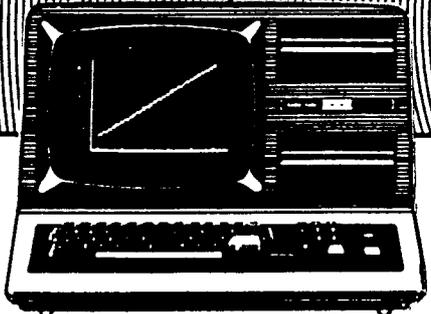


61

TRS-80 SYSTEM 80 COMPUTER GROUP

BRISBANE, QLD.



REG. BY AUSTRALIA POST PUBLICATION # QBH 3667
 CLUB NEWSLETTER
 ISSUE: JANUARY 88

COMMITTEE 1987-8:

PRESIDENT / CHAIRMAN	: B ADAMSON	378 7408	
VICE-PRESIDENT	: A WEST	399 5620	ADDRESS ALL MAIL TO:
TREASURER	: J HANDLEY	397 5872	GEN. SECRETARY,
GENERAL SECRETARY	: F SECCULL	263 6313	41 MONTCLAIR ST.,
NEWSLETTER EDITOR	: W ALLEN	343 5771	ASPLEY,
LIBRARY & TOPICS MGR	: J BIRD	261 1952	QUEENSLAND 4034
PROGRAMS CO-ORDINATOR	: N HODGE	287 5020	
PRINTER	: J MELKSHAM	814 4333	
COMMITTEEMEN: B. ARTHUR (075) 48 2839, P. GOED 880 1136 & B. STOMFAI 300 4129			

MINUTES OF MONTHLY MEETING DECEMBER 1987

Meeting held at St. Cintra's Hall, Bowen Hills on December 6th 1987 at 2 p.m.
 Apologies received:- Geoff Lorne, Bob Maskiel.
 The President, Barry Adamson, opened the meeting and greeted members, new members and visitors.

GENERAL SECRETARY'S NEWS: The Secretary reported the month's correspondence, which included a letter to Tandy Australia, asking for a contribution, preferably in the form of equipment, to assist in the operation of the Group's bulletin board service, as well as the usual members' requests and some new membership applications.

EDITOR'S REPORT: The Editor reported that the Newsletter was at last getting some new authors. He would particularly welcome articles from members on the subject of what you are doing with your computer, or how you got involved with your machine.

TREASURER'S REPORT: John Handley presented the monthly report.

Income	\$217 (Subs & Library)
Expenditure	\$496 (\$20 Petty cash, \$30 Rent for 2 months \$446 2 months rent photocopier)
Excess Expenditure over Income	\$297
Cash Balance	\$2800

PRINTER'S REPORT: Jim showed examples of the card which he has obtained for covers for the documentation, (the extra cost will be 65c) and of the indestructible paper, which he recommends for such things as making lasting copies of your quick reference cards.

PROGRAMS CO-ORDINATOR: Noel reported that he has a Dot Writer for 7 pin printers and a Mailing list program for Model 4, Powermail.

LIBRARY: John asked members who want to refer to technical manuals to order these with the disk orders, as we do not normally bring them to meetings.

MEMBERS' PROBLEMS - DECEMBER 1987

NOVEMBER FOLLOW UP

John Aarsse (problems backing up disks on a Model I under Multidos and Newdos 80) - still under investigation.

Dave Gillingham (memory location for the poke to change the cursor character) - 4023H for the Model III, 4022H (16418 decimal) for Model I location is correct for non-disk basic; otherwise 4501H. Article following.

John Bird (single key entry of DATA statements in BASIC programs) - Micro 80 program (FILES/BAS) listing and tape obtained from Bill Stomfai. Works as required but is in non-disk basic - to be amended to run under NEWDOS80.

George Russell (format problems with an external Shugart 40 track SS drive) - still outstanding.

Jack Bognuda (difficulties with Editor Assembler Plus) - there was some confusion between object and source code which has now been resolved, but the buffer space (even using EDAS) is still not large enough for the required program and a modular approach will be necessary.

Barry Keats (graphic dumps from an Amstrad to an Epson printer) - now identified as a software problem.

Terry McCarthy (Typing Tutor bug on the Model IV) - fixed.

John Hildyard (replication of dates with Visicalc) must use string functions (which not all versions of Visicalc have) or numeric format.

Duncan Cameron (locked block with Superscript on a Model III) - still outstanding.

DECEMBER PROBLEMS

Bruce Bussenschutt was after a copy of the IJG book on Disk I/O - Peter Goed has a copy.

Dave Clarke is looking for a power supply for a hard disk drive - must be able to provide 4 amps at 12 volts and 1 amp at 5 volts. Bill Allen suggested Jerry Fox as a possible contact.

Jim Melksham has encountered problems in using Longview (a utility which prints Visicalc spreadsheets sideways) on an Epson LX-86. Bill Stomfai to follow up.

George Kirkham is trying to locate Model IV memory upgrade documentation - Peter Townson has the full manual and Barry Keats has recently carried out the upgrade.

During discussions regarding the Club establishing its own Bulletin Board, the subject of the Tommorowland Bulletin Board arose. Although the Club has its own area on this Board, the sysop had banned 300 baud between 6 am and midnight, which prevented most club members from accessing the system. This ban has recently been lifted and the new number is 371-0944.

GENERAL BUSINESS

Swiftforms have supplied us with a new introduction card for members to get discounts on printing and sundry supplies. The Secretary distributed some to members at the meeting. Any member who was not at the meeting but who would like one, please contact the Secretary..

The President, reporting on the lack of response to the request for feedback on the future direction of the group, said that we would have to assume that we were doing things which met with general approval. However, if anybody has any suggestions as to how things could be improved, these would indeed be very welcome.

DISCUSSION OF B6B No 60

Peter Goed then led the discussion on the current B&B.

The meeting finished at 15.30 and was followed by a talk by Peter Townson on Disk Drives. This is again a subject which members would benefit from followups (perhaps little and often would be the ideal). Many thanks Peter.

The next meeting will be on FEBRUARY 6th at St. Cintra's Hall, at 2 p.m. Peter Goed will give us a run down on the Bulletin Board and how we should use it. See you there.

BETTER COMPUTING, Fred Seccull

IMPORTANT NOTICE

The Committee held a special meeting at Alf West's on Saturday 23rd January to discuss the group's proposed constitution and possible incorporation. At this meeting it was decided to proceed with incorporation. It will, therefore, be necessary for the proposed constitution to be adopted by a General Meeting of the group.

IT IS HEREBY NOTIFIED that the main purpose of the March 6th General Meeting, to be held at 2pm at St. Cintra's Hall, Bowen Hills, will be to ADOPT A CONSTITUTION, based on the Model Rules supplied by the Justice Dept.

Notice dated 26th January, 1988.

-----<TRS80SYS80CG>-----

THE MARKET PLACE

THIS SECTION IS FOR THE INFORMATION OF MEMBERS' WANTS, AS WE NOW ARE PRACTICALLY THE SOLE CONTACTS IN OUR AREA FOR MATERIAL FOR OUR MACHINES.

HARDWARE ETC FOR SALE:

1 Tandy DWP510 Printer 6 months old w/ tractor feed & spare ribbons and daisy wheels. Current top of line model -- value approx. \$3000. It's yours for \$1150 if you hurry, because of upgrade to Laser printer. Urgent sale, must sell by or at Feb. meeting. Contact P. Goed, 880 1136.

2 Model 3s 48K 2 disk drives RS232 i/face orig manuals dust covers etc. Heaps software. \$1000 the lot or will split. /also Multiplan for Model 4 (offer). Contact Ken Stewart, 271 1211 ah 848 9246.

Complete set 80 Micros from start to June 87 -- will split. Model 4 hires board, Model 3 hires board, Mouse interface, speech synthesiser. Will listen to offers for any of these goods. Contact Jerry Fox on 071 43 2153 or after Feb. 6th meeting closes.

2 Tandon 40 tr ss 1/2 ht drives ex 4P, nr new \$120 each. Contact J. Bognuda on 231 0674

Re PETER GOED'S 'GOODIES'. At the February meeting I intend to liquidate all remaining stocks from my van after the meeting. If not sold then, the remainder will go to the rubbish tip, so I can get back into the computer room.

SOFTWARE

- # NEWDOS86 -- Warwick Sands' update package for NEWDOS80 v2 (Model I, System 80, Model III, Model 4/4P in Model III mode), has just about all the extra functions you've always wanted in an operating system, plus very powerful CUSTOM BASIC FOR '80 USERS programming and runtime enhancements to Disk Basic and screen oriented Enhanced Line Editor. Ongoing support always available, being followed up by further enhancements. \$75 for either Model version, \$5 extra for both versions (Installation disk(s) and 150+ page manual). Add \$5 for P&P if by mail in Australia. Obtain from Bill Allen at meetings, or 16 Laver St., Macgregor, Qld. 4109, (07) 343 5771.
- # HELPDISK -- a facility to interrupt most programs under Newdos80 (or Newdos86) or from Basic or Dos Ready to comprehensively, rapidly and flexibly view information on any of the specially formatted and indexed /HLP files in the system and return with the interrupted status completely restored. Proceeds to CLUB FUNDS. Once only cost: \$10 (contact Bill Allen as above).
- # MACASMON -- a very powerful MONITOR program for the Models 1, 3, and 4/4P (in MODEL 3 MODE ONLY). (See review in Bits and Bytes No.47 P.8) Full documentation files on issued disk. Features 6 separate screen displays, excellent disassembler (forwards AND backwards), screen snapshot capability, buffers last 7 executed instructions, calculates and keeps running total of T-states, etc etc etc. (Sold on shareware basis) Bring \$5.00 and pick it up at any meeting (see Bill Allen), or send \$10.00 and I will post it. Glen Mc Diarmid, 28 Marginson St., Ipswich, Qld, 4305 (07) 281 7057
- # TIME MACHINE FOR THE 4/4P (Utility). BARGAIN !!! NOW ONLY \$20 + \$5 pandp anywhere in Australia. Includes Disk and manual. (See review on Page 10, Bits and Bytes No. 45). Add concurrency to Model 3 dos's on the 4P, run 2 programs simultaneously in the machine, cheat like crazy at games (learn to master even the most difficult key control -- e.g. would you believe over 200,000 scored from Volcano Hunter?), plus numerous other VERY USEFUL functions using banked memory and sophisticated key control of the machine available at all times. A MODEL 4 VERSION IS BEING PREPARED at last so that these machines can get in on the act. It has been successfully tested, and there are only some loose ends to tidy up. Contact Peter Goed (07) 880 1136.

BOOKS:

- # NEWDOS/80 HACKERS HANDBOOK by Kevin O'Hare. Packed with useful info for programming under this DOS and memory usage in our machines, VERY USEFUL FOR OTHER DOS USERS ALSO. \$25 (add \$4.00 if ordered separately by post). Contact Peter Goed at meetings or 203 4882 or Bill Allen.

-----<TRS80SYS80CG>-----

BITS & BYTES

ISSUE NO. 61

THE NEW BULLETIN BOARD

by assistant Sysop, Bill Allen

Well, Peter got the board up and running quickly enough during the break -- he certainly doesn't let the grass grow under his feet. It has proved a very friendly system that seems fairly bomb-proof now we've ironed out a few initial kinks.

There was quite a lot of unsatisfactory (to us) spelling and screen formats built into the board and we took a great deal of trouble to modify it much more to our ideas.

The number? Well, would you believe it's Pete's old number --

(07) 203 4882.

It's on line for 24 hours 7 days a week, so go to it, members. We haven't uploaded many Public Domain programs yet, as there are a couple more hard drives to be installed (maybe one from Tandy) and we don't want the extra job of copying lots of files all over the place until the hardware is fully installed.

LOGGING ON --- OK, so you initialise your modem program (there's a good 1200/75 one on the board to download and use later), dial the number, wait for the modem to auto answer and when you hear the constant pitch carrier tone, you then switch to modem operation. The first time, you will be asked a series of questions to answer and then you will be asked to repeat your answers to verify them to ensure your bona fides. Answer them truthfully (no bull) or the system throws you off -- this is out of our control -- if anyone plays up on the board, it will promptly throw him off. The next time the sysop checks the new callers (which is usually daily), you will be logged in as a member and receive a status to access the various departments on the board thereafter. READ and use the MENUs (MENU and MENU2) to find your way about. TRY ALL THE OPTIONS! Download HELP files -- we still have to put some more suitable ones for this purpose, but only because the ones for the display have control codes in them which send some printers nuts. But for the present you could scan them at leisure with your word processor if you'd rather do that than scan them on the board itself -- which takes a lot of time (there's about 50 k of them so far).

SOME DO'S AND DON'TS:

DO -- When you hear the carrier tone and have switched the modem to online (modem position), put the phone down in its cradle or whatever to switch it out. Some of us have had troubles caused by the phone echoing garbage into the works, which ceased when we put the phone off the line. Previously when we had been sending files to each other in normal modem operations, leaving the phone off its cradle had not caused any problems.

DO -- When you've had enough, LOG OFF PROPERLY by typing BYE and several cr's or BYE Q. This can be done by selecting the exit from a menu or from Remote Newdos Ready.

DO -- Read the information on how to leave messages and USE THIS FACILITY as much as possible. It's one of the main facilities on a BBS, so familiarise yourself with the method of using it right away -- reading, entering, killing or editing your messages.

DON'T -- Type in bad language or the board will hurl you off so quick your head will spin. Young people will be using the board also, so all efforts are made to maintain decorum.

DON'T -- Tell funny things to the board (Mickey Mouse etc is strictly non U) when logging on at first, because YOU DON'T GET FURTHER ACCESS unless the information you input checks out. This is to protect the board from evil intent creatures strewing their garbage over it.

DON'T -- Forget to kill messages addressed to yourself after you've read them. This greatly assists the sysops to keep the board tidy. Actually, if a message is left there too long, it is automatically deleted but hopefully not much of this will be necessary.

I think that's enough to get you started for now. Please let us know any ideas or observations you may have -- by LEAVING A MESSAGE TO THE SYSOP in the message section rather than the comments just before disconnecting.

We will put a lot more departments into the system as we develop it from our own ideas and input from users.

Happy BBS'ing,

Bill

-----<BNB>-----

HOW I GOT HOOKED - CONFESSIONS OF A COMPUTER JUNKIE

by Bill Stomfai

Like a lot of other people in the Club, my involvement with computers needs to be viewed in a historical perspective -- many people today don't realise that there were any microcomputers before IBM arrived (rather late) on the scene.

Back in 1979, IBM had never been heard of in the microcomputer field; the whole thing only started in 1976/77 when APPLE launched their first offering. Prior to this, the only micros around were electronic hobby kits that you made up yourself -- the thought of commercially available software for micros hadn't occurred to anyone because they didn't think that there was any market for it. By 1979, the big names in personal computers were APPLE, TANDY (the TRS-80 Model I had only been around for a year or two, but was very popular), COMMODORE (with the PET, which looked a bit like the Model III), and SINCLAIR (with the ZX-80). Very few systems had disk drives; most were cassette tape only. The ONLY 'business' machines ran CP/M - usually on eight inch disks with 64K, dBase II and Visicalc.

At this time, I was most of the way through a management degree and trying different hobbies every few months. I knew a little bit about (mainframe) computers from my studies, but tertiary institutions generally considered micros beneath their notice and weren't quite sure what to do about the new programmable calculators.

Then a friend(?) of mine (an electronics hobbyist) happened to mention that an acquaintance of his was moving on to a new job with Computerland (for which he had to buy an APPLE II) and had a second-hand 16K System 80 and a green screen monitor for sale. I went along for a demo out of curiosity, but when I saw that you could play Space Invaders AND write your own BASIC programs, I was convinced that this was the perfect gadget for me -- I could learn more about computers and programming, and spend all my idle hours playing games.

So later that week I became the proud owner of a microcomputer, the envy of all my friends, and sleepless from playing Formula One Racer until half past two in the morning. For the next six months, I didn't do much else with my new toy -- aside from acquiring a few more games as my little brother kept beating my best scores on each one. I did start buying the occasional computer magazine (Micro 80, Australian Personal Computer and Your Computer had only recently started up) to see what new games were around and to type in any useful BASIC programs.

After the first six months of enthusiasm, the novelty had worn off. I still played the occasional game -- the first Scott Adams Adventure kept me going for a while -- but I didn't really use my computer for anything aside from sporadic entertainment. After about a year of this, I decided that I should either get rid of it or somehow get more out of it -- so I joined a user group (the Computer Owner's Group). I soon expanded my collection of games and got interested in disk systems -- mainly because I got sick and tired of slow and unreliable tape storage.

The next time Dick Smith dropped their price on Expansion Interfaces, I splurged and bought one with 32K of memory, and then lashed out and bought a 40 track, single sided disk drive. I then spent the next six months struggling with NEWDOS80 and wondering what on Earth was going on when it wouldn't do what I wanted it to. However, after numerous disasters, I eventually sorted out how to get most things done and poked along quite happily for a while.

The real turning point for me came in 1981, when dot matrix printers first broke the \$1000 price barrier. All this time my long-suffering mother had been typing my QIT assignments, so I suggested that we share the cost of a dot matrix printer and thereby save her all this work (of course, I was only thinking of her). We eventually bought an Amust 80DT -- the first dot matrix with true descenders and multiple fonts for less than \$700. It was then that I really started to USE my computer, and I've never looked back since. The constant grind of assignments meant that I was spending more time on my computer, and because I was more dependent on it, I really needed to sort out all the major functions properly -- near enough was no longer good enough.

Another problem that rapidly emerged was the shortage of disk space. Because I only had a single disk drive, I had to keep a subset of NEWDOS80 on each disk. By the time I put applications programs on there as well, there wasn't a lot of room left for data files -- particularly word processing text files. So when an opportunity to buy another two second hand disk drives (one of which was double-sided) arose, I took it. This had the advantage of enabling me to have single-sided system disks with NEWDOS80 in Drive 0 and double sided data disks in Drive 1. The disadvantage was that it forced me to learn about the PDRIVE command. All the understanding I thought I had seemed to evaporate as I cursed PDRIVES and whoever created them.

Eventually, however, I managed to get things working and proceeded to re-organise all my single-sided disks onto double-sided ones. By this time I

had accumulated enough software so as not to be able to find any of it when I wanted it, so I started to get a bit more systematic about what I was putting on each disk. It was about this time that I also joined the TRS80/SYSTEM80 User Group, as I felt that it had a lot more to offer than the Computer Owner's Group -- which was trying to be all things to all men and not doing any of it very well.

By this time I had finished QIT and started another course at the University of Queensland. Various subjects required use of the mainframe out there, and it was difficult to get on a terminal. I'd heard about modems, but didn't know much about them -- only enough to know that there was no real reason I shouldn't be able to use one from home instead of going to Uni to access the mainframe. Once more, I lashed out and bought an RS-232 board for my Expansion Interface and the original Dick Smith Dataphone. After a month of trying everything I could think of, I finally got it to work - and it did actually save me a lot of hassle.

During this time, I'd changed jobs and was fairly heavily involved with FORTRAN programming on one of the State Government Computer Centre mainframes. I found that my experience with my micro was a big advantage as it gave me a much better understanding of the operating system and computing generally. However, since I was doing so much programming at work, I lost most of my interest in BASIC programming on my micro and generally stuck to packages instead of writing my own software.

Once again my software collection had expanded to the point where I was running low on disk space. I'd heard about double density, but thought I had to have special disk drives to use it and I wasn't about to replace all my drives. Eventually I found out that that MOST drives would support it and that I could probably use most of my single density disks as well. So, once more, I lashed out (although it was actually pretty cheap) and bought and installed a doubler -- and once more started cursing PDRIVES.

I think that this was really the time that I realised that I had passed beyond the conventional limits of what a user is 'expected' to know. I no longer believed everything I read (or was told) because I found that I often knew better. I think that there are a number of reasons I got to this stage. First and foremost, I had a real need for word processing, communications, and mail list facilities - it wasn't just a hobby. Because of this, it was worth my while to persist and really get to know my system properly. Secondly, as a hobbyist, I was interested in getting the most out of my machine that I possibly could.

Because I'd started at the base level and expanded progressively, I had a pretty solid understanding of each level before I went up to the next one. This meant that I didn't have to assimilate too much at once and had a clear understanding of all the basic components. Thirdly, through the Club, I had access to the facilities I needed - i.e. people that knew more about computers than I did - to put me straight when I got into trouble or couldn't figure out something for myself.

So nowadays, having achieved one of my objectives (making the most of my computer), I look at computing more from the point of view of helping others through the same process I've been through. It IS horribly confusing when you have to sort out something you haven't encountered before, and a bit of advice at the right time from someone who has 'been there, done that' can save a lot of frustration.

I suppose the message to other Club members is that if you've got a computer, it's worth your while to make the most of it. You do that by seriously looking at your need for computing, acquiring the necessary hardware and software to meet that need, and persevere in learning how to use it properly. DON'T give up when something doesn't work the way it's supposed to; bring it up at Club meetings and get it sorted out -- the odds are that someone has solved the problem before; and even if it's a new one, the Club usually has the resources to crack it. The Club services are there - USE THEM!

Happy Computing, Bill Stomfai

-----<BNB>-----
ESCAPE FROM ASYLUM -- PART 5

by Peter Evans, Canberra Group

Have you been working through ASYLUM using the series of articles Peter Evans? If so, you must have been getting frustrated having to wait two months this time to get the next dose. However, as a reward for patience, here it is:

Now you are getting close to escaping from this madhouse. The copper key grants access to only one new room, but this one room is very important. In it is a man who offers a wire hanger, but there doesn't seem to be any way to make him listen to you. Try offering cigarettes, and if this doesn't work TRADE a cigarette FOR the hanger. The inmate will disappear and you will have the hanger. Examine it. You should have got the passkey by now. Is there something that has been missed? Try using the wire hanger to unlock some of the doors that have defied your attempts so far. It doesn't work. Therefore that inmate must have something else as well as the hanger, because the instructions say "All drawn figures are characters that you must deal with in order to escape." Maybe he had the passkey. Go back to your last save and try getting the passkey out of him.

At last the coveted passkey is yours. Now go and try all the doors you couldn't open before. When going into the guards area, don't forget to wear the uniform. What does it say on the desk? Don't sit on the fool's bed. Who is the fool? The person who is foolishly giggling, of course. Go and sit on his bed to see what happens. Since he won't allow you to do this, you must think of a way to get rid of him. Have you looked at the gold? It didn't fool you but it might trick a fool. Now try sitting on the bed again.

-----<BNB>-----
\$TRINGS AND THINGS (ALL MODELS)

by Bill Allen

Although our type of computer is fairly pedestrian when it comes to number crunching -- compared to much more expensive purpose-designed types -- they can do an adequate job for most purposes with good software.

Graphics are very rudimentary, but are not really important in most of our applications and the plethora of games for diversion from serious matters amuse us enough without worrying about inadequacies compared to games-orientated machines. Our machines were really designed more for business than most of the competition. If all you'd wanted to do was play games, then you'd have bought a "games" machine long ago. I solved the graphics problem in my 4P (for CAD etc purposes) by adding the optional hi-res board (640x240), which is quite adequate.

However, the real power of our type of machine is in string handling. They can hold their heads high in any company in this field. String handling means data handling, so it is a relevant exercise in programming to write your own special applications in BASIC on them for your data handling needs.

A long time before I actually bought a computer, I read quite a lot of computer magazines to see what computing was all about. One point I picked up very early on was that one of the main differences (apart from the obvious ones) between Personal Computers and Main Frame Computers was in string handling. The main frames could do so much more with strings, I read. This little snippet was stuck in the back of my mind and resurfaced when I later got to know Warwick Sands and his work of extending Basic functions.

When you look at the added string functions in Warwick's Custom Basic, and the TOS to a lesser extent, you'll note that there are quite a few extensions to string functions on top of a very good list already. Some of this even came from my requesting Warwick to satisfy a need I conjured up when writing a program to handle a club members records. So what I am just trying to do here is to get across the importance of string handling, as I see it, and some of my experiences with the subject.

One of the functions of the program I wrote was to swap characters in elements in a string array. I originally wrote my own USR routine when I was using "MU" type files for a rose club members records handler. The records were of varying length, as permitted by the "MU" format. One need was to swap the last four characters (which happened to be the postcode) in each record with the first four temporarily to do a postcode sort (with \$SORT or CMD"O"), because at the time the records were few enough to hold the complete file in a string array. The USR was called to roar through the array and do this swap. It took about 1/4 sec. with about 400 elements, the sort took about 3 sec., then the USR was used to restore the data to its normal form. So the complete process took about 3.5 sec.-- very satisfactory speed, I found. The array was now ready to print posting labels in postcode order. This was a most important function of this program and saved much time preparing mail for bulk posting. The idea of this USR was used to develop \$SWP in Custom Basic, where you can swap characters in strings even more flexibly -- and this is only one of the many added string functions.

That particular program finally outlived its usefulness when the (rose) club I'd developed the program for grew to well over 500 members and I started to run out of memory (in the computer!). I had to completely change the method and operate more from disk directly with "FF" files, keeping only an 8-character index array in memory itself.

This gives me the capability of handling over 1000 members now. It should do me for quite a while in that form, but if pushed beyond this point I have ideas which hopefully may lead to 6500 records or so capability without sacrificing speed -- apart from the obviously longer sort. Several derivatives of this program are used to maintain our computer club's records and print the various lists therefrom you'll see floating about.

If I can get enough time to write the code, I've cheekily promised myself sometime to write a Creator or Producer type program, where one can design the screen/input etc (only with much more scope than the others, I hope) to suit a specific data handling application. This would use the kernel of my present program, which has operated satisfactorily for me for several years now.

I wouldn't like to attempt to write it down to standard Microsoft Basic, because of its use of Custom Basic's extra string handling features and added program structure functions. It is using NewDos86's added features as hard as it can go -- one reason was to cut down on total Basic program space to leave plenty for string space; another was to give Custom Basic a good workout before it hit the deck and I now find it (Custom Basic) hard to live without. I believe in driving the computer up to its full hardware capabilities -- not down to the lesser levels of basic models and DOS's on the grounds of "compatibility". Why drive around in first gear all the time when the machine can do much better than that?

Some of Custom Basic's added commands save space because there is much less interpreting (increasing execution speed). The extra memory taken by the larger Basic/cmd is thus paid back in shorter coding needed. The only program I've found to foul up with the restriction of space in NewDos86 is DOTWRITER version 3 (written in BASIC). This is really irrelevant because version 4 is in machine language anyway and is so much better in many ways than version 3 and certainly gives no trouble under NewDos86.

This brings me to another item (mentioned in Custom Basic Manual): There are several other machine language programs that need NewDos86 to be initialised without the high memory enhancements so those programs will work properly.

My perspective of this problem is that although Warwick has the mechanism to handle it, the real culprits are the programmers of these few applications programs. LDOS will run into the same problem at times, as it uses high memory as well. Anyone writing a machine language applications program MUST have a mechanism to honour the HIMEM pointer and adjust its buffer size or relocate code or whatever to suit the memory conditions. The DOSes maintain the HIMEM pointer. Were the programmers too lazy to write a little extra code to honour it or did they need all of the high memory?

Some of the very people who plumped for "compatibility" failed badly when they didn't honour HIMEM. This (to my mind) is a very slack practice and, fortunately, very few programs actually exhibit this problem. But we must recognise it exists occasionally. If a program falls over under NewDos86, you must re-initialise (BOOT or RESET) with the SHIFT key held down (CLEAR or F2 in the 4P) thus leaving the high memory free to accommodate the program's naughty little habits. This is no great loss to the user, as the high memory enhancements are mainly for programming and disk housekeeping aids and thus are not really needed at runtime.

The feature which you WILL MISS, however, if you don't have the high memory enhancements, is the automatic disk format determination routine (the one which dispenses with most PDRIVE problems). This background task will ensure the applications program will be able to read/write to its data disk, even if the PDRIVES are not set properly, provided that there's no impossible situation like an 80 track disk in 40 track drive etc.

The lesson in this is: If you start to write your own machine language programs and they use as much memory as is available or use quite a lot of memory, you must not forget to add the test during initialisation to read the HIMEM value and adjust the program's pointers accordingly or just test against the pointer to see when the program's buffer is running out of memory -- whatever suits the occasion -- and you'll steer clear of this problem under any Dos.

-----<BNB>-----

PROGRESS MADE ON HARD DISK / NEWDOS86

by Warwick Sands

I hope that you have enjoyed the holiday period and are feeling refreshed and ready to face another year. I spent three weeks on the Mid-coast of N.S.W. and enjoyed a good holiday. On the way home, Bill Allen kindly extended me his hospitality and we managed to successfully tackle the hard disk version of ND86. This is a progress report of what was discovered during that time. I must also profusely thank Bill Stomfai for the loan of his 15-Mbyte hard drive for development purposes -- it allowed us to have two 4Ps set up with hard disks at the one time and this speeded progress no end.

I had hoped to be able to 'fix' NewDos86's hard disk formatting program (HDFMTAPP/CMD) to cope with the 15-Mbyte drives. This wasn't as simple as it first appeared and no great progress was made, due to the lack of meaningful information on the WD1010 Hard disk controller chip. Peter Goed is going to try get the missing information; until then we will have to leave the matter in abeyance.

On a more positive note, we did manage to get NewDos86 to boot in the standard fashion on the hard disk system. The automatic disk format determination routine now works correctly. As it now stands, all of the ND86 enhancements are now available to hard disk users with the exception of the enhancements to the COPY command. They will follow shortly.

Additionally, we proved that we could make a self-booting disk for the Model 4P that would automatically load the MODELA/III file without having to go through various gyrations to fool the ROM Loader. We also managed to get Glen McDiarmid's excellent program, Time Machine, to run on the Model 4. Before the week's end I hope to be able to provide a mechanism whereby we can get the Model 4 to act like a 4P and automatically load the MODELA/III file instead of switching in the Model III ROM in the hardware.

One aspect of hard drive operation was decided upon. We are going to divorce the user from the PDRIVE concept as much as possible. Instead, a MOUNT command is going to be provided. It will work in the following manner:-

A BASIC program will allow you to create logical partitions (or Volumes) on your hard drive. This information will then be permanently stored on the hard drive. You give each partition a unique Volume name. Whenever you wish to access that partition, you use the MOUNT command. E.g:-

MOUNT FRED, 3

will cause the partition/volume called FRED to be mounted as drive 3. MOUNT can store up to 256 partitions. There are 5 default volume names provided, they are: FLOPPY0 - FLOPPY3 these allow access to the floppy drives; CDEFAULT is used to set up the drive configuration on the SYSTEM at boot.

The C in CDEFAULT stands for Configuration. MOUNT allows you to save your logical drive assignments as a VOLUME name and then reinstate them with a single MOUNT command.

The full implementation of MOUNT should be available sometime in February '88.

A club member took me to task for writing articles that are beyond the ken of 70%-80% of the club members, e.g. the article on MAKELIB.

[Well, that's ONE person's ASSUMPTION that 70-80% are as lacking as much computer knowledge as himself. This is NOT the type of feedback we've been getting at the meetings. (Also see following article) -- Ed.]

I must plead guilty, but in my defence this style of article was designed for the somewhat more advanced programmer. The more people that are familiar with the 'internal' workings of ND86, the greater the chance that there will be other public domain programmes written for you and I. However, in the future I will try to explain the 'jargon' that I use. My aim is to educate, so PLEASE LET ME KNOW if something isn't explained enough for you.

To bring my feet back on the ground, I have written a BASIC program that repairs the GAT table. It still needs some spit 'n' polish, but it should be in the next newsletter. As well as providing a useful utility, I hope that you may gain some insight into how the directory works.

Enough chatter from me. To finish off, here is a small (warts and all) CUSTOM BASIC program that allows you to copy sectors from one file to another.

```
10 CLEAR 5000:DEFINT A-Z           'Set up the defaults
20 OPEN "R",1,"file1"             'Source file
30 OPEN "R",2,"file2"             'Destination file
40 I=1:WHILE NOT EOF(1)           'Copy the whole file
50 GET 1,I
60 #MOV FCB(1)+32,FCB(2)+32,256   'FCB(1) looks at file1 sector
                                   'FCB(2) looks at file2 sector
                                   '#MOV copies file1 sector to file2
70 PUT 2,I                         'Write the sector into file2
80 INC I                           'Get the next sector
90 WEND                             'Do the full file
100 CLOSE                          'Finish up properly and
110 END
```

See if you can see how this could be adapted to allow a file with several unreadable sectors in it to be copied? Answers next month.

Happy Hacking,

Wak

-----<BNB>-----

THE KNOWLEDGE GAP OR LIFE ISN'T ALWAYS EASY

by the Editor

We get the sort of NEGATIVE feedback from time to time (from only a few people, fortunately) as mentioned in Warwick's article above -- mainly from those who either haven't done enough study on the basic computer manuals or they don't even have them. But these NEVER state exactly what they DO EXPECT. Well, I'll restate what has been said many times in the past. The newsletter can't hope to reproduce the information in the manuals -- it'd cost too much. It simply consists of articles written by people who HAVE STUDIED the manuals and put the knowledge gained to practical use and are KIND ENOUGH to make the effort to SHARE THEIR KNOWLEDGE and experiences with us. Some of the material is highly advanced, but certainly not all. One has to assume some sort of computer knowledge and willingness on the reader's part to make progress by the study of these advanced articles.

It has always been understood that ALL the info in the articles may not be relevant to EVERYONE at a particular time, but is a record to be retained and

understood and used WHEN THE READER IS READY FOR IT. Should we ban assembly language from the newsletter just because some don't study it at all? Of course not! The utility derived from MAKELIB/ASM is EXCEEDINGLY SIMPLE TO USE and can be had all ready to use by requesting same from me (this was stated plainly enough in the article, I thought) -- you don't have to torture yourself if you don't understand assembly language -- just let it ride.

Maybe it's time we had a series of simple introductory lectures on THAT subject, but then we DO have taped lessons on the subject in the Library and you will certainly learn better about this subject in the privacy of your own home. These lessons are excellent -- they start with the simplest basics of the computer and feed you the info at a nice steady pace. You can go back over the ground again and again if it's slow getting through to you.

This is not to say that simple articles are not wanted for the newsletter -- by all means let us have your little routines or lurks and kinks discovered in applications programs and tell us of your experiences, even when you are a tyro. I always did it in the early days as soon as I made some little discovery (well, to me it was a discovery!). I recorded it and submitted it to the then Editor just in case it would be useful info to someone else, but I never for one moment imagined that it would be useful to EVERYBODY.

On another aspect, there are not many VERY SIMPLE articles printed at present simply because you members are not submitting them! I'd also particularly welcome requests for articles on specific topics and we'd see what we can conjure up between us if we could get some helpful feedback. I don't consider a bland statement that 70-80% can't understand the articles any help at all! I am forced to think that anyone making this sort of comment isn't even trying to READ (and REREAD) these articles, but just GLANCING at parts of them and then skipping the rest, thereby missing the simple punch line.

While most members are sitting back keeping mum, it's extremely hard to know what's really wanted generally! I'd love to be supplied with a nice long wish list -- it would help my editorial operations no end -- so don't be shy, I don't 'byte' hard and not at all to positive feedback. One nice bit of feedback from the other side of the world was that our newsletter is considered one of the best in the world in this category, so somebody likes us out there.

If you carefully study the index supplied with this issue, you'll find that we've covered a lot of ground since the club started and I suspect many of answers to your particular knowledge gap are in the back numbers. We still have the masters of all the newsletters, so any can be reproduced on request -- so contact Jim Melksham for your requirements.

One genuine problem I've noticed -- in the case where new members started with the Model 4/4p as their first computer. Reading through the manual supplied with the computer, I gather that the writer(s) seemed to have made an erroneous assumption that all purchasers would have graduated from the Model I/III etc environment. This fact may not necessarily be evident to the tyro. So a lot of groundwork information definitely is lacking in the Model 4 manuals. Therefore the people in this category wishing to overcome this deficiency should buy, beg or borrow such books as: LEVEL II BASIC REFERENCE MANUAL (Cat # 269-2102 at Tandys -- it's VERY cheap and a goldmine of help), BASIC FASTER & BETTER (THE Basic bible for our machines -- I still have a couple left @ \$15 but you'll have to pay a heap more at Dicky Smith's & Tandy if they have them still).

Those who are lacking in the necessary manuals and can't purchase them can find most of them to borrow from the club Library (See John Bird). Failing that, they should SPEAK UP to see if they can borrow or buy them from another member, then take the books seriously and make an effort to learn the lessons therein. It's completely up to the individual -- NOBODY CAN DO YOUR LEARNING

FOR YOU!

So I am bouncing the ball back in your court. I'd hate to think if I had to put in a bill for the midnight and early morning hours spent getting the edition out (after the phone has quietened down for the night). If the editor got nothing but negative comments for all the pains, then you would soon have no editor and no newsletter and probably soon after -- no club. No one would bother taking it on. But I am sure that this would never happen in fact, because we DO get some good messages back -- especially from the distant members.

I always have considered the computer as a great teacher. The more hours spent on it, the more you will learn. I realise that probably only 10% or so of users (in my experience) in the past have tackled even BASIC programming seriously, but my aim is to provide as much material as possible for this aspect and hopefully get many more members highly motivated in this direction to provide an excellent outlet for their creative abilities. It is the most rewarding aspect of our hobby. Just being a keyboard operator is what one would mainly expect in the workplace and is never enough to stimulate long-term interest.

-----<BNB>-----

SETDATE / CMD -- MODEL III & MODEL 4/4P IN MODEL III MODE

by Bill Allen

Those who have a meaningful use for dates in their applications programs have a number of choices to cater for their need:

- 1) Have the operating system prompt for the Date-Time when doing the initial cold startup, the normal setup in an unmodified computer.
- 2) Do a hardware mod. by installing a real time clock -- such as the project by Jakub Szajman published in BNB # 55.
- 3) Turn the Newdos (80 or 86) prompt off (SYSTEM 0,AY=N) and use SETDATE as an AUTO function on BOOT (AUTO SETDATE).

What's Setdate? It is a machine language utility that comes on NewDos86 as a bonus anyway and we originally found it in Northern Bytes as a public domain program for Newdos/80. It comes up with the date on which it was last used. The operator then uses the right arrow to advance the date to the current one and then presses <enter> to record the new date on disk (it is already stored in memory). If the operator overshoots, the left arrow key will back it up to the correct date. Now all that is simple enough. If you haven't got it, then you'll either go for broke and properly update your DOS by getting NewDos86 or get SETDATE from the public domain programs in the Library. Whichever way, I assure you that you'll love this little utility.

It does, however, lack the time-of-day function. Hopefully, someone will add this to the original source code (yes, the source is also available) one of these days or the operator can manually type in the time after the SETDATE function is completed.

TIME, hh:mm:ss<enter>

Some people have either modified or acquired a modified version of Newdos that puts the date in the order that Australians normally think -- i.e. DAY/MONTH/YEAR. The American method as used by all DOS's is MONTH/DAY/YEAR. As I soon got used to that order, I have never worried about changing it. In any case, SETDATE insulates you from this apparent aberration. I guess, though, that it will come into conflict with any "modified date" DOS, so make sure that you are using an American format DOS when you install SETDATE.

-----<BNB>-----

ADDING EXTERNAL DRIVES TO EARLY 4P + CHANGEOVER SWITCH

by Bill Allen

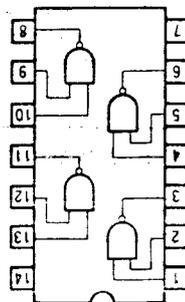
I thought maybe some country or other members of ours might like to do this mod., but did not have the info. I've tried to spell it out in great detail and not assume access by the reader to too much technical info but one needs to have a small wattage soldering iron suited to TTL/pc board work and a little expertise in handling the small scale work. Too high a wattage can cause the glue holding the copper pads and tracks to the board to break down as well as damage the chips.

The technical info for this mod to the EARLY TYPE 4P (up and down arrow keys on the left of the keyboard) came from an article in the Sydtrug Newsletter some while back. There is another, surer, way to identify the earlier model. Apparently when the changeover to the gate array model was taking place, they ran out of the old type keyboards (which I prefer) and fitted the new keyboards (arrows in a cluster) to a number of the old models. How you positively identify the version is that the earlier type has METAL clips on the side to secure the carry cover and the later gate array version has PLASTIC clips. To fit external drives to the gate array version (plastic clips), see P. Goed's article in Bits & Bytes # 38.

Parts required for Mod to early type 4P:

1 Double pole double throw switch
1 74LS32
1 36-way D connector (female) w/ribbon cable compression type cable attachment
1 36-way D connector (male) w/ribbon cable compression type cable attachment
4 34-way ribbon cable-to-edge-card compression connectors
1 34-way header-to-ribbon-cable compression connector
about 1+1/2 metres 34-way ribbon cable
Quantity fine insulated hook up wire to suit
Drives to suit (your pocket)

First we must establish the numbering convention for DIL (dual in line) chip pins. Looking from the top of the chip, locate the notch or spot or whatever mark the manufacturers have made on it to identify the reference end. With the mark towards you, pin 1 is the one nearest to you in the right row. The other pins are numbered sequentially, working anti-clockwise, so the highest number will be the pin nearest you on the left row (see diagram):



Having established that convention, let's tackle the job itself:

1) Disassemble computer and expose top of main pc board. I assume the rear of the computer pc board is towards you. Note at this point that all the

notches/dots on the chips on the board are towards you.

- 2) Locate U15. The 74LS32 will be piggy-backed on U15. We will now call this 74LS32 U15A.
- 3) Bend all pins except 7 and 14 on U15A outwards to clear the pins on U15 safely. Do this carefully and gently so as not to annoy the metal pins too much.
- 4) Join pins 13-12-11-10-9 of U15A together with wire, carrying the end from pin 9 over to pin 7 (not bent) and solder to the outside of pin 7.
- 5) Join pins 3 & 4 of U15A together.
- 6) Now solder U15A into place on U15 (with the notch/dot in the correct orientation with the one on U15 under it) soldering the pins 7 and 14 of U15A to pins 7 and 14 on U15 respectively. Carefully check that no other pins on U15A are in contact with those on U15. You now have connected U15A to the 5 volt and Ground rails.
- 7) Connect pin 6 of U15A to pin 32 of U13 (WD1793, the FDC). This will need only about 2cm of wire. Be very careful to identify pin 32, starting from 21 at back left row and counting towards you.
- 8) Connect pin 5 of U15A to U74 pin 12. This will take about 10cm wire.
- 9) Connect pin 1 of U15A to pin 11 of U20. This will take about 13cm wire.
- 10) Connect pin 2 of U15A to pin 9 of U20. This will take about 13cm wire. Pin 8 of U15A has no connection. Just leave it float.

We have now finished modifying the top of the pc board. Carefully check the previous steps at this stage before proceeding further.

11) Now we have to prepare the switch. First get the orientation. We are at the rear of the pc board. The switch lever is towards us. The switch is on its flat (i.e. there are two horizontal rows of three terminals at its rear. The locating groove on the mounting shaft is on the left side. Now we will number the terminals. Starting on the left top terminal from our aspect, number the terminals thus (I'm imagining the switch is transparent in this orientation, as the terminals are on the far side.):

1	3	5
locating groove -->		
2	4	6

Now connect terminals 1 to 6 and terminals 2 to 5 with 2 short jumper wires.

Attach about 15cm leads individually to terminals 1, 2, 3, 4.

12) This switch I've mounted on the rear plate which covers the Modem slot in the computer. I drilled a 1/4" hole (or whatever suits the switch mounting size) level with the screws mounting this plate and centrally located above the RS232 25-way D outlet. The small locating hole for the tab on the locating washer will be drilled on the left (outboard side) of the mounting hole. If your orientation is correct, when the switch lever is pointing to the inboard direction, the normal Drive 0 will be the actual Drive 0 (standard configuration) and when the switch lever is pointing outboard, the chosen external drive will be drive 0. In that configuration the normal Drive 0 will become the number of the external drive taking over its role. I've put a bit into this switch orientation so that it's easy to remember where you are.

because this switch is going to be toggled blind by your reaching behind the computer and under the rear flap. My aim in this mounting scheme is not to make any changes visible externally so as not to mar the very neat finish of this computer.

13) Now let's tackle the rear plate for mounting the female 36-way D connector. I've mounted this on the left side using the screw hole for the above-mentioned Modem cover for the right side of the plug and drilled a hole to suit the other side of the plug. I won't give an exact measure as there may be a slight difference between brands of plug. Then a rectangular window is cut in the plate between these mounting holes just large enough to accommodate this fitting midway through the plate. Arris the sharp edges so that the ribbon cable is not abraded if it comes into contact with the edges during final installation of the plug and cable.

14) We can now finish the major surgery. We now have to remove some of the side of the rear recess in the main plastic case. This removal will not be visible when the case is closed and is necessary to allow the male D 36-way plug to be inserted in its opposite number to connect the external drives. Make sure you locate the correct side and cut vertically about halfway from the top of this "cheek" (as I will call it). Stop as soon as you are level with the inside of the outer shell of the case. Then start a cut from the top of this cheek with the blade resting flat against the inside surface of the outer shell and cut to meet the the end of the first cut. Clean up the cut and arris any sharp edges and the "butchering" is over.

15) Ok, now back to the delicate work. We are now going to operate on the underside of the pc board. Remove the plug for the keyboard cable, so that the board may be lifted later. Remove the holding screws, 9 I think. Once free, carefully manoeuvre the board, hinging it upwards from its mounting and being very careful to allow the three plugs at the bottom (front of the computer not to foul the case above them and strain the relationship. When partly up, you'll have to identify the location of the pins for four chips under the board. Be most careful about this and BE VERY SURE you have got the pin numbers right.

First we have to locate U74's pins under the board. When you have located that double row underneath (and facing the underside), pin 1 would be the upper one in the right row. We have to cut a track now. It is the one leading from pin 3 of U74. This track just passes to the right of pin 1 going upwards and finally dives into a through hole in the board. I cut mine just past where it goes past pin 1. As there is another track right next to it here, you MUST BE VERY CAREFUL NOT TO DAMAGE the other track. A rough idea where this cut is -- it's about 2.5cm to the right of the centre left mounting screw hole and level with it. Ok, we are finished with U74.

16) Locate J5, where the drives ribbon cable plugs into the board, then view it from the underneath. The lower row, counting from the left, goes 2, 4, 6, 8, 10, 12, 14, etc. We have to cut a track leading from pin 10 of J5 to pin 2 of U20. U20 is the chip on the upper left as viewed from under the board. This track makes a right angle bend in the open, clear of other tracks about halfway -- and this is where I cut it. This cut will now be bridged by the switch. Solder the lead from terminal 1 on the switch to pin 10 of J5. Solder the lead from terminal 3 of the switch to pin 2 of U20 (2nd from the top on the right row as viewed from under).

At this point a choice has to be made as to which external drive will be switched with Drive 0. As I mounted the two external drives immediately, I opted for Drive 3. If you are initially installing the one drive, naturally

you must opt for Drive 2. So I will describe the two methods and notate them thus: 17.2) means the switched drive will be #2 and 17.3) means that the switched drive will be #3. DO ONLY ONE of those steps.

17.2) Join switch terminal 2 to pin 14 of J5 (drive 2 enable). Join switch terminal 4 to pin 8 of U20. Join pin 6 of J5 (drive 3 enable) to pin 10 of U20.

O R

17.3) Join switch terminal 2 to pin 6 of J5 (drive 3 enable). Join switch terminal 4 to pin 10 of U20. Join pin 14 of J5 to pin 8 of U20.

18) Locate U32 (just to the right of the 1793 FDC). Under the board join pin 7 of U32 (second from the bottom on the right row as viewed from under) to pin 9 of U20. Join pin 15 of U32 (second from the bottom of the left row) to pin 11 of U20. About 12cm wire each is needed to bridge these.

19) Reassemble the pc board on its mounting, plug in the keyboard cable and replace the screws, taking care to route the switch leads away from being pinched between the board and the screw platform (second from left). The screw in the far right corner has the earth strap terminal and can be awkward, but just VERY patiently deal with it and it will co-operate. Reassemble the pc board and its mount in the housing, taking care to give the switch leads a neat path.

20) Make up the external cable, making sure that the COLOUR MARKED (usually red) lead is located in pin 1 position (it's marked on the plug) on the male D plug and the cable goes to the plug from the bottom direction (the upper side is the broader side of the D). As we are fitting a 34-way cable to a 36-way plug, there will be two pins not used, so we must ensure that the red lead is level with the first position on that side and the two-lead gap is on the other side of the cable. Position the two 34-way edge connectors along the cable to suit, making sure that the red lead goes to the pin 1 marked. If the drives have a locating slot in the edge card (this is usual) and the plugs have the key fitted, the red lead will be on the slot or key side. If there aren't keys fitted, I suggest you obtain some and fit them yourself (makes life easier when plugging in drives -- you CAN'T get it wrong).

21) Now a modified internal cable must be made. As Tandy remove pins from their edge card connector plugs to program the drives, you can (if you are astute and careful) save yourself a little cash and trouble if you can remove these plugs from the original cable without destroying the clips -- lotsaruk. If you make up the cable with new complete plugs, you will have to remove the internal drives to program their shunt headers (Tandon TM50-1 drives). Be sure to get the notch in the correct orientation (matching that on the socket) when replacing them:

TM50-1 SHUNTS		(End Notch/Mark up and viewed from above)	
Drive 0		Drive 1	
1 not used	16	1 not used	16
2--connect----	15	2 disconnect	15
3 disconnect	14	3--connect----	14
4 disconnect	13	4 disconnect	13
5 disconnect	12	5 disconnect	12
6 disconnect	11	6 disconnect	11
7 not used	10	7 not used	10
8--connect-----	9	8--connect-----	9
			Motor On

You can position the plugs on the new internal cable now, starting with header (check that the red lead will go to position 1 as marked on the pc board at J5. You can increase the space to the drive plugs very slightly to help manoeuvring later. The first edge card plug goes to drive 1, then it loops over to drive 0. When you have progressed this far, plug it in to the assembly to see how much cable lead you need to carry on to the female D connector mounted in place. Again allow a little extra for manoeuvring, but not too much -- as it's a confined space and the air has to get through to the fan. Mark the projected position and remove the cable to affix the D connector. Make sure again that the red lead goes to pin 1 and the two-pin gap is on the other side and that the large side of the D is upwards. This time the cable should ideally meet the plug from above, because the header plug will be very close beneath and behind the D connector when they are mounted. If you've got the lengths right, now cut off the surplus cable on the other side of the D connector and assemble. First plug in the header to J5, then manoeuvre the D connector through the window cut in the rear plate (you'll have to fold the cable at 45 degrees to do this) and fit the mounting screws. Fit the switch in its hole and replace the plate. Then fit the cable plugs into their respective drives (I take it that they are fully mounted after you programmed the shunts).

Also you will have to ensure the termination resistor pack is installed in the LAST drive in the external cable (some new drives have the resistors built in and you select/deselect them with a battery of jumpers. Tandy normally don't use termination for two drives but you need one when you fit extra drives on the daisy-chain cable. The normal place for this termination in most drives is a socket near the card edge where the external cable plugs in. In the case of MPI drives, the socket nearest to the card edge is for the jumper (or dip switch) programming of the drive and the termination is placed in the socket behind it, with the terminator pins occupying the outersmost holes. The socket has 2 more holes than the pins on the terminator pack.

At this point it would be wise to (CAREFULLY) plug in the power cord and test the computer in its normal configuration (with the switch lever pointing inboard), before final reassembly of the case. If all is well, plug in the external cable and drives and establish that you can address the external drive(s). Don't forget to program THEIR shunts -- in the case of the 4P, they are shunted as Drive 2 and 3, NOT as the Model 4 and Model 3, which have the external drives shunted as 0 and 1. Beats me why these strange differences came about at Tandyville. If the ext. drives start up ok when addressed, try the switch to see if the selected new system drive is addressed (select light on) when you press Reset. If all is well, then power down, remove the ext. cable and finish off reassembly of the case. Refix the external cable, manoeuvring it into the gap cut in the "cheek" at the end of the case well and secure at least the inner clip. The outer one is not really esay to reach, but can be pulled into position with a paper clip straightened and a tiny hook bent into one end.

If you want to restore portability at any time, all you have to do is switch the switch to the inner side and push the cable plug's clips loose and manoeuvre the plug out and voila -- back to standard configuration of 2 internal drives. If you fit 2 80 track double sided drives to the system as I did, you wonder why the manufacturers didn't do so as standard because of the power and extra capability it gives the computer. There's no problem in addressing these extra drives with Newdos (by changing System option AL=whatever number of drives in the system and setting the Pdrive slots for the added drives to suit their type) and TRSDOS 6.x or LS-DOS 6.3 can be configured to use them also with no great drama.

-----<BNB>-----