii copied over to your SD card, shoot me an email.

Ian Mavric

ianm@trs-80.com

Ian Mavric is an IT Specialist who also restores and collects TRS-80's and classic cars. He live with his wife and kids in Melbourne, Australia.

(CONTINUED FROM PAGE 1)

ARTICLES IN THIS ISSUE, FROM THE DAY.
THEY INCLUDE A BEGINNERS GUIDE, A REVIEW OF CP/M V3, FOR THE M4 AND A REVIEW OF THE OMIKRON MAPPER.

PASCAL HOLDRY HAS, AS USUAL, BEEN BUSYING HIMSELF WITH ALL SORT OF HARDWARE ISSUES. FOR FULL DETAILS, PLEASE CHECK OUT HIS WEBSITE. THERE'S A LINK ON TRS-80.ORG.UK

MAV GIVES US THE LOW-DOWN ON CO-CO DISK DRIVES, PART 2 OF HIS 'SOLE' ARTICLE WITH THE PERC-DOUBLER AND THERE'S PART 4 OF HIS AUTO BOOT EPROM FOR THE M4P

FINALLY, THERE IS THE RESULT OF THIS YEAR'S ONE-LINER COMPETITION. THERE WERE ONLY FOUR ENTRIES THIS YEAR, BUT DEE HAS AWARDED THE PRIZES TO **GEORGE PHILLIPS** WITH HIS "PLANET & MOON" PROGRAM. THE OTHER 3, FROM MAV, PETER PHILLIPS AND ET FONEHUME ARE GIVEN HONOURABLE MENTIONS BECAUSE THEY ARE FASCINATING PROGRAMS. THANKS TO EVERYONE WHO TOOK TIME TO ENTER.
SO, ALL THAT REMAINS FOR ME TO DO IS THANKS EVERYONE WHO HAS CONTRIBUTED ARTICLES OVER THE LAST YEAR. PLEASE STAY WITH US FOR 2017 (OUR 11TH YEAR)

HAPPY CHRISTMAS EVERYONE - DUSTY

TRS-80 Emulators . com

TRS32: A Model I/III/4/4P Emulator For Windows

written by Matthew Reed

Unregistered Shareware Version:

- Works under all current versions of Windows
- Full Windows application — no low-level hardware conflicts!
- Model I, Model III, Model 4, and Model 4P emulation
- Four floppy disk drives (with optional realistic disk drive sound)
- Cassette tape drive with graphical on-screen controls
- Exatron Stringy Floppy emulation
- Printer support
- Serial port for RS-232 communications
- Joystick support (using a Windows joystick — TRISSTICK and Alpha Products joysticks are emulated)

Registered Version:

- All features included in the shareware version
- Built-in emulation of an Epson FX-80 dot matrix printer (including graphics and control codes)
- High resolution graphics (Radio Shack and Micro-Labs)
- Up to 1 megabyte of additional memory in Model 4 and 4P modes
- Hard disk support
- Orchestra 85/90 music generation

Interested?

- Read the TRS32 emulator documentation
- Download the shareware version
- Register online

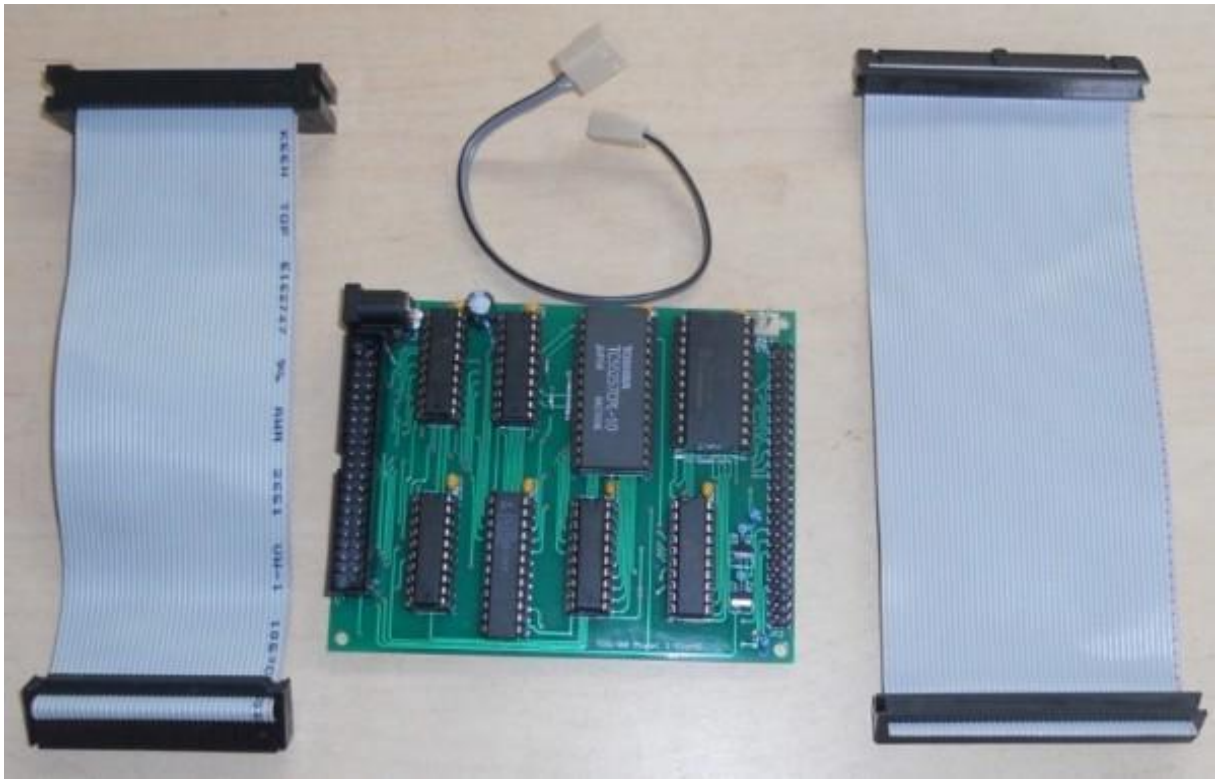
Actual Dotwriter output — emulated FX-80
click for larger sample

MATTHEW'S EMULATOR IS QUITE ASTOUNDING.
IT'S MY FIRST CHOICE WHEN DEVELOPING IDEAS FOR TRS8BIT.
HE OFFERS FULL SUPPORT AND IT IS STILL UNDERGOING ENHANCEMENT.
IF YOU HAVEN'T REGISTERED YOUR SHAREWARE VERSION YET, PLEASE DO SO AND ENCOURAGE
AND SUPPORT MATTHEW'S ENTHUSIASM TO CONTINUE WITH HIS WORK.

ED.

“Quinnterface” Mini Expansion Interface for 16K Model 1 FreHD users.

- COMPLETLY ASSEMBLED AND TESTED -



This is THE perfect device for all Model 1 users who own a 16K Level II unit, (which is most of us!), but no Expansion Interface or disk drives, especially if you don't want to modify your M1 with upgraded boot ROM or memory upgrade.

The ‘Quinnterface’, developed by J. Andrew Quinn from New Zealand, adds 32K RAM and auto-boot functionality to you FreHD.

U.K. And Europe, contact Bas. at BetaGamma Computing.
U.S.A. And rest of the world, contact Mav. At “The Right Stuff”

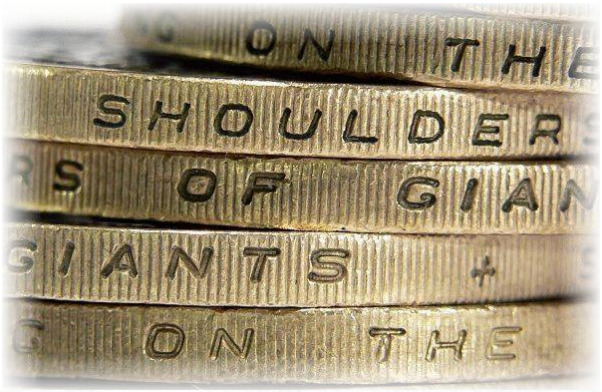
LOOKING FOR FAST, INEXPENSIVE, UNLIMITED MASS STORAGE FOR YOUR TRS-80 MODEL I/III/4/4P/4D?

The amazing

"FreHD"

-Emulates a TRS-80 hard drive, but faster than any hard drive!
-Works with your favourite DOS (LS-DOS, LDOS, CP/M, Newdos/80 2.5)
-Uses SD card for storage medium
-Bonus free Real Time Clock function!
-Designed in Belgium and proudly built and shipped from Australia
-Kit form or fully assembled

Order yours today:
http://ianmav.customer.netspace.net.au/trs80/emulator



USING NEWTON'S QUOTE "STANDING ON THE SHOULDERS OF GIANTS" AS INSPIRATION, I OFFER THE FINAL IN THE SERIES OF SOME OF THE KNOWN, AND LESSER-KNOWN NAMES WHO INSPIRED MUCH OF WHAT WE, TODAY, TAKE FOR GRANTED.

Alan Turing

This British mathematician gave his name to the accepted test for machine intelligence. Much of his work, however, was for military intelligence during the war

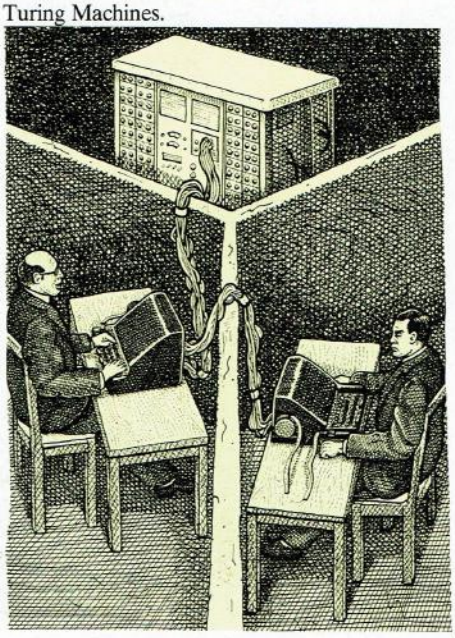
Mathematical Feat
Alan Turing (1912-1954) found inspiration and relaxation through long-distance running. He was intrigued by the effect of physical exertion on creativity and mental agility

The young Alan Turing showed a remarkable insight into science. He wrote to his mother from school 'I seem always to want to make things from the thing that is commonest in nature'. Mathematicians show their talent early and as soon as Turing could read and write he was factorising hymn numbers and designing amphibious bicycles.

While his father was away in Madras working in the Indian Civil Service, Turing was winning school prizes and then the scholarship that took him to King's College, Cambridge. It was at Cambridge, first as a student and then as a fellow of King's, that his interest began to focus on the problems of mathematical logic.

In 1931 the Czech mathematician Kurt Gödel astonished the scientific world with the discovery that there were mathematical theorems that were true yet could never be proved. Alan Turing set out to investigate those which could be proved.

He proposed a machine, the construction of which he left to the imagination, that could carry out mechanically the processes usually performed by a mathematician. For each process there was one machine — for example, a machine to add, another to divide, and a third to integrate and so forth. These machines later came to be known as Turing Machines.



BOB VENABLES

Can Machines Think?
To answer this question, Turing proposed his famous test, called the Imitation Game, but which has subsequently become known as the Turing Test. A man is put into a room that features a teleprinter (keyboard-cum-printer). This is linked to a teleprinter in another room, operated by another man; and also to the computer under test. The first man is allowed to ask any questions he likes of either. If he is unable consistently to determine when he is communicating with the man and when with the computer, then the machine may be deemed to be intelligent. After all, the argument goes, we have no way of telling for certain whether other people do think or are conscious, except by a comparison of their reactions to circumstances with our own

Turing investigated the workings of these imaginary machines and came to a remarkable conclusion. Rather than each mathematical process needing a separate machine, it was possible to design a 'universal' device that could be made to imitate any other of the specialist machines by being 'programmed'. Turing had stumbled upon the theory of the programmable computer.

When the Second World War broke out Turing was quickly recruited from the academic world to the Government School of Codes and Ciphers at Bletchley Park, Buckinghamshire. Had it not been for the war, his machines might have remained imaginary, but Bletchley Park was involved with the highly secret and urgent work of breaking German military codes.

Because these codes could be changed each day, machines were needed to crack the ciphers before new ones were introduced. Bletchley Park became a huge information processing centre. In the middle of the war Turing was sent to America to establish secure codes for transatlantic communications between the Allies.

The secret nature of his work at this time means that few records of his movements are available. However, it is widely supposed that he met Von Neumann while at Princeton, New Jersey. Towards the end of the war Turing was asked to draw up plans for an all-British computer for the National Physical Laboratory, to be called ACE.

The Automatic Computing Engine was named partly in honour of Babbage's Analytical Engine. Like this pioneering machine, ACE took a long time to be constructed, but in many ways it was far in advance of ENIAC.

Frustrated at the slow progress, Turing resigned and moved to Manchester where he joined the university's computer project. At the same time he became a consultant to the Ferranti company and subsequently became involved in the first computers to be built in Britain.

Turing was an eccentric who pursued what he knew to be important without regard for social conventions or legal constraints. A friend said he was 'divinely retarded' when it came to seeing faults in others, but his scientific genius was flawless. In 1952 he was convicted on charges relating to homosexuality, and committed suicide two years later. Who can tell what a contribution Turing might have made to artificial intelligence, had he still been alive today?

Mav Quarterly

Updates on the state of the Biz, in no particular order

by Ian Mavric

Kit costs: I sell two project kits currently which are designed for those skilled in soldering to both partake in the enjoyment of building up a working electronic attachment for their computer and save them some money at the same time. These are the FreHD CKD kit and the RE-FD502-KIT floppy disk controller for the Coco. Some have commented on the price of these kits, being seemingly higher than that of similar kit projects back in the day. Yes I would like to be able to bring these kits to the marketplace more cheaply but please look past the low cost parts in them and recognise some parts are hard to find and/or expensive. The best example of this is the WD1773 disk controller IC in the RE-FD502-KIT which costs me around $20 in a kit I sell for $40.

Kit components: Following on from the above topic, the ever popular FreHD Kit A and Kit B are geared toward those who have parts drawers of resistors, capacitors, LEDs etc. If you are buying, say, a Kit B to save $ and have most of the other parts, ask me about the parts you need to complete the device, I'm happy to sell them at the going rate and include them in the package with your Kit B. This way when it arrives you can start soldering and have your project up and running quicker than if you received it, and need to order a handful of parts from and wait for those to be delivered before commencing.

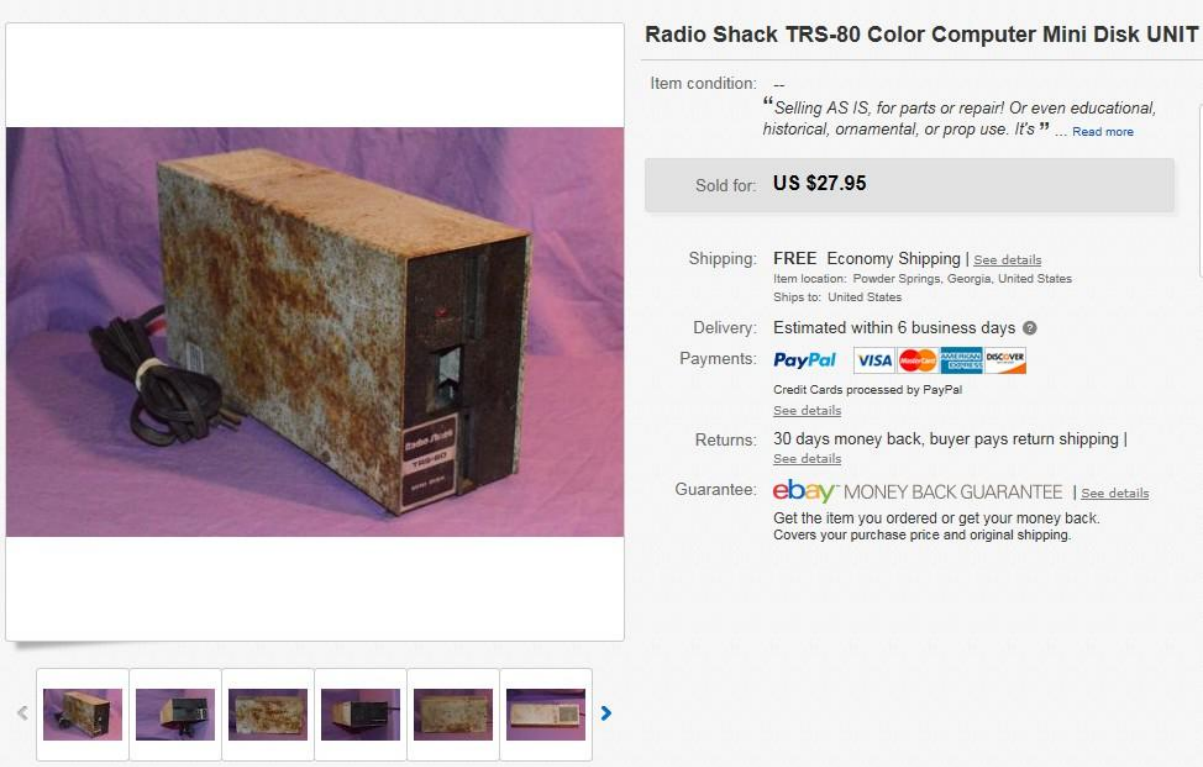
Important cable reproductions for the Model I: In the last few months I have brought to the marketplace some important cables which replace those which if they are missing or not working properly, will severely limit what you can do with your Model I. The first is the NewBUFF cable which is a reproduction/replacement of the Buffered Interface Cable used on early Expansion Interfaces. At $45 it's cheaper than many of the used 35+ year old Buffered Cables which appear on eBay. Email me if you need help deciding whether you need one of these for your Expansion Interface. I can work out what is needed from a couple of photos of your EI PCB. The second cable is the RS232 cable reproduction, Oftentimes people buy an EI which has an RS232 board inside it and are missing the all-important cable. Or you buy an RS232 board off eBay or CL and it doesn't come with the cable. The reproduction serial cable can get you connected up to you serial device with ease.

Coming in 2017: Other things I have been working on is a reproduction of the printer interface cable for the Model I which plugs directly into the keyboard unit, a RS232C serial board for the Model I Expansion Interface, and a version of the RS232 cable which terminates in a DB-9 connector instead of the original DB-25. No dates are set thus far but if you are interested in any of these future products drop me a line.

South America: Once again a parcel shipped to South America, this time Mexico, has gone into the snail-mail-deliver-it-when-we-feel-like-it cue at the local mail sorting and delivery warehouse. I feel sorry for the buyer waiting patiently for his items. The rationalist in me says I should just do what annoyed eBayers do, which is simply limit sending their wares to countries which delivery is consistently good, and exclude everyone else. To my mind this is a cowards way out as we are all trying to grow the vintage computer hobby and community and restricting sales to "easy" groups is a cop out. I think the best way forward is to prepare an email stating that while roughly half the shipments to South America go smoothly, the other half get held up for whatever bogus reason (Australia post advises me the date that parcels arrive in South America, usually 5-6 days after I send them, the rest of the wasted time is due to the postal systems in their respective South American countries), stating clearly that if you really REALLY want their items to arrive in a timely, fully trackable manner, you need to pay for a more professional service, like EMS.

Where is the Order button?: A few people over the years have asked me how to order items from my web site, since there is no clear ORDER button like on other web sites. Email is your friend here and I usually respond within 12 hours. If I'm online when your email arrives you may hear from me straight away. The reason for this is because so many items I sell are system specific and usually require additional questions to make sure that the thing you are ordering will work on your particular system. The same goes for my eBay store where most things are shipped as they appear on the site however I may ask a couple of questions just like if you'd ordered it from my web site. A classic example is people who buy an Improved Grafyx board for a 4P and order the Model 4 version: people new to TRS-80s often don't know what they need to achieve a particular result. I aim to make sure your project or upgrade goes well.

Next Year's Competition: In the last 12 months I've seen a growing number of really crap-tastic badly looked after TRS-80 items appearing on eBay. Oftentimes, I am horrified... this stuff was EXPENSIVE when new. As a teen when this stuff came out I know how long it took to save for each item doing menial jobs to buy an Expansion Interface, Disk Drive, and Daisy Wheel printer, just so I could do more menial jobs (programming and WP) to save more money to buy more of this expensive hardware. Seeing it in a poor state these days is disheartening, but quickly becoming commonplace. Lets have a competition to see who has the worst cared for piece of hardware which still functions. I call the competition "I Bet They Couldn't Wait To Get It Home". Below is a picture of the sort of thing I'm talking about:



The Right Stuff: One stop TRS-80 shop for restored systems, hardware upgrades, spare parts, books, magazines, software, cables and advice.

Ian Mavric

ianm@trs-80.com

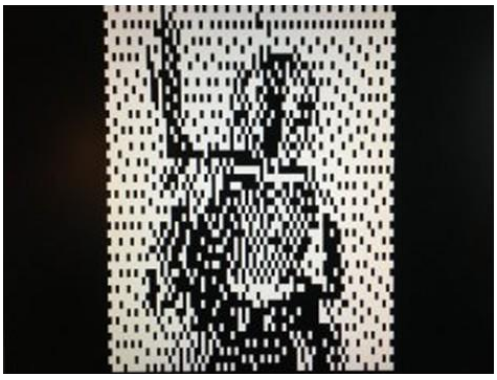
TRS-Pic.

Coming soon for your MISE equipped TRS-80 Model I or your M3SE equipped TRS-80 Model III or 4. Supports both 128x48 standard TRS-80 resolution or 640x240 hi-resolution with an optional Hi-Res graphics card. Full details from - http://pski.net/trs-pic/

Once you unzip, you can copy the appropriate commands for your TRS-80 using the MISE/M3SE FTPD service. Be sure to transfer using Binary mode. See the README.TXT for more details.

You can search for images just like you do in Google Images. Here is what the TRS-Pic title screen looks like on a TRS-80 Model I.

Here is Boba Fett as viewed on a TRS-80 Model 4 with a Hi-Res card.



Here is Boba Fett again, this time viewed on a TRS-80 Model I with standard 128×48 resolution.

Here is Boba Fett again, this time viewed on a TRS-80 Model I using the MISE PCG-80 emulation on a VGA monitor.

The *BEST* in TRS-80s

Call The Right Stuff

Ask for Ian
The number is +61 416 184 893

That's The Right Stuff
And he's in Melbourne

http://ianmav.customer.netspace.net.au/trs80/