



SYDNEY TRS-80 USERS GROUP NEWSLETTER

P.O. BOX 297, PADSTOW 2211.

Volume. 6 Issue. 7 March 1986

<u>Title</u>	<u>Author</u>	<u>Page</u>
Editor's Soapbox	Gary Bryce	2
Secretary's Sayings	Darrell Hegarty	2
General Meeting Report	Jim Whittaker	3
Printer Buffer Reveiw	Robert Gareb	3
Letter to the Editor	Algis Bizys	4
The Comm Line	Michael Cooper	4
TRSDOS 6.x Spooler	Jack Decker	6
Model 4/4P ROM Changes	Jack Decker	6
One Meg RAM for your TRS-80	Gary Bryce	9
TRSDOS Patch Error	Gary Bryce	10
The Professional Dos	Michael Cooper	10
SuperSCRIPSIT Model 4 Patches	Jim Whittaker	11
Fast Fix for Model 4 Slows	Dave Owens	11
Public Domain Disks		11

Meeting News

As always the first and second monthly meetings will be held at the rear of Pattersons Florists, Botany Rd, BOTANY (entrance from Chagwyn St) while the third meeting (on the fourth Saturday of the month), will be held at the 1st Sefton Scout Hall, 2 Waldron Rd, SEFTON, all meetings commence at 1 PM on the following dates :-

March 8th	Botany	April 8th	Monthly General Meeting
March 15th	Botany	April 15th	Special Interest Meeting
March 22nd	Sefton	April 22nd	South West Meeting

Who's Who

President	Ted Romer	498-2399
Vice President	Jim Whittaker	772-2009
Treasurer	Greg North	635-1438
Secretary	Darrell Hegarty	560-9681
Newsletter Editor	Gary Bryce	628-5058
Club-80 Sysop	Michael Cooper	331-7136
CBBS Secretary	Peter Wignell	759-8024
Hardware Co-ordinator	Errol Rosser	709-7646

SYDTRUG Bulletin Board

CLUB-80 Bulletin Board operates for members, seven days a week twenty four (24) hours a day on (02) 332-2494. The data format used is as follows :- 8 data bits, 1 stop bit, No parity, Full duplex, CCITT V21 modem standard 300 bps (set your modem to ORIGINATE mode). Limited access is granted for visitors.

The contents of this publication are (c) 1986 by SYDNEY TRS-80 USERS GROUP. All rights reserved. Enquiries should be directed to the "Secretary, Sydney TRS-80 Users Group, P.O. Box 297, PADSTOW N.S.W., AUSTRALIA 2211. Material appearing in this publication may be reprinted in similar computer club newsletters and nonprofit publications if accompanied by the following notice:

Reprinted from SYDTRUG NEWS, P.O. Box 297, PADSTOW 2211.

EDITOR'S SOAPBOX

by Gary Bryce (02)628-5858

Those of you with Model 1's, III's and SYSTEM 88's may wonder why there is not much in this issue for you. There is a simple answer, I can't print that which I have not received!! So get to your keyboards and start typing.

The group bar-b-que has come and gone, and I dare say due to the poor attendance will probably be the last. Those eight members and family who attended had a good time as the venue was excellent (I got a good appreciation of the place myself when I searched from one end to the other for my 11 year old nephew - who turned out to be playing in a sandpit not 15 metres from our table).

On another subject, I received a phone call from Peter Goed of the Brisbane TRS-80 SYSTEM 88 Computer Group and he mentioned in passing that Warwick Sands has developed an improved version of NEWDOS/88 (for the Model 4, I think) that should be available sometime next month. I will reserve judgement until I see it but it will have to be good to shift me from LDOS/TRSDOS 6. I know there are a lot of NEWDOS devotees out there but the inconsistencies in command syntax and those horrid PDRIIVE hassles when trying to read someone else's disks, truly leave me cold.

Maybe others don't use the extensive JCL, ROUTING, LINKING and FILTERING capabilities of the LSI DOS's (not to mention the ease of assembly language programming with the SVC calls), but I do, and I would be lost without them. Add to this the fact that I can take any LDOS/TRSDOS/VTOS (plus single sided DOSPLUS and single density NEWDOS and MULTIDOS) disks and read and write to them without having to find out how many SPT's, DDGA's, GPL's etc, there are on the disk before I can read it and you begin to realise why I switched from NEWDOS some two years ago.

I would recommend NEWDOS to anyone with only one disk drive (as LDOS won't allow a single drive BACKUP by itself) and to anyone who requires the extended file handling capabilities of the Disk BASIC interpreter supplied with NEWDOS. But these are the only times that I would make such a recommendation.

Anyway, enough for now see you all at the next meeting!

SECRETARY'S SAYINGS

by Darrell Hegarty (02)368-9681

As I write this column, I am reminded by a phone call I received last night, of MODEM88 for the SYSTEM88. This program, as we all know, would not exist but for one of our fellow SYSTEM88 owners doing the quite considerable modifications, and otherwise would not be available at any cost! Just a thought of the effort that goes into owning a computer, and banding together as a group to help each other. This is the real spirit of SYDTRUG.

We have purchased a Model IV for the Bulletin Board, which (hopefully) will enable the software to run faster, and allow better output when ultimately operating at 1200 baud, together with the hard disk - yes - the hard disk is now on again, and is being repaired as a matter of urgency for us. This will now give us a spare Model III which will be kept as a backup for the BBS, and will be used at Botany meetings to reproduce the Public Domain software on the spot, as well as for demonstrations and other club uses. It will also be used by the committee for club duties - yet to be decided.

At the last Botany meeting, it was mentioned that a purchase of basic kits for a Printer Buffer would be raised by the Group at a cost of \$35.00 each. Members present were asked to indicate who would like one of these kits, and the response was overwhelming, so payment was asked for, and no less than 41 Buffer kits were ordered at a sum of \$1400.00. The Group has since ordered these kits, plus only a couple of extras, and we intend to have a special interest meeting devoted to building the buffer - for those of you who wish to build your own buffer, you will need a SMALL soldering iron of

about 20 watts, and some fine (about 1mm thick) resin-cored solder.

The BBQ has not yet taken place (at time of writing), but I certainly hope that those of you who did go, had a good time. The response at the Botany meeting was rather poor, when the members were asked for a show of hands for those intending to attend the BBQ - I hope that the turn-up will be a LOT better than that indicated. (ED. NOTE - The eight members and families that turned up did enjoy themselves thoroughly, the venue was great for the kids and oldies alike, too bad a few more didn't attend).

By the way - congratulations to Ian Sergeant for the very topical game he left on the Bulletin Board. He says he has copyrighted it so that everyone in SYDTRUG will eventually get a pirate copy (ha ha!). The game is called SYDADV/CMD, and is an adventure game in the classic (Scott Adams) mould, but with a VERY up-to-date theme of a SYDTRUG meeting being raided by the Federal police, looking for "artificial apples". Get it if you can - it's quite good.

Our Treasurer is not too well at the moment, and I am doing most of his duties as well as my own - the workload is rather high. I'm having to battle with VISICALC for the Treasurer's bookkeeping (at which I'm not very good), and PROFILE 1+ for the membership database, LAZYWRITER for this column, and SuperSCRIPIT for other correspondence. DOTWRITER is also used for letterheads and information sheets. One has to get used to a number of programmes for these duties.

The financial report for January follows:-

Income to 31/01/86		Expenditure to 31/01/86	
Items for sale	342.88	Newsletter postage	59.55
New members	38.00	Printing	215.00
New BBS	10.00	Sundries	40.00
		Sefton rent	40.00
-----		-----	
Total income	\$382.88	Total expenditure	\$354.55
Excess for month	\$27.45		

I have received (admittedly a fair few weeks ago), some correspondence from a Tony Briggs in New Zealand. He writes with some queries about hardware mods to his SYSTEM88, along with some other questions. He also enclosed some fine public domain software written by himself and his brother for distribution amongst the members only, and a "blurb" for some software which is being sold by his brother. The PD software is available for copying at Botany any meeting, or can be ordered by mail (if you send a disk and indicate the format, or else pay \$3.00 for one of our disks). The software just about fills two sides of a 35 track, single density disk, and is described briefly below.

AID3ZAP/CMD	Allows AIDS+ v1 and NEWDOS to read double sided disks.
COPYPACH/CMD	Modifies COPYCAT v1.6 to operate double sided. Model 1 only.
DIRZAP/CMD	Patch for DIRCAT/CMD supplied with AIDS+, to allow double sided drives.
DISKMAP3/CMD	Maps NEWDOS88 disks, showing allocation of every granule. Will analyse most NEWDOS disks, setting the PDRIIVE accordingly. Model 1 & 3. An excellent utility for NEWDOS.
DUPACH/CMD	Modifies DISKUP/CMD v3.1 to operate double sided. Model 1 only.
PASSWORD/CMD	Encodes and decodes passwords. Faster than SU3+.
PDRAW/CMD	Modified from a program in 88-MICRO July 1984, to use a Gemini Star printer.
PDRAWINS/PCL	Instructions for the above.
*PENCIL09/SYS	Interfaces PENCIL with DOTWRITER.
*PENCIL10/SYS	Cursor driven directory for PENCIL.
PEN0910/DOC	Documentation for above two programs.
PRINTGRF/BAS	Speeds up printing of graphics from BASIC data statements.
PRINTGRF/PCL	Instructions for the above.
CLOCKS/BAS	A silly (author's own words) program, developed for fun only.
CONVERT/CMD	Modified from 88MICRO - converts U/C text to UC/LC in BASIC programs.
DISKLABL/CMD	Model 1 program to print disk labels with GEMINI 10X or EPSON-type printers only.

LOADER3/CMD) for use with DISKS
 LOADERE/CMD) for use with EXATRON stringy floppy -
 these programs will read multiple origin
 machine code tape or disk programs with ANY
 number of origins.
 LOADIN3/DOC Doco's for the above.
 PDRIIVER/CMD)
 PDTEXT/PCL)Not too sure about these.
 GAS/BAS Keep tabs on your fuel consumption and
 costs.
 COMPARTS/BAS Education aid about computer parts.
 LOGTRAIN/BAS Logic trainer program.
 VISISORT/BAS SORTing technique demonstration.
 * ---->> These two programs are being submitted to
 80-MICRO for possible publishing, so the
 MUST NOT be circulated outside the group
 yet.

There is also another file called PGMBLURB/PCL, which is a "blurb" about some software being sold by Tony's brother. There are some modifications for OS-80 (MICRODOS), but that is all I can see with SUPERZAP, as the file is corrupted, so I will endeavour to get another copy.

That is about all for this month - see you at the Botany meeting.

GENERAL MEETING REPORT

by Jim Whittaker

Chaired by Ted Romer on the 8th February '86

Well, the christmas rush over, the SYDTRUG rush began. My apologies to any who felt a little put off by the lack of personal attention but Darryl and myself just didn't have time to scratch ourselves. Someone was in a rush and asked me to take their money for the printer buffer and then the flood gates opened. Luckily, we were able to take a rest when the main meeting started.

First order of business is the Hard Disk Drive. We are still waiting for TANDY to pull their collective fingers out or we may have to withdraw our earlier congratulations.

We are running short of ink for the ribbon re-inker but one of our more illustrious members has offered to take a sample (specimin) and get some made up. Thanks Con.

VIATEL - how they ever do business I will never know. They have not responded to any of our requests for information and advertisements. Maybe they will get the wooden spoon award.

The BBQ is a goer and we will be supplying some sausages and some soft drinks. Gary will be getting there early to pitch camp so roll up and don't forget to "Bring your family with you" and have a good time. Thanks Gary. (ED.NOTE- see comments in Darryl's column).

I am now taking money for the Printer Buffers. I have bought my own along so you all can see what the "GUTZ BITZ" look like. Mine cost me a total of \$100.00 to complete but I used some stuff from the "Junk Box", so you could expect to spend a maximum of \$150.00 if you had to buy everything from scratch.

Quotes for the 256k RAMs have varied widely, the lowest being \$6.50 each (you need up to 8). If anybody knows of cheaper RAM then please let me know. 360 RAM chips is a significant order. If all goes well, we will have this as a SIG project in MARCH. Checkout Robert Gareb's article in this issue.

Insert :- All 45 buffers have been paid for and we are awaiting delivery. Hopefully they won't be long.

Disk Mailers. If anybody is interested, let me know, we have unearthed 2 good types. The first is a MEMOREX mailer, made of compressed cardboard with sealed edges, a static free liner and a sealable flap, for about \$2.00 each. The other is a 2 piece corrugated cardboard affair which is stiffer, cheaper (at 0.75) but more bulky and fiddly.

The SYDTRUG committee, in its infinite wisdom, has decided to purchase a 128k MODEL IV. The long story is that I bought the machine for myself at the great price of \$900.00 and as always, two weeks later I was offered a MODEL IVP. Not being able to afford both, I offered the IV to the club at the same price of \$900.00 (Actually I had it sold to someone else but they had to pull out at the last minute). Mike Cooper now has it and will undoubtedly fill you all in as to its progress.

In its first week of operation, Mike told me that it has "Flakey memory", "Flakey disk controller", "Flakey screen" and "Flakey Disks". Later, he confided that in actual fact, there may be a slight chance that there is one or two "LOGIC" errors in his code. Heaven forbid, my code works first time every time.

Someone stood up and asked if there was any interest in a bulk purchase of modems. This started some discussion as to what was wanted in a modem, eg. auto answer, 300 or 1200/75 baud etc. It appeared that only 2 or 3 people were interested. We did at one stage get cheaper modem kits from AVTEK if we bought in lots of 4. As always it is up to you. Get 4 people together with money in hand and I'll do the rest.

Darryl gave us a quickie on what we have received in the mail. In response to a members request, he asked what everybody wanted in the way of a mail report. The general consensus appeared to be a brief report on any items of interest.

The Public Domain Software has been sold out this week and we are trying to get some extra copies made right now. Bill McDougal has been cornered and is doing the copying. I don't know if you all know but Bill copies up all the PD software that we sell. He spends a lot of hours behind the scenes, so thanks from us all Bill.

Nash Pilling popped up and asked if anybody knows of an OMNI READER type service that can read typed documents and make disk files of them. If anybody can help then please let him or me know as he has a pile of urgent documents that need transferring to disk.

Someone else asked about the connection of S100 compatible peripherals to the System 80 S100 bus. The short answer appeared to be that the System 80 S100 bus had a timing that is not exactly standard and as such causes problems for all.

Other members later asked about having a CPM SIG meeting. Anyone who has the knowledge and would like to help out on a CPM day see me.

Someone else wanted to know how to copy the PD Software onto their own disks when they have a Model III but only 1 drive and NEWDOS/80.

A late report on the BBQ is that the attendance has been poor but those that did make the effort were well rewarded. Because of the poor roll up, the drawing of the Doubler raffle has been postponed until the third meeting of the month at SEFTON.

Well, thats about it for this month. I make no apologies for all the spelling mistakes, cause I can't spell for nutz, but I hope you can all get the drift of what is happening at the meetings and a general feel for what is happening at your club. Also remember that if I am busy, I may bite but generally, I don't so come up and make yourself known to me. --- Regards JIMBO.

PRINTER BUFFER REVIEW

- Kit by Don McKenzie -

Review by Robert Gareb

For those of you who are not aware what a printer buffer is, read Jim Whittaker's Vice Presidents report in the last newsletter. Basically, a printer buffer is a completely separate 'black box' that sits between the computer and the printer. It is a complete stand alone microprocessor based unit (in our case Z80A CPU, 8255, 2716 EPROM, 4164 or 41256 RAM chips. RAM total 8K up to 256K Bytes.)

For a simple analogy of what it does, take as an example a dictaphone. You dictate at a fast speed what you want typed, then it is given to your secretary to type. She will not be able to keep up with the speed you dictated at, so when she listens to as much as she can remember at once, she presses the pause button. This is the equivalent of the busy signal to the computer. When free again, she releases the pause button. In this case the computer gets a not busy.

In the January meeting there was a lot of interest shown in the printer buffers, which cost hundreds of dollars to buy. However there is a cheap alternative. Build one yourself! It's not that hard thanks to Don McKenzie. For those of you who do not know Don, he's a hacker from way back in the good old Model 1 days. (Yes those good old days, when hardly anybody had a standard machine!) Don has designed many mods including Donmon, a brilliant monitor resident in the 2K unused portion of memory. All his products are sold with full documentation and backup, and I have no hesitation in supporting him 100%.

Don has developed the buffer, and as always he only charges a very reasonable charge for the 'basic essentials', the rest you do yourself. I have personally got the early version, the 64K version (it can be 8, 16, 32, or 64K depending on how much memory you install), and the new 256K version. Yes, I know you are saying that you have software spoolers, etc, but I find this so much easier - you leave it in line with the printer and it does not need initialization. Now for the other brilliant features - :

- 1) Reset switch for any problems or to clear the buffer if you realised you made an error eg forgot to set right justification.
- 2) Test switch so if it doesn't work, you can see if it's the buffer or the computer/cable causing the problem.
- 3) Copy switch - repeat the last print-out since your last reset.
- 4) Hex output - a great feature for my personal application. Have you ever wondered what those unprintable characters were from an erroneous text file or bulletin board? Find out in hex print mode, and filter them out!

OK. I know you are all asking - how much?

Normal price of the bare board, EPROM (with Don's software) and crystal is \$42, which also now includes an extra board to simplify connecting the ribbon cable to the main board at no additional cost.

However, I have arranged a bulk buy by the group, so there is a special price for club members. We have 48 boards at \$35 - a saving of \$7. This will be first come, first served, however if the demand is too great, we will organise another bulk purchase. Country readers will be accommodated within this offer as many don't make it to our meetings.

NOTE - At the time of ordering you MUST specify 64K or 256K version of EPROM.

The total cost of all other parts depends on memory size, but even 256K should be under \$100 total. At present we will try to find a cheap source of 41256's, but the going price is approx. \$6 each (8 needed), or \$3 each for 4164's.

What else is included in the cost? Full documentation, which gives an overlay of the board, component list, step by step assembly instructions. It is geared at the level of a novice without too much experience. Those with practical experience will say - That's obvious, but you have to remember that we all started somewhere.

The documentation even includes fault diagnostics in case (heaven forbid) it doesn't work first go.

Don also has a serial output version coming out soon, a brochure is included.

Don't say that's still not enough. OK - a new feature is software pause - send backslash twice (hex 5C or decimal 92) and it stops till you either fill the buffer or press/release the copy button.

That's it - by the time you read this, some of you will have paid for your unit, as I will organise money collection at the February meeting. If you haven't paid there, please send a cheque to the club clearly stating your name, address and whether you want the EPROM for 64K version or 256K version. Please add \$2 to cover postage, ie total \$34.

LETTER TO THE EDITOR

17th, February 1986.
To the Committee and Members of SYDTRUG,

I would like to draw attention to what I consider an important matter regarding correspondence to SYDTRUG members.

As the membership secretary for 84/85, my duties included assisting in the distribution of the newsletter, maintaining the membership list and collecting correspondence which I was instructed to open, enabling me to list all renewal's and new memberships. The correspondence would then be passed onto the club's secretary. On many occasions, items of correspondence contained attractive offers of software items which were then passed on to SYDTRUG members. More recently, I was, unfortunately, put in the uncomfortable position of having to draw a committee members attention to the fact that some of the correspondence was not in fact being passed on to the committee or club members.

I believe that this point has been heatedly discussed at committee level and that it has now been agreed that SYDTRUG members should in future be informed of all correspondence items.

It did strike me as quite odd, that the week following this meeting, the amount of mail I collected had dwindled considerably. On bringing this to the attention of the secretary I was informed that another source would be responsible to collect the mail, and that my services were no longer required and that I was to return my mail box key at the first opportune time. This decision I believe has not been discussed at committee level.

It is not the intent of this letter to behave as a dog with a bone, but rather, to draw other members attention to the fact that the SYDTRUG BY-LAWS, has no firm policy in procedures regarding the tabling and distribution to the members of, correspondence items and that, at present it appears to me that it is solely at the discretion of the secretary as to whether a correspondence item is labelled for members or not.

In view of the above, I would like to know if other SYDTRUG members share my personal opinion (and those of others) in this being an unsatisfactory arrangement within a club of our magnitude. I believe that it is in the best interests of SYDTRUG members that a firm BY-LAW regarding the tabling and distribution of all correspondence be immediately included into the BY-LAWS.

In closing, I cannot help but wonder what access to club offers (incidentally that our membership fees entitle us to) we may-be being denied.

Algis Bizys. A887

THE COMM LINE

by Michael Cooper (SYDTRUG Sysop)

Well March already, doesn't the time fly. This month I will be covering the introduction to the set of I/O drivers used on the BBS and what they are designed to achieve, and how they are controlled. Next episode I will move onto the code itself and cover that over a couple of issues or so. The final assembly language code will be left on the BBS when the series is complete. To begin however, is the usual BBS report.

CLUB-80 BBS NEWS

A milestone has been reached with the system topping 10,000 callers. There have been several other changes take place since the new year, and more yet to come.

The SYDTRUG committee have approved funds to purchase a Model 4 with 128K and a Hard Disk (5 Meg) upon which to run the SYDTRUG CLUB-80 BBS. Formerly the system operated on a 48K Model 3 with 80 track floppies, and at the moment it's running on the new Model 4, but alas the Hard Disk is still with Tandy in Perth getting fixed. We hope to see that in the near future.

I have also been busy rewriting the communications supervisor in assembler to enable faster and more sophisticated operation. Currently I have written all the primitive (low level) routines and am running them now. I intend to publish the routines in these pages over the next several months starting this month, but more of that later.

The new code offers the following benefits; better XON/XOFF flow control, instantaneous abort of listing using (Esc) or (Ctl-C), and faster keyboard input rate. The primitives have all the hooks incorporated to allow the use of a proper terminal driver featuring several different output formats (40x8, 40x16, 40x20, 40x24, 64x16, 80x24). As well I expect to eventually permit ASCII file transfers using XON/XOFF flow control (to be announced).

Moving right along - Rob Gareb reported problems using XMODEM (Christensen) File Transfer mode in the BBS Library if using an IBM-PC (or clone). In response I tried it out for myself, firstly with a Macintosh, and encountered no problems. Next I tried using CPM (Montezuma) on my Model 4P with the MODEM7 program. This produced a similar error to that experienced by Rob (Timeout on Block 16 at the senders end, and a long delay before the receiver's end timed out, the nett result of which was loss of synch between each end). Further investigation and a thorough re-reading of Ward Christensen's protocol document led me to some timing problems in the file transfer software. I have made the necessary changes and have hopefully corrected the errors. During the testing phase, Library transfers were successfully carried out with the following computers and terminal software.

TRS-80 Model 3 using LDOS and running MODEM80
 TRS-80 Model 4P using TRSDOS 6.2 and running MODEM804
 TRS-80 Model 4P using CPM (Montezuma) and running MODEM7
 Macintosh using Finder 4.1 and running MacTerminal

If you are using any other types of machine/software combination then I can't guarantee anything, but try leaving (Uploading) a file to the Library and see if all goes OK.

BUILDING YOUR OWN BBS (Part One)

Over the next few issues I intend to cover the design and implementation of a versatile set of machine language drivers suitable for use with BASIC programs. These will permit the conversion of practically any BASIC program in your collection, or perhaps one you write yourself, to run remotely on a dial-up TRS-80. The only limitations are that you have a Model 3 or Model 4/4P and use LDOS as the operating system. If you have a Model 1 or System-80 then all is not lost, just a few changes to the code supplied will effect the necessary conversion. This however, will have to be a job for one of you keen readers out there, as I don't have old "Bertha" (my Model 1) working anymore and can't test out the changes.

1. System Considerations

Before a byte of code is written, a few considerations on the needs of dial-up system software need to be taken. Firstly there is the issue of SECURITY against unwanted callers tampering with your software, next there is the need for some Communications Monitoring to ensure the system can handle being used remotely by MODEM, and lastly there is the need for some sort of recovery mechanism should your system suffer a crash or runtime error.

The routines I will publish address all the aspects mentioned above, and several others that you may not need to consider. For example on the BBS there is a need to distinguish between callers who are using a TRS-80 Model 1/3/4/4P type system, and those who may be using another (any) type of micro or terminal. TRS-80 callers require no linefeeds after a carriage return, and you can clear the screen by sending a CHR\$(31), you can backspace by

sending a CHR\$(8), and you can display graphics characters. For others however, a (LF) must be sent after a (CR), each will have a different way of clearing the screen, to backspace will require sending a 3 byte string of (BS) (SPACE) (BS) to ensure the character is deleted on the remote screen, and the graphics (if any) will be different. On CLUB-80 I handle this by having two basic terminal modes; viz TRS-80 or GENERIC. In your particular case, you may only need the TRS-80 mode to be active at all times.

Communications monitoring needs are fundamentally concerned with ensuring that there is Carrier Detect from the remote end. Other monitoring should be done on the keyboard to prevent callers from leaving the system sitting idle for hours on end, and also monitoring can be done to ensure that the system is always running an application program and is not at LDOS or BASIC Ready prompt. As well you need to provide some remote flow control (Using XON and XOFF characters to start and stop the output) and some sort of (Break) key to abort listings etc.

Input driver features should allow you to force all input into Uppercase, suppress the Echo of characters to the remote end, and control the BELL (or Buzzer) to prompt for input at the right time.

To handle all the above mentioned goals, the BBS software needs to be able to be controlled from within BASIC or Assembler programs. The simplest way to do this is to use a series of FLAGS and preset values to initialise the control routines. I have used the top 16 bytes of memory (FFF0 - FFFF) to hold a parameter block that controls the action of all the BBS Driver routines. The specification is as follows.

2. Parameter Block Specification

ADDR	GLOBAL LABEL	DESCRIPTION
FFF0H	QSYSFLG	Communications Flags Bit 0 = Failsafe (Set=Disabled) Bit 1 = Carrier Check (Set=Disabled) Bit 2 = TRS-80/Generic (Set=Generic) Bit 3 = Timeout Check (Set=Enabled) Bit 4 = Sysop Breakin (Set=Enabled) Bit 5 - 7 are RESERVED for later use
FFF1H	QINPFLG	Input Driver Flags Bit 0 = Line Terminator (Set=ETX) Bit 1 = Force U/case (Set=Enabled) Bit 2 = ECHO (Set=Disabled) Bit 3 = BELL (Set=Enabled) Bit 4 = Input Mode (Set=File) Bit 5 - 7 are RESERVED for later use
FFF2H	QOUTFLG	Output Driver Flags Bit 0 = Paged Output (Set=Enabled) Bit 1 - 7 are RESERVED for later use
FFF3H	QUSRFLG	User Flags Bit 0 - 7 are for your own use
FFF4H	QOUTLIN	Terminal Lines (8-32) Init=16
FFF5H	QOUTWID	Terminal Width (40-80) Init=64
FFF6H	QINPLEN	Input Line Length (60-255) Init=60
FFF7H	QTIMLEN	Timeout delay in mins (1-5) Init=3
FFF8-9H	QTIMCTR	Timeout Counter for 1 min Init=1500H
FFFA-BH	QCDLY	Carrier pause for .25 sec Init=3000H
FFFCH	Q555555	RESERVED for later use
FFFDH	Q666666	RESERVED for later use
FFFEH	QCDCNTR	No. of Carrier Losses (0-20) Init=0
FFFFH	QXFERR	XFER80/CMD Error Status Init=0 0 = Transfer went OK 1 = DOS Opening ERROR 2 = DOS I/O ERROR 3 = MODEM Link Error (Bad line etc) 4 = Duplicate Filename (On Rcv only)

Well that's the end of part one, next month I will publish the I/O Primitive routines that use the above parameter block, and as well, the Communications Primitives routines (including the Timeout routines). Until then, happy MODEMING.

TRSDOS 6.2.x SPOOLER

Printed from "Northern Bytes", Vol. 6, No. 3

(Jack Decker's Note: This file was downloaded from CompuServe by Bob Grommes of the Christian Computer Users Association, who passed it along to us. I assume that it was authored by someone at Logical Systems, Inc)

It has come to our attention that some users are having difficulty using the system spooler on TRSDOS 6.2.0. These problems fall into two categories:

- 1) The spooler won't run at all in the back bank (extra memory).
- 2) The spooler runs fine at the DOS level but blows up in BASIC.

Problem #1:

This is generally the result of a hardware difficulty. We have two machines here which exhibit this problem. Once the spooler is installed, the machine may lock immediately, or run until printer output is attempted.

It appears that the CPU can't reliably execute programs that reside in the extra 64K RAM. Both are very early release machines with one or two wait states, and the problem may be related to the PAL chip not inserting M1 wait states for memory accesses in the alternate banks. Solution: get your CPU board swapped out. It is likely that Radio Shack will charge you for this.

Problem #2:

This one is a software problem. Contrary to the Radio Shack 6.2 manual, back bank spooling was supposed to work with BASIC. It turns out, however, that there is a conflict between BASIC's use of vectored (break) and the spooler. The following patch will allow the use of the spooler in the back bank from BASIC:

```
PATCH BASIC/CMD.BASIC (052,83=00 00:F32,83=06 7D)
```

It may be possible to prevent this conflict in a future release of TRSDOS 6.

In either case, spooling by using a disk file contained on a memDISK in the back bank should work.

MODEL 4/4P ROM CHANGES

by Jack Decker

Reprinted from "Northern Bytes", Vol 6, No. 2

This article is a not-too-well organized collection of data on the differences between the "old" and "new" versions of the Model 4 ROM/4P ROM image (that is used when the computer is run in the "Model III mode"). The "new" version was first introduced as the Model 4P ROM image (the "MODEL4/III" file supplied with the 4P), then copied back to the "un-portable" Model 4 at about the same time that Radio Shack relocated the arrow keys and started putting green CRTs in the 4 and 4P. The "new" Model 4 ROM and the "new" 4P ROM image have a few minor differences, mostly due to hardware. sk The major changes are as follows:

- 1) Revisions of the ROM in locations 0000H-2FFFH. Although Radio Shack has released several versions of the portion of ROM from 3000H-37FFH, this is the first revision of code in the portion of memory below 3000H since the very early days of the Model III. A revised printer driver with some added character translation capabilities, but this will cause no end of grief for unwary programmers, particularly those that have programs that output graphics to the printer.
- 3) A completely new routine that allows a 4P to be booted off the RS-232 interface.

I'll cover each section individually, but first the credits: Thanks to Nate Salisbury and especially to John Hallgren of Clearwater, Florida for providing the information below. Most of what you see here was extracted from John's handwritten notes on the new ROM, of which he very graciously sent me a copy. Additions, corrections, and clarifications to this article are most welcome. That said, we begin...

A. ROM CHANGES AND ADDITIONS

0001H-0004H Changed in the 4P ROM image only - not on the standard Model 4. Old: XOR A followed by JP 3015H (jump to reset/power up routine). New: LD A,1 followed by OUT(9CH),A (on the 4P port 9CH is the Boot ROM switch - this code disables RAM/enables ROM boot).

0006H-0007H Changed in the 4P ROM image only - not on the standard Model 4. Old: Part of unused JP 4000H instruction. New: Part of JP 3015H instruction (moved from 0002H, jumps to reset/power-up routine).

0043H-0045H Old: Unused RET instruction followed by two zero bytes. New: JP NC instruction at 3790H (part of line printer driver - see commented disassembly).

0063H-0065H Old: Leftover code from the Model I time delay routine (JR NZ,60H followed by RET). New: JP 041FH instruction, accessed from JP C instruction at 3795H (part of line printer driver - see commented disassembly).

006CH-0074H Old: LD DE,4210H followed by JR 001BH (Start of I/O re-router routine which has been deleted in the Model 4) followed by NOP, and finally followed by an unused JP 06CCH instruction (left over from the Model I ROM code). New: RET instruction (disables I/O re-router) followed by LD BC,1A18H and JP 19AEH instructions and two unused zero bytes. The LD BC,1A18H and JP 19AEH instructions comprise the old Model I "warm re-entry to BASIC" code that was originally found at 06CCH on the Model I, and was erroneously deleted from the Model III. It is used by the BASIC "SYSTEM" command when the BREAK key is depressed (accessed from 02C3H).

00FFH-0101H Old: JP 37EBH (part of the routine that prints "Radio Shack Model III Basic" and copyright message. In the Model III, the first instruction at 37EBH was CALL 021BH). New: CALL 021BH (done here instead of at 37EBH).

0103H-0104H Old: Unused JP 1A19H (left over from the Model I). New: JP 01E6H (on the Model 4) or JP 01E7H (on the Model 4P) (jump to remainder of routine to print copyright message).

01D9H-01F7H In the Model I, this block of memory contained the code to output one bit to the cassette. In the "old" ROM, this area contained the screen print routine (plus leftover garbage from the Mod I routine at 01F5H-01F7H). In the "new" ROM, this area contains a vector to the new screen print routine (a JP 3027H which in turn jumps to 37A5H), the printer status test that was formerly located at 0440H-044AH in the "old" ROM (now moved to 01DCH-01E6H in the "new" ROM), code to display the Tandy copyright message and put the BASIC TIME\$ vector into place at 4176H (this code was formerly located at 37EEH-37FCH and has been moved to 01E7H-01F5H in the "new" ROM), and unused zero bytes at 01F6H-01F7H. Note that a POP AF instruction is found at 01E5H in the 4P ROM image, but this instruction is deleted in the "new" Mod 4 ROM, and the instructions between there and 01E9H have been moved forward one byte. 01E9H contains a zero byte in the "new" Model 4 ROM only.

- 0210H-0211H Old: Leftover garbage from the Model I "turn off cassette" routine. New: Two unused zero bytes.
- 0232H-0234H Old: Leftover garbage from the cassette "blink asterisk" routine. New: Unused zero bytes.
- 02A8H-02A8H Old: Unused RET instruction. New: Unused zero byte.
- 02B6H-02B6H Old: part of LD SP,4288H instruction. New: Part of LD SP,42E8H instruction. This code is used by the BASIC "SYSTEM" command.
- 02C4H-02C5H Old: Part of JP 06CCH instruction. New: Part of JP 086DH instruction. This code is used by the BASIC "SYSTEM" command and fixes a bug. When the BREAK key was pressed in response to the "SYSTEM" command prompt, the "old" ROM would jump to 06CCH which was a "warm re-entry" to BASIC on the Model I, but is part of the LIST/LLIST code on the Model III ROM. The "new" ROM fixes this by jumping to the code that used to be found at 06CCH on the Model I, but is located at 086DH in the "new" ROM.
- 03CAH-03CAH Part of the printer driver - see detailed disassembly.
- 03CEH-03CEH Part of the printer driver - see detailed disassembly.
- 03D7H-03D8H Part of the printer driver - see detailed disassembly.
- 03E3H-03F9H Part of the printer driver - see detailed disassembly.
- 03FBH-041EH Part of the printer driver - see detailed disassembly.
- 0428H-0449H Part of the printer driver - see detailed disassembly.
- 0469H-046AH Old: Leftover garbage from the Model I video driver routine. New: Two unused zero bytes.
- 0501H-0506H This was a "printer ready" test in the Model I, which was destroyed (for no apparent reason) in the "old" Model III ROM by changing the first three bytes to the characters "RON". In the "new" Model III ROM, the "Cass ?" string literal that was formerly at 37F6H-37FDH has been moved here.
- 18C2H-18C2H In the Model I and in the Models III & 4 using the "old" ROM, memory location 48B9H was used to flag DATA statements while encoding BASIC lines. In the "new" ROM machines memory location 489FH (a previously unused location) is used for this purpose. Thus this byte is part of a reference to location 48B8H that has been changed to 489FH.
- 18DDH-18DDH Same as above (18C2H). 1C68H-1C68H Same as above (18C2H).
- 2409H-2409H Same as above (18C2H) ON THE MODEL 4P ONLY. In the "new" Model 4 ROM, this change and the next (2452H) were restored to the previous value of B0H. Why? Only Tandy knows! John Hallgren speculates that the reason may have something to do with the fact that these two are part of arithmetic routines, and the 48B8H location is used to store tokens (JG's 'Microsoft Basic Decoded & Other Mysteries' says that the "arith token of last operand (the one to be performed)" is stored here).
- 2452H-2452H Same as above (2409H).
- 2FFBH-2FFFH Old: Unused garbage. New: Unused zero bytes.
- 3815H Modified vector to reset/power-up routine (Old: JP 3481H; New: JP 3426H).
- 3824H Modified vector to the keyboard driver routine (Old: JP 338EH; New: JP 3106H).
- 3827H Modified vector - in the original Model III ROM, this was a jump to the I/O re-router routine at 3739H, in the original Model 4 ROM it contained a RET instruction followed by two NOPs, and in the "new" ROM it contains a jump to the screen print routine at 37A5H.
- 382DH Modified vector to part of the LIST command (Old: JP 37A4H; New: JP 375CH).
- 3838H Modified vector to the BASIC TIME\$ function routine (Old: JP 37C2H; New: JP 37EAH).
- 383DH Modified vector to the Non-Maskable Interrupt handler routine (Old: JP 34CEH; New: JP 358BH).
- 3842H Modified vector to the \$SETCAS routine that prompts the user to set the cassette baud rate (Old: JP 318BH; New: JP 33FFH).
- 3845H NEW vector (JP 378DH) used by the line printer driver to translate a character in the range C0H-DFH using the table pointed to by an address stored at 4226H-4221H. This vector is referenced by a CALL at 03F1H if the byte stored at 41FBH = 1.
- 3848H NEW vector (JP 377AH) used by line printer driver to adjust counters at the start of each new line.
- 384BH NEW vector (JP 3179H) used to "boot" the computer from the RS-232! There are no references to 384BH in the ROM, however, another reference to 3179H is found at 3518H.

OTHER CODE CHANGES IN THE ROM ABOVE 3880H:

- 384EH-3885H Keyboard lookup table that formerly started at 3845H has been moved forward nine bytes in the new ROM (locations 3869H through 386DH contain unused zero bytes).
- 3886H-388DH Part of RS-232 boot routine - see detailed disassembly.
- 388EH-38C5H Keyboard lookup table that formerly started at 3885H has been moved forward nine bytes in the new ROM (locations 38A9H through 38AEH contain unused zero bytes).
- 38C6H-38C8H Part of RS-232 boot routine - see detailed disassembly.
- 38C9H-38CDH A NOP instruction followed by the Floppy Disk Controller time delay routine formerly found at 3518H (in the Model III) or 37E1H (in the "old" Model 4 ROM).
- 38CEH-3185H Keyboard lookup table that formerly started at 38C5H has been moved forward nine bytes in the new ROM (locations 38E9H through 38EDH contain unused zero bytes).
- 3106H-3178H Part of the keyboard driver routine equivalent to code found at 338EH-3408H in the "old" Model 4 ROM.
- 3179H-31A4H Part of RS-232 boot routine - see detailed disassembly.
- 338EH-33F8H Part of keyboard driver routine equivalent to code found at 3739H-37A3H in the "old" Model 4 ROM.
- 33F9H-3425H Code moved from 3185H-3131H in the "old" Model 4 ROM, includes the \$SETCAS routine.
- 3426H-3516H Equivalent of code found at 3481H-34D9H in the "old" Model 4 ROM (bootstrap routine). Note that the number of retries for a diskette (value at 3487H in the "new" ROM) has been doubled (from five to ten).

NEW AND MODIFIED JUMP VECTORS:

- 3812H Modified vector to disk bootstrap routine (Old: JP 3461H; New: JP 3486H).

3517H-3527H Part of RS-232 boot routine - see detailed disassembly.

3739H-375BH Part of keyboard driver routine equivalent to code found at 34DAH-34FCH in "old" Model 4 ROM.

375CH-3779H Part of BASIC "LIST" command routine moved from 37A4H-37C1H in the "old" ROM.

377AH-37A4H Part of printer driver - see detailed disassembly.

37A5H-37C6H Screen print routine moved from 34FDH-351EH in "old" ROM.

37C7H-37C8H Part of keyboard driver routine equivalent to code found at 313BH-313FH in the "old" Model 4 ROM.

37CCH-37D4H Part of keyboard driver routine equivalent to code found at 37D8H-37E0H in the "old" Model 4 ROM.

37D5H-37E4H Part of RS-232 boot routine - see detailed disassembly.

37E4H-37FCH BASIC TIMES routine moved from 37C2H-37D7H in "old" ROM.

3. THE NEW PRINTER DRIVER CODE:

```

00100      ORG      0043H
00110 ;xxx CHARACTER >=0E0H & FLAG @ (41FBH)=1 xxx
00120 00043H JP      00434H
00130 ;
00140      ORG      0063H
00150 ;xxx VECTOR TO "NORMAL" CHAR PRINT ROUTINE xxx
00160 00063H JP      0041FH
00170 ;
00180      ORG      01DCH
00190 001DCH CALL    0044BH ;CHECK PRINTER
00200      RET
00210      CALL    0029DH ;CHECK <BREAK>
00220      JR      Z,001DCH ;LOOP IF NOT PRESSED
00230      POP     AF
00240      RET
00250 ;
00260      ORG      020DH
00270 0020DH LD      A,(0043H) ;GET <BREAK> KEY ROW
00280      AND     04H
00290      RET
00300 ;
00310 ;xxx ACTUAL START OF PRINTER DRIVER xxx
00320      ORG      03C2H
00330      LD      A,C
00340      CP      20H
00350      JR      NC,003E7H ;JUMP IF NOT CONTROL
00360 ;xxx CHAR IS CONTROL <(20H) xxx
00370      CP      00H
00380      JR      Z,00414H ;JUMP IF SO
00390      CP      0CH
00400      JR      NZ,0041DH ;JUMP IF NOT
00410 ;xxx CHAR IS FORM FEED xxx
00420      LD      A,(IX+3) ;GET # LINES ON PAGE
00430      SUB     (IX+4) ;SUBTRACT CURRENT LINE #
00440      LD      B,A
00450 003D6H CALL    001DCH ;WAIT FOR PRINTER READY
00460      LD      A,0AH
00470      OUT     (0F8H),A ;OUTPUT IT
00480      DJNZ    003D6H ;LOOP UNTIL PAGE FEED
00490      LD      (IX+5),0 ;# CHARS PRINTED = 0
00500      LD      (IX+4),1 ;RESET CURRENT LINE COUNT
00510      JR      00448H ;GO TO EXIT ROUTINE
00520 ;xxx CHAR IS NOT CONTROL <(20H) xxx
00530 003E7H LD      A,(41FBH) ;GET FLAG
00540      OR      A
00550      JR      Z,003F6H ;IS IT ZERO ?
00560      CP      01H
00570      JP      Z,00405H ;GO IF SO
00580      JR      0041FH ;GO IF FLAG NOT 0 OR 1
00590 ;xxx CHAR NOT CONTROL & FLAG @ (41FBH)=0 xxx
00600 003F6H LD      A,(41FCH) ;GET FLAG
00610      OR      A
00620      JR      NZ,0040AH ;GO IF NOT
00630 ;xx CHAR NOT CTRL & FLAGS @ (41FBH)&(41FCH)=0 xx
00640      LD      A,C
00650      CP      0A0H
00660      JR      C,0041FH ;GO IF SO
00670      CP      0C0H ;IS CHAR <0C0H ?

```

```

00680      JR      NC,0040FH ;GO IF NOT
00690 ;CHAR IN RANGE A0H - 8FH
00700      ADD     A,40H ;MAKE CHAR E0H-FFH
00710      LD      C,A ;SAVE IT
00720      JR      0041FH ;GO OUTPUT IT
00730 ;xx CHAR NOT CTRL, FLAG @ (41FBH)=0, @ (41BCH)<0
00740 0040AH LD      A,C ;GET CHAR
00750      CP      0C0H ;IS CHAR <0C0H ?
00760      JR      C,0041FH ;GO IF SO
00770 ;CHAR IN RANGE C0H - FFH
00780 0040FH SUB     20H ;MAKE CHAR A0H - DFH
00790      LD      C,A ;SAVE IT
00800      JR      0041FH ;GO OUTPUT IT
00810 ;xxx CHAR IS CARRIAGE RETURN xxx
00820 00414H LD      A,(IX+05H) ; # CHARS PRINTED IN LINE
00830      OR      A ;ANY CHARS ON THIS LINE ?
00840      JR      NZ,00434H ;GO IF SO
00850      LD      A,0AH ;CONVERT <CR> TO <LF>
00860      LD      C,A ;STORE LINEFEED CHAR
00870 ;xx JUMP HERE IF CHAR CTRL BUT NOT <CR> OR <LF> xx
00880 0041DH JR      00434H
00890 ;xxx THIS ROUTINE PRINTS "NORMAL" CHARS xxx
00900 0041FH LD      A,(IX+06H) ;GET MAXIMUM LINE LENGTH
00910      INC     A ;IF NO MAXIMUM, NOW A=0
00920      JR      Z,00434H ;GO IF NO MAXIMUM LENGTH
00930      CP      (IX+05H) ;MAXIMUM LENGTH REACHED ?
00940      JR      NC,00434H ;GO IF NOT
00950      CALL    001DCH ;WAIT FOR PRINTER READY
00960      LD      A,0DH ;CARRIAGE RETURN IN REG A
00970      OUT     (0F8H),A ;OUTPUT IT
00980      CALL    00046H ;ADJUST COUNTERS
00990 00434H CALL    001DCH ;WAIT FOR PRINTER READY
01000      LD      A,C ;GET CHAR TO PRINT
01010      OUT     (0F8H),A ;OUTPUT IT TO PRINTER
01020      INC     (IX+05H) ;INC # OF CHARS PRINTED
01030      CP      0DH ;CARRIAGE RETURNS ?
01040      JR      Z,00445H ;FIX COUNTERS IF SO
01050      CP      0AH ;LINEFEED ?
01060      JR      NZ,00448H ;SKIP IF NOT
01070 00445H CALL    00046H ;ADJUST COUNTERS
01080      XOR      A ;CLEAR STATUS
01090      LD      A,C ;RESTORE CHAR
01100      RET
01110 00448H IN      A,(048H) ;GET PRINTER STATUS
01120      AND     0F0H ;MASK OFF IRRELEVANT BITS
01130      CP      30H ;READY ?
01140      RET
01150 ;
01160      ORG      03043H
01170 03043H JP      0378DH ;CHR NOT CTL, (41FBH)=1
01180 03048H JP      0377AH ;ADJUST COUNTERS
01190 ;
01200      ORG      0377AH
01210 0377AH LD      (IX+05H) ;RESET #CHRS PRNTD ON LINE
01220      INC     (IX+04H) ;INC # LINES PRINTED
01230      LD      A,(IX+04H) ;GET NEW LINE COUNT
01240      CP      (IX+03H) ;ON NEXT PAGE ?
01250      RET      NZ ;RETURN IF NOT
01260      LD      (IX+04H) ;RESET LINE COUNT TO TOP
01270      RET ; OF PAGE AND RETURN
01280 ;xxx CHAR NOT CONTROL & FLAG @ (41FBH)=1 xxx
01290 0378DH LD      A,C ;GET CHAR
01300      CP      0E0H ;CHAR <0E0H ?
01310      JP      NC,00043H ;GO IF NOT (TO 0434H)
01320      CP      0C0H ;CHAR <0C0H ?
01330      JP      C,00063H ;GO IF SO (TO 041FH)
01340 ;CHAR IN RANGE 0C0H-0DFH
01350      SUB     0C0H ;MAKE CHAR 00H-1FH
01360      LD      B,0DH ;BC=ADJUSTED CHAR
01370      LD      C,A ; (IN RANGE 0000H-001FH)
01380      LD      HL,(4220H) ;GET ADDR OF LOOKUP TABLE
01390      ADD     HL,BC ;ADD OFFSET
01400      LD      C,(HL) ;GET CHAR FROM LOOKUP TABLE
01410      JP      00063H ;GO TO VECTOR TO 041FH
01420 ;
01430      END

```

Memory locations 41FBH, 41FCH, and 4220H-4221H are used for the first time (by the printer driver) in the above code. I don't know why the changes were made, but the net effect is as follows:

1) If memory location 41FBH contains any value other than 0 or 1, all characters are handled normally. Unfortunately, this location is initialized with a value of zero. So if your graphics printer starts doing funny things when connected to a TRS-80 Model 4 or 4P with the new ROM (or ROM image), simply try POKEng memory location 41FBH with any number between 2 and 255.

2) If memory location 41FBH contains a value of 1, then all characters in the range C0H through DFH will be converted according to the contents of a 20H byte long table beginning at an address stored in 4220H-4221H (these locations contain an address of 0000H at initialization, which obviously is not pointing at a lookup table!). If you want to use this feature from BASIC, you could create a 32 (decimal) byte string literal (for example, A\$="...(32 bytes)..."), then use the VARPTR function to find the start of the string in memory and POKE that into locations 4220H-4221H. Don't forget to POKE a value of 1 into 41FBH. After that, any byte you send to the printer should be converted according to the characters in your string literal (C0H would be converted to the first character, C1H to the second, and so on...).

3) If the memory location 41FBH contains a value of 0 (as it does at initialization), then characters in the range C0H through FFH will have 20H subtracted from their value (they will be converted to characters in the range A0H through DFH). IN ADDITION, if (and only if) memory location 41FCH contains a value of 0 (as it does at initialization), then characters in the range A0H through BFH will have 40H ADDED to their value. They will be converted to characters in the range E0H through FFH).

The following chart may illustrate this a little better:

Value in 41FBH	Value in 41FCH	Original character value (Offset shown below)		
		A0H - BFH	C0H - DFH	E0H - FFH
2-255	XXX	no change	no change	no change
1	XXX	no change	table lookup	no change
0	1-255	no change	subtract 20H	subtract 20H
0	0	add 40H	subtract 20H	subtract 20H

C. THE NEW RS-232 BOOT ROUTINE CODE:

```

00100      ORG      304BH
00110 Q304BH JP      Q3179H ;VECTOR TO RS-232 BOOT
00120 ;
00130 ;xxx GET CHAR FROM RS-232
00140      ORG      3086H
00150 Q3086H IN      A,(0EAH) ;GET UART STATUS
00160      OR      A      ;DATA RECEIVED IN UART ?
00170      JP      P,Q3086H ;IF NOT, TRY AGAIN
00180      JR      Q30C6H ;ELSE GET CHAR FROM UART
00190 ;
00200      ORG      30C6H
00210 Q30C6H IN      A,(0EBH) ;GET CHAR FROM UART
00220      RET      ; & RETURN
00230 ;
00240 ;xxx ACTUAL START OF RS-232 BOOT ROUTINE xxx
00250      ORG      3179H
00260 Q3179H XDR      A      ;A=0
00270      OUT      (0EBH),A ;MASTER RESET OF UART
00280      LD      A,0EEH ;SELECT 9600 BAUD FOR
00290      OUT      (0E9H),A ; INPUT/OUTPUT
00300      LD      A,6DH ;INIT UART ODD 8 BIT 1 STOP
00310      OUT      (0EAH),A ; NO PARITY RTS BRK DTR OFF
00320 Q3184H IN      A,(0E8H) ;GET MODEM STATUS REGISTER
00330      BIT      6,A ;"DSR" ON ?
00340      JR      Z,Q3184H ;IF NOT, KEEP WAITING
00350      LD      A,6CH ;INITIALIZE UART AGAIN
00360      OUT      (0EAH),A ; WITH DTR ON
00370 Q318EH IN      A,0E8H ;GET MODEM STATUS REGISTER
00380      BIT      6,A ;"DSR" ON ?
00390      JR      NZ,Q31BEH ;IF SO, WAIT FOR OFF
00400      LD      A,0FH ;CHAR TO TRANSMIT = <SI>
00410      CALL     Q37D5H ;OUTPUT CHAR TO RS-232
00420      CALL     Q3086H ;GET A CHAR
00430      CALL     Q3086H ;GET ANOTHER CHAR &
00440      CALL     Q37D5H ; OUTPUT IT TO RS-232
00450      JP      Q3517H ;CONTINUE IN ROUTINE
00460 ;
00470      ORG      3517H
00480 Q3517H OR      A      ;WAS 2ND CHAR A ZERO ?
00490      JP      NZ,Q3179H ;IF NOT, ABORT AND RETRY
00500      LD      HL,4300H ;4300H = NORMAL BOOT BUFFER
00510 Q351EH CALL     Q3086H ;GET A CHAR
00520      LD      (HL),A ;SAVE IT TO BUFFER
00530      INC      L      ;INCREMENT BUFFER POINTER
00540      JR      NZ,Q351EH ;BACK TO 4300 ? NO, DO NEXT
00550      JP      Q370EH ;GO TO EXIT ROUTINE
00560 ;
00570      ORG      37D5H

```

```

00580 Q37D5H PUSH     AF      ;SAVE CHAR
00590 Q37D6H IN      A,(0EAH) ;GET UART STATUS
00600      BIT      6,A      ;ANY CHAR WAITING ?
00610      JR      Z,Q37D6H ;IF NOT, TRY AGAIN
00620      POP      AF      ;RELOAD CHAR
00630      OUT      (0EBH),A ;GIVE TO RS-232
00640      RET      ; AND RETURN
00650 Q37E0H LD      A,6DH ;SET "RTS" OFF
00660      OUT      (0EAH),A ; AGAIN
00670      JP      (HL) ;JUMP TO 4300H
00680 ;
00690      END      Q304BH

```

Quite frankly, I'm not absolutely certain of the purpose of the above code, but I believe it was designed to allow the 4P to be booted off the RS-232 interface in a "networking" type situation. As most 4P owners know, holding down certain keys while booting up the 4P will select different modes of operation (see "MODEL 4P BOOT MODE KEY SELECTION" in NORTHERN BYTES Volume 5, Number 4, page 15). One of the "undocumented" combinations is to hold down the SHIFT and BREAK keys while booting up, which supposedly will boot from the RS-232. This is most likely the code that handles that.

D. CONCLUSION

If my mail is any indication, the most significant change in the "new" RCM (the one that seems to be giving folks the most fits) is the revised printer driver. So remember, if you have a program that prints properly when using a Model III (or an early Model 4), but doesn't work correctly when using a "new" RCM Model 4 or 4P, go to BASIC and type:

POKE 16891,2

This "should" restore normal operation. Since this memory location (41FBH) is unused under the "old" RCM, you may wish to add this instruction near the beginning of your Model III/4 BASIC programs that send graphics to the printer.

Once again I wish to give special thanks to John Hallgren for providing the bulk of the above information, and for proofreading and debugging a first draft of this article.

ONE MEG RAM FOR YOUR TRS-80

by Gary Bryce (82)628-5058

Having just finished the installing a 1 Meg RAM Supermem kit from ALPHA TECHNOLOGY into my Model 4P recently, I thought that I would pass on a few tidbits to anyone considering a similar conversion.

THE KIT

The kit is available for the Models 1/3/4 and 4P with or without memory (the memory can be supplied in increments of 256K :- ie. 256K, 512K, 768K, 1024K). The volatile nature of RAM prices (and the AUST\$) will determine whether it is cheaper to buy a kit with RAM or obtain the required chips locally.

FITTING THE BOARD

The 1 Meg Supermem for the 4P (not having seen any other version I can't say whether they are any different) consists of a single PC board, which plugs into the 200 CPU socket with twelve wires leaving the board to various IC's on the main computer board. The positioning off the board in relation to the metalwork of the chassis of the 4P leaves something to be desired. To prevent the back edge of the metalwork recess short circuiting the SUPERMEM board, approximately 1cm of the back edge and rear filler panel must be removed with a nibbler or similar tool for approximately 120cm.

Although a publicity "blurb" says "just tell us what kind of graphics board you have when you order and we will send you the correct version" (of the Supermem - presumably), the board does foul the "Graphics Solution" hi-res board as the Supermem board is mounted directly underneath the graphics board. The only way to solve this problem was to replace the existing 34pin connector

to the hi-res board on the main CPU board with another approximately 1cm taller - this also necessitated the removal of the metal bracket in the centre of the chassis recess.

FITTING THE RAM

If you are familiar with the old 48K in the keyboard modification using 4116's for the Model 1 then the actual installation of the RAM chips should not prove difficult to physically perform. The first two banks of 256K chips go into the existing RAM sockets while the 768K and 4K (1 Meg) Supermem upgrades involve "Piggy Backing" the 3rd and/or 4th banks of RAM onto the 1st and 2nd banks respectively - so some skill with a soldering iron is definitely required.

INSTALLATION INSTRUCTIONS

The installation was not all "Beer and Skittles" though, and I would have been in considerable trouble had not two other intrepid travelers gone the same way. The installation instructions were incorrect in one section (reversal of two wires required - the purple and grey wires on the kits I have seen) and two other sections conflicted with each other (sections 6 and 7 on the 4P sheet are the valid ones).

INSTALLATION PROBLEMS

Add to this the fact that two of the units would not work with ROY SOLTTOFF's patches to TRSDOS 6.2 initially. Two of the boards were equipped with a HITACHI 74LS244 (with the board viewed from the top and the Z80 socket to the left, the 74LS244 is the 20 pin IC directly to the right of the Z80 socket) and the working board had a FAIRCHILD IC. Not having any FAIRCHILD chips on hand the HITACHI chips on both boards were replaced with NATIONAL SEMICONDUCTOR versions of the LS244 with the result that the boards now function correctly. This problem did not show up during the initial test of the installation or during formatting of a RAMDISK using the RAMDRIVE program, the only common factor here appears to be that banks 0 and 1 are not checked during either of these procedures. Needless to say a considerable amount of frustration abounded at the time.

Having turned you all off now let me say that this mod is to be the MOST FANTASTIC and INCREDIBLE addition that I have done on any TRS-80 - and I have done a few!!

RAMDRIVE SOFTWARE

The RAMDRIVE software (\$45 extra) gives you the capability of formatting a RAMDISK of up to 876K with room left over for PRNTO etc. RAMDRIVE is installed in the same manner as MEMDISK with the added bonus that it can be installed as drive 0. On initialisation RAMDRIVE checks to see how much RAM has been installed, automatically reserves the normal 128K and then asks if you wish to exclude any further banks for other purposes before asking whether you want to perform a "FULL" format or not. The "FULL" format actually performs a dynamic memory test of all memory used by the RAMDISK, exercising it four times to verify its integrity. If the full format is not selected the memory is still initialised (and verified) to binary zeroes. The bank switching scheme is exercised by either option to ensure it's integrity.

The formatting pattern of the RAMDISK is not compatible with 5" floppies, so backups to and from it must be done as a BACKUP-BY-CLASS, using the MOD flag to save files back to floppy disk before powering down. If you have either the LSI program "DiskDISK" or "PRO-ADE" from MISOSYS you can format a "Subdisk" on the RAMDISK that is an "IMAGE" of a 5" disk (be it 40/80 track, Single/Double density or Single/Double sided) to allow a "MIRROR IMAGE" backup to and from the RAMDISK, using this method, it takes 210 seconds from switch-on to TRSDOS Ready and 45 seconds after pressing RESET. I have found that the RAMDISK is still active after running NEWDOS or in Model III mode and returning to TRSDOS 6.2. The RAMDRIVE software aborts a /JCL file if you try to use one to initialise the RAMDISK and then copy your "SUBDISK" and initialise it, this can be done using the LSI program "TYPEIN" or any similar program to take keyboard input from a file.

CONCLUSION

Having used the RAM for about three weeks now, I would find it very difficult to be without it. The problem with the HITACHI chips appears to have been due to a bad batch of IC's and not a design error. More serious is the error in instructions, I read in his review that Hardin Brothers experienced the same or a similar problem which should have been corrected by the time my board was purchased. It may be reasonable to telephone FLORIDA from CALIFORNIA but repeated calls from AUSTRALIA become rather expensive (as Denis Pagett has found out).

I have mixed feelings about the unit, when you spend over A\$500 (Supermem plus RAM) on something, you have a right to expect it to work (unless you foul it up yourself!!). So if you are considering purchasing one, don't attempt to instal it yourself unless you are a true "Hardware Hacker", get a "Techo!" to do the job and be prepared for possible hold-ups.

Would I buy it now in hindsight? I truly don't know, had I been asked three and a half weeks ago I would have said NO!, but the sheer convenience of having as many as fifty programs "IN MEMORY" plus data at one time has to be seen to be believed.

TRSDOS PATCH ERROR

by Gary Bryce

There was an error in the first patch listed for TRSDOS 6.2.x in the Feb issue of the newsletter, the patch to disable password checking should be as follows:

```
PATCH SYS2/SYS.LSIDOS (D82,33=18;F82,33=28)
```

Sorry, it must have been a case of "FAT FINGERS" as the JCL that I used to test these patches has the correct info and I had copied it directly into the article to prevent just such an occurrence, oh well, we can't all be perfect no matter how hard we try.

Please note that many of these patches will be overridden by any CONFIG/SYS file which you may already have on the patched disk (the actual purpose of most of these patches is to allow you to create a disk to your own specifications without having to waste space on a disk with the configuration file), so hold down the CLEAR key when you BOOT the disk and then type SYSGEN (OFF).

THE PROFESSIONAL DOS

by Michael Cooper

Welcome to the group for those of you with the newer 8 bit Tandy machines. If you have a Model 4 or 4P then you will already know of the advantages of using LDOS/TRSDOS 6 to simplify data mobility. Since Tandy now support the Logical Systems DOS's and as LDOS and TRSDOS 6 are compatible in disk format and in most respects compatible in command syntax, it seems to me that sensible people would be using one or both of these DOS's. If you have a Model 3 then you will show great forethought by using LDOS on your machine.

The advanced device independent features of LDOS and the well documented system calls make it easier to use at machine code level than any other DOS, especially NEWDOS.

Ongoing support and patches from Logical Systems allow these 2 DOS's to grow with new hardware add-ons. An example is the new 1 Megabyte RAM boards that can be installed in a Model 4/4P. Roy Soltoff of MISOSYS has already supplied the patches to the TRSDOS 6 Supervisor Calls to allow DOS to Bank switch the whole 32 Banks.

If you want your Tandy to be still useable in a few more years then you must begin to use this DOS now before it becomes too late. Apparatus have disappeared from view, so NEWDOS is dead, and no longer supported by its creators. DOSPLUS is rapidly coming within the LDOS/TRSDOS 6 standard and MULTIDOS is only good for format switching of data & from DOS X to the Tandy standard. Get with the strength...

SUPERSCRIPSIT MOD 4

by Jim Whittaker

I have just zapped my superscriptsit (mod 4 version) to stop those infernal triangles appearing all over the screen when I press the space bar twice. The two zaps are for version 1.0 and for both the keyboard handler and the ascii conversion. (ED.NOTE :- I have presented these patches in the original format that Jim supplied as well as the normal patch format, so that those who have not got the LSFEDII program or similar utility may still perform the patches.)

SCRIPSIT/CTL RECORD X'0014'
LOAD ADDRESS X'43D2' CHANGE FE 20 20 TO FE 20 10

PATCH SCRIPSIT/CTL(D14,20=FE 20 18:F14,20=FE 20 20)

SCR35/CTL RECORD X'0001'
LOAD ADDRESS X'99C7' CHANGE FE 20 20 TO FE 20 10

PATCH SCR35/CTL(D01,30=FE 20 18:F01,30=FE 20 20)

Note that these patches are in the format used by LSFEDII. HAPPY WRITING ----- REGARDS Jimbo

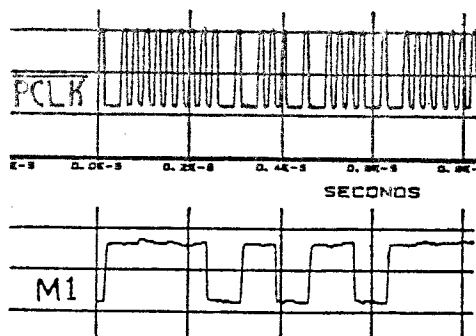
FAST FIX FOR MOD 4 SLOWS

by Dave Owen, CCS Technical Specialist

Reprinted from "Northern Bytes" where it was reprinted from the "Micro Info Exchange" (newsletter of the Cabrillo Computer Society.)

The Radio Shack Model 4 Computer is advertised to operate at a two or four megahertz clock rate. Our investigation has revealed the truth of this claim. The Tandy specifications do not, however, state the fact that the master clock does not always 'tick'. Because of this persistent miss, the machine operates at approximately 3.37 MHz or nearly one-sixth slower than claimed.

The pre-programmed PAL (programmable array logic in the back panel's U3 socket) logically ties the op-code fetch signal (M1) to the crystal oscillator output (*PCLK). The result is that clock cycles are suppressed during each op-code fetch on those PAL's labeled with Radio Shack part number 8975166. A later version PAL has reportedly eliminated this design feature. But most of these sold in 1983 and earlier do run below specifications.



This shows how M1 causes *PCLK (Primary Computer Clock) in U3 to also output erratically, so that the missing pulses mean, on balance, slower speed.

The clock suppressions aren't needed with the Z80A (the computer was originally designed to operate with a ZILOG Z800 that was never released). If you have this particular PAL, the solution is simple :- Remove IC U3 by prying it up at each end with a small screwdriver. With a pair of small pliers, bend pin 7 horizontal so it will no longer sit in the socket. Replug the chip back into your board (being careful to orient it in the original direction). Presto, a 4 MHz operating speed.

Does this really make a difference? Software, particularly the operating system, is "tuned" to certain operating speed environments. When formatting my 8 inch drives under TRSDOS 6.2, for example, the time required dropped from 8 minutes to just under 2 minutes. Other

operations also work more efficiently, such as mathematical calculations. We timed FOR-NEXT loops at 16% faster. Try it, you'll like it. And it's free!

Public Domain Software

Below is a listing of the Public Domain Disks to be released this month. The disks are available at group meetings at \$5 per disk (2 volumes) to members and \$10 to non-members (makes it cheaper to join the group if you want a couple, doesn't it!). Each disk is a true "FLIPPY" formatted as 35 track Single Density and holding 2x88K (176K) of software.

Country and Interstate members can order via Mail by sending cheque or money order for \$5 per disk plus \$1 for postage, to :- SYDTRUG, P.O. Box 297, PAOSTOW 2211. Cheques should be made payable to "Sydney TRS-80 Users Group" please.

PDBUSI07 : Business & Home

BUDGET/BAS Complex budget manager
FILEMAST/BAS Information filing system
FINRATIO/BAS Financial ration analysis
HOMEBDGT/BAS Home budget manager
INFLATON/BAS Inflation projection
LEDGER/BAS General ledger
MAILLIST/BAS Home mailing list
PCALENDA/BAS Perpetual calendar
PHONWORD/BAS Make words out of phone numbers
REGWITHD/BAS Regular withdrawal interest calculator
SEASONAL/BAS Seasonal financial analysis
STOCKPAG/BAS Stock market tracker
TRENDING/BAS Billing trend analysis
YIELD/BAS Interest yield on bonds

PDBUSI08 : Business & Home

ACCTREC/BAS Accounting records for a small business
ANNUITY/BAS Calculate annuity of principle
BRKEVEN/BAS Calculate production of break even after sales
CALNCALC/BAS Calculate calendar dates
CASHBOOK/BAS Cash book journal entries for ACCTREC
CASHREGS/BAS Cash register simulation
CHECKWIZ/BAS Verify and record bank statements
DDEPREC/BAS Double-declining balance depreciation
ELECOSTS/BAS Electrical usage costs
EXECDEC/BAS Executive decision maker
FINCALC/BAS Financial calculator
LOANPAY/BAS Calculate total interest and av. loan payment
MEANSTD/BAS Total mean, std deviation, std error of mean
ORDQUANT/BAS Most economic quantity of units to order
PRESVAL/BAS Calculate present value of amount of money
PRESVALA/BAS Calculate present value of annuity
REACTEST/BAS Timed reaction test
SALARY/BAS What you don't know about your paycheck
SBACCT/BAS Small business accounting procedures
SIMPLINV/BAS Store, recall, and update inventory
SINKFUND/BAS Sinking fund and fixed investment calculations
SLDEPREC/BAS Straight-line depreciation
STEXTED/BAS Super text editor
SYDEPREC/BAS Sum of the years digits depreciation
UNITPRIC/BAS Find the better buy through unit pricing

POEDUC06 : Education

ALGEBRA/BAS Learn factoring of polynomials
COUNTRYS/BAS Select a country, and the computer will it
GEOMETRY/BAS Introduction to geometry
PLANETS/BAS Learn about our solar system
POEMCOMP/BAS Compose your poem with the computer's help
PREFIXES/BAS Learn common prefixes for building vocabulary
ROMANUM/BAS Learn the Roman system of counting
SPELLBEE/BAS Test your spelling skills
STATGUES/BAS Guess the United States

POEDUC07 : Education

ACCEL/BAS Calculate acceleration until impact
ALGEBRA1/BAS Learn distributive axiom of multiplication
ALGEBRA2/BAS Extract binomial roots of simple polynomials
BASE2DEC/BAS Convert base 2 to 9 to decimal equivalent
CASHREG/BAS Cash register simulation
DICESTAT/BAS Print out probability of dice throws

DIVISION/BAS Long division practice
 ECHONUM/BAS Repeat sequences of random numbers
 ENGLANT/D01 Data file for HANGMAN
 ENGLFREN/D01 Data file for HANGMAN
 ENGLSPAN/D01 Data file for HANGMAN
 ENGLSYNO/D01 Data file for HANGMAN
 GRAPHICS/BAS Tutorial on TRS-80 graphics
 GRAPHING/BAS Tutorial on graphing techniques
 HANGMAN/BAS Bilingual hangman game
 HANGMAN/DOC Documentation for HANGMAN
 HANGMAN/DAT Data file for HANGMAN
 MATHCHR/BAS Improve mathematics aptitude
 METRCON/BAS Metric conversions
 MULTIPLY/BAS Sharpen multiplication skills
 PROGRESS/BAS Compute geometric & arithmetic progressions
 PROJECT/BAS Calculate date pertaining to projectile flight
 TIMPIECE/BAS A clock for the TRS-80
 WORDIN/BAS Another word quiz
 WORDQUIZ/BAS Test ability to identify spelling words

PDGAME13 : Games

BARTER/BAS Assume various occupations and trade goods
 DOTDOTDOT/BAS Connect the dots and claim the boxes
 FLIPDISC/BAS Outflank the computer and gather more pieces
 ROUTLETT/BAS Give the barrel of the revolver a spin
 TAROT/BAS Yes/No spread of ancient fortune-telling cards
 TRAP/BAS Two-player action as you avoid being trapped
 TRESHUNT/BAS Explore the Lumus Caves in search of treasure
 WESTWARD/BAS Can you survive the trek westward in 1847?
 WIPEOUT/BAS Elimination game
 ZCHESS/CMD Chess with 18 levels of difficulty

PDGAME14 : Games

AIRCRAFT/BAS Pilot a fighter against kamikaze jets
 AIRPILOT/BAS Only the foolhardy need apply for the job
 ASTROL/CMD Become an astrologer
 CAMELOT/BAS Stop Mordred's siege of Camelot, or help him
 DEADSTIK/BAS Pilot Columbia on it final approach
 PIGSKIN/BAS Computer football game
 SPACWAR2/BAS The Deathstar, you goal, is heavily shielded

ME15 : Games

AIRRAID/BAS How well can you handle the air raid?
 DARTS/BAS Play darts from a new perspective
 DOGFIGHT/BAS High-flying shoot'em up
 FLIGHT/BAS Simulated runway approaches
 IGOR/BAS Elimination game with an insulting robot
 JETFLYER/BAS Pilot a jet in combat
 KREGII/BAS Tank survival game
 PARACHUT/BAS Bail out of a crashing plane
 POWERS/BAS Power of concentration game
 RADAR/BAS Air Traffic Controller program
 SAILPLAN/BAS Ride the air current to stay aloft
 SAND/BAS Building game

PDGAME 16 : Games

ALIEN/BAS Shoot down the invading creatures
 ALIEN/INS Instructions for ALIENS/BAS
 BLOCKADE/BAS Manuevering game
 BLOCKERS/BAS Break through the walls
 CARTOON/BAS A short space tale
 DOGSTAR/BAS Escape from the alien space station
 ELECTION/BAS 1988 Election simulation
 GALAXY/BAS Real time space combat
 HANGMAN4/BAS Word game
 KALEIDO/BAS Kaleidoscope for your TRS-80
 SPACRACE/BAS Collect treasure and avoid mines
 TRIFIGHT/BAS Survive attacks by the Trifighters
 ZARBOR/BAS Explore the moon

PDUTIL87 : Utilities

BITSRCH/BAS Search utility
 TSTRNG/BAS String search utility
 CK/BAS On-screen clock utility
 DIRMENU/HEX Disk directory manager
 DISKCAL/BAS Disk calabrator
 EDTASM/DOC Reference card for EDTASM
 ELEMUP/BAS Element duplication subroutine
 FLASH/BAS Flash video routine
 FREEFORM/BAS Free form video entry program
 MENU/MAKE/BAS Program generator
 MONITOR/BAS Status monitor

NDHIMEM/DOC Himem setting hints
 PRGLISTR/BAS Program lister
 PRNDRV/ASM Printer driver
 SEARCHER/BAS Search routines
 SORTKEY2/BAS Sort on keys utility
 SORTKEY3/BAS Sort on keys utility
 TRSDOS13/DOC Reference for TRSDOS 1.3
 TRSPOKE/DOC TRS-80 Peeks & Pokes
 VARPASS1/BAS Variable passing routine
 VARPASS2/BAS Variable passing routine

PDUTIL88 : Utilities

AUTOPROG/BAS Program Generator
 AUTOPROG/DOC Documentation for AUTOPROG
 BFUI/BAS Basic File Utility
 KBRDR/BAS Keyboard disk driver
 KBDISK/HEX Hex file for keyboard driver
 LHELP/CMD LDOS Help Utility
 LDOSHELP/HEX Hex file for LDOSHELP/CMD
 MXLABELS/BAS Label Maker for MX-80
 NOBUG185/BAS Note Bug Message Center
 SCROLL/BAS Screen Scrolling Utility

The above volumes are being released in pairs - TWO to a disk as follows (There will be FIVE disks released this month) :-

PBUS187 + PBUS188
 PDEDUC86 + PDEDUC87
 PDGAME13 + PDGAME14
 PDGAME15 + PDGAME16
 PDUTIL87 + PDUTIL88

The cost of each of these DISKS is \$5.00 each, plus \$1.00 postage for up to 5 disks.

DISCLAIMER

No Patent liability is assumed with respect to the use of the information contained herein. While every precaution has been in the preparation of this publication, the Sydney TRS-80 Users Group or it's appointed office bearers assume no responsibility for errors or omissions. Neither is any liability assumed for damages arising from the use of any information contained herein.

DOUBLER RAFFLE

The winner of the Model 1 Doubler Raffle was drawn at the Sifton meeting on the 22nd of February (postponed from the previous week due to poor attendance). The winner is :-

Gordon Symonds of Burwood

Congratulations and we hope that it proves to be of great use to you!

