PENNSYLVANIA CONNECTION

by: Elizabeth Deal

MORE ABOUT VERSIONS OF B-COMPUTERS

It's becoming increasingly obvious that there were several B128 versions released. The latest version is the one most people have. That includes England and the people who wrote Superscript (when the issue of keybounce came around, Precision asked to check what ROMs we have, they wanted to see version 1). Via the grapevine I'm learning that in B computers previous to version 1 the SYS command didn't work if the routine was in a non-system bank. Recently I've learned from one of the B users that on his B machine BLOAD is different and BASIC ROM code doesn't match what most people have. Paddress is mandatory, whereas it's optional on the current Bs. Four people I know of have trouble running a lot of software. At first we Norman and I suspected bad disk duplication, but we are pretty sure now that ROM versions can well play a part. So I've looked at some dumps people sent in and I think that the version number (that's what I call it) in byte $fff9 can be relied on. I've isolated these values from the dumps:

```
           v#              irq
           --              ---
ffff8  60  aa  77 fb ea f9 13 fc    -old b500
ffff8  60  00  3d fb 9e f9 e5 fb    -old b128??
ffff8  60  01  31 fb 97 f9 d6 fb    <-- common
                                    b128/256
```

BANK 15:PRINT PEEK(65529) tells the version. $AA (170) was likely an early system, un-numbered. Then the zero version and finally the current, most common version 1. The current B256 machines are identical to B128, both are version 1 as far as the KERNEL routines go. Basics differ, but we know where. Many programs which run on the B128 also run on B256 unless their machine code uses a lot of fixed position BASIC routines. I know that of my programs KEYTRIX and SUPERMON run in B256 v1, but COPY ALL does not (due to their variables scattered in 3 banks!). Others may also work, I haven't tried them, I don't have a B256 computer.

In any case, if your peeking the version byte does not return #1, you're in a soup. It borders on economic foolishness to program anything for the B machine. It would be an economic suicide to program anything for the few orphans of the B family. I think that the only solution is that suggested by our leader (that name is beginning to stick, eh?) - ROM replacement. Norman has a permission from CBM to burn ROMs. If you'd rather do it yourself, one of my previous disks contained B128 and B256 dumps. Those dumps were short by one byte (one at $ffff doesn't save) so you'll have to append it: it's the last byte you see in the table above, $FB. The last pair of bytes in the machine is the IRQ vector, it's $FBD6 in version 1 machines.

Precision Software has another test: if the Machine Language Monitor returns eight two-character values across the screen when you display memory, you've got a wrong version. Try it: BANK 15:POKE 6,0:SYS 6 <return>, then: M 8000 and count how many bytes are there, eight or sixteen. As far as I can tell, the machines which return 16 values also are of type 1 as returned by the PEEK above.

As you can see, we now know of 3 machines. If your value at 65529 is not any of 0, 1, 170, or if it is 1 but programs don't run the way they should, send us the dumps as per instructions on page 12 in the winter/spring ESCAPE.

PRE-SUPERSCRIPT

As originally submitted, this program uses a bit of machine code to change things so that the keys don't bounce. It then prints a line on the screen. The line contains a superscript loading command. There are reasons why I did not chose to make it a load-and-run command and why you have to press RETURN when that line appears on the screen:

1. RUN key forces a drive 0 load. If you want to ignore the "place superscript in drive 0" instructions and load the rest of superscript from drive 1, you can do it in direct mode by editing the line that my program prints. Otherwise you're stuck with drive 0.

2. Superscript permits modifying printer modules and function keys. So if you do just that, you need to load all your private things before Superscript is loaded. My setup permits you to do this easily: load and run the fix program, then your things, then Superscript. You can, but do not have to use my pre-printed line.

3. I didn't want any commands or program lines in that area of computer's memory into which Superscript loads, for those commands could get overwritten by the Superscript program if Basic is not in a normal place or if the loader gets too long. Direct mode commands are processed in the input buffer so we're safe.

Unfortunately when this program got to Marlin's group in California they changed it. They made you press lots more keys than just one RETURN and they made it a load-and-run type of thing. So some people have an inflexible system. Other people have modified the version that CBUG distributes, with the same result - lack of flexibility. Joe Rotello brought this to my attention in his note about printer setup in the winter/spring ESCAPE on page 20 where intercepting a load-and-run system is discussed. If you leave the program intact, if you have a legitimate copy from CBUG, you should have no problems loading and running anything you wish between the keybounce fix and the big Superscript program.

This note is NOT meant to discourage anybody from changing things anyway they please, honest. But cosmetic changes sometimes can create problems which then seem rough to fix.

MAKE USE OF DUAL DRIVE

8050 is a dual drive, a marvellous invention. It can be used to backup floppies quickly and for zillion other things. I don't see how any serious program, especially business kind, can run on single drive systems. But unfortunately

980 V$=V$:GET I$:PRINT:RESUME NEXT :REM AT CLOSE 3.
    Note: Don't know if this will run in B256. Try changing the BANK 2 command to something like VB=PEEK(83):BANK(VB).
The point is to send the string data to the bank where simple variables are.

THE SID-SINGING B128

   Converting music programs from C64 to the B needs to be handled with care. Several pointers:
   1. TI on all computers except the B is reserved variable. It counts time in jiffies, intervals of 1/60th-second. So
if you see code such as 10 IF TI<TX+DR GOTO 10 you can it translate it as delay until TI reaches a value TX+DR. B machine
has no TI. We have a real, on-the-chip clock which returns TI$. TI$ is seven digits long. Last digit is 1/10-th of a
second. Conversions can be made once you know that. All C64 music needs to be slowed down on the B.
   2. Pitch table: C64 programs often contain DATA lines with note values for the SID chip. These values are wrong for
the B machine because B runs at 2 megahertz instead of 0.92 of the C64 rate. All pitch values need to be converted.
Protecto Guide has a good table of correct pitches. If you'd rather calculate your own things, use the formula which is
in the SID chip specs in the C64 Programmers' Reference Guide, but substitute the correct system clock rate, 2e6 or 2
million.
   3. It's unwise, in my opinion, to do any additional calculations over what the C64 program had. They normally slow
things down, but, and this is the worst problem of all: extra calculations can change the beat so badly that songs become
unrecognizable. The best way to handle SID chip in BASIC is by using lots of lookup tables and minimum of calculations
while the notes are being scanned. If you must run extra calculations, make sure that even if a voice doesn't need one,
things must balance out - stick in some dummy delays to keep things even for each note and each voice.

NON-SINGING B128

   SID chip doesn't shut up correctly in some programs such as Superscript. When I use the video-output option, the SID
whines at a very annoying high pitch while the program waits for a key. Simple solution: plug it up. Any mini-plug (1/8th
inch from Radio Shack) disconnects the internal speaker. To me that's simpler than constantly struggling with some
control keys to turn off the sound, and anyway SID doesn't turn off in many programs which only rely on setting the
volume register to zero.

REVIEW OF ANDERSON'S MEMORY CARTRIDGE

   B128 computers are short of memory in Bank 15. Consequently, many machine code programs or routines need to be
coded the hard way - in banks other than 15. To remedy that situation, one can expand the memory in Bank 15. You can do
it yourself if you have a Calc Result cartridge (see w.exp15 on cbug#17 floppy).
   If you'd rather not, Gary Anderson has built a cartridge, ready to go. It has more memory than can be put in the
Calc Result cartridge, but it has no RAM/ROM switch. The memory goes from hex $2000 to $7fff for 24K expansion. You can
plug in either RAM (read and write) or ROM (read only) chips, but you cannot lock what's in RAM to behave as if it were
ROM.
   The cartridge shell is much better than the Calc Result version - it doesn't wobble and is unlikely to come lose.
I'm not too knowledgable about the electronics part of it, but a friend who is, is very impressed with how the cartridge
is built. Not only is the expansion done correctly but all the nitty gritty resistors and things are designed just right,
the way it should be.
   Needless to say, the extra memory is just fine. I've ran a thorough RAM test on it several times, and it never
fails. Having extra memory in the B machine's Bank 15 is a time saver. I recommend it highly.
   There is a version of the cartridge that you can put together yourself. I haven't seen it, but the instructions look
very good. In fact the instructions about using the cartridge and/or assembling it are well done, most educational to
people like me and seem quite correct.
   The ready to plug in cartridge sells for $40, the unassembled version sells for $15 and the shipping charge is $2.
It's all available from:
Gary L. Anderson
Anderson Communications
1528 34th St. SE.
Cedar Rapids, IA 52403

REVIEW OF BEELINE

   BeeLine is a terminal program for the B128. It was written by John and Ken Lemkelde of Dover, Pa. I've used it now
for about three months on DELPHI and like it for the most part. BeeLine captures the incoming and outgoing communications
into a huge, two-bank buffer or a disk file. In both cases, unlike BTERM, the text is clean, i.e. stripped of all the
garbage DELPHI sends, such as line feed characters (j) , deletes (h) and the binary zeros (@). When read into
Superscript, BeeLine's files are clean and easily managable.
   BeeLine is able to send information typed at the keyboard in real time. It sends control characters and whatever is
necessary for the networks such as Delphi. BeeLine can also send a text file directly from disk but not in the same way
as you may have learned in BTERM - follow you Bulleting Board's instructions.
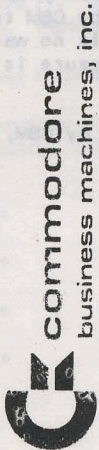   I haven't had much luck with function key definitions. I went through the motions of defining function keys as per
instructions. They looked OK, they listed OK, they saved OK. But when it came time to actually use them on DELPHI, all I
got was garbage on the screen.
   I can save all or any part of the buffer to disk or print portions of the contents on the printer. It all works like
a charm! One poorly documented section caused me lots of problems at the beginning: once a block, a section of a buffer
is defined, you cannot GO outside of that block until you return to the menu, which in turn may set the block to be the
whole text - so don't think you've lost data if you can't seem to get to it!
   One thing I don't like about BeeLine is how the buffer is displayed. It is byte-number oriented and it can only
print text in the forward direction. No back and forth scrolling as you know from BTERM. All the necessry tools for

Page 2   Technical Support Bulletin .

# OF FLASHES	RESOURCE	COMPONENT	8050 LOCATION
1	Zero page	6532	E1,C1
2	ROM $E000-$FFFF	6332	L1
3	ROM $C000-$DFFF	6332	H1
4	Unused		
5	Controller ROM, Zero page	6530,6502	K3,H3
6	Common RAM $1000-$13FF	2114	C4,C5
7	Common RAM $2000-$23FF	2114	D4,D5
8	Common RAM $3000-$33FF	2114	E4,E5
9	Common RAM $4000-$43FF	2114	F4,F5
10	Controller ROM	6530,6502	K3,H3

If the LED's come on and stay on the main microprocessor is probably at fault.

3. Track Offset Recovery Technique.

a) To enhance the error recovery capabilities of the disk drive by off setting the head from the central track position. This technique allows recovery of data that appears off track (i.e. mis-aligned drive or misclamped disk).

b) The controlling processor determines the error recovery sequence defined by the recovery status flag/counter. The power up state includes the following sequence.

	Number of steps from original track	Number of recovery attempts
1.	0	6
2.	1 out	5
3.	0	5
4.	1 in	5
5.	0	5
6.	2 out	5
7.	0	5
8.	2 in	5
9.	track 0 restore	1
10.	0	5

Notes: A. Steps 5 through 8 used only on 8050 (four per track).

B. The recovery status may be modified under program control to:
    i) Change the number of retry attempts.
    ii) Disable steps 2-8 and 9.

—

Page 3   Technical Support Bulletin .

C. For the write operation, steps 2 through 10 are executed using a sector seek job rather than attempting to write off track. After steps 2-10 are executed, the write operation is retried an additional 5 times.

c) The file interface controller is the controlling processor.

The disk controller will respond to a track offset request via a phase variable for each drive, resident in common RAM.

The controller will detect the following values of phase variable set by the DOS:-

    $00   No action.
    $01   Step out one step from current phase.
    $FF   Step in one step from current phase.

NOTE: 1. These actions are pending during movement from track to track. The phase offset is valid only when the controller has finished the seek to track.

2. The controller must replace the non-zero PHASE value after executing the offset.

TECHTOPICS

CAUTION IN USE OF DISK I.D'S

The user must be careful if he formats his blank diskettes with the same ID. It is recommended that the ID vary from diskette to diskette. This can easily be done by using the RND function in basic or picking two numbers such as the minutes past the hour of a clock.

A sector allocation map is maintained in RAM by the DOS for diskettes in drive 0 and drive 1 respectively. This map in RAM is updated only when an initialized command is executed. If two diskettes have the same ID, and the second is inserted into the same drive that the first diskette has been initialized in, then the allocation map will not be updated because the DOS does not know a new disk has been inserted.

The danger of diskettes with identical ID's occurs when the user now creates a new file, expands an existing file, or replaces an existing file. The allocation map, which is updated, is from the first diskette and not the second diskette which is being re-written.

The following example demonstrates this problem.

1. Create two diskettes with the same ID by duplicating a diskette.
2. Write a new file to the backup diskette. Use a unique program you can distinguish from others.
3. Initialize the original diskette.
4. Remove the original diskette from the drive and replace it with the backup.
5. Load a different program unique from step 2 and save it under a new name.
6. The program from step 2 is now lost.

REGIONAL NEWSLETTER

Feb. 8, 1982

Commodore Midwest

The big news this week is the Second Annual Software Show. I'm looking forward to seeing you all there. Be sure to read the enclosed announcement CAREFULLY. It's good news.

8050 Solution

Several people have called lately with an old problem, so its time to touch on it again. Under certain situations, maybe once a year, the head will wander off it's beaten path and get lost. The result is a drive that won't respond to a shift/run. The solution is simple: just initialize the drive with this command:

open1,8,15,"I" (return)

That will force the drive to do a 'bump' and work properly.

This actually occurs very rarely. It can happen when you run diagnostic programs, programs that do some diagnostic checking (like LTA) and certain other rare situations.

By the way, turning off and on the disk will not solve this problem; only the above simple command.

Tax Prep

Tax Prep is out to everyone who had a standing order. If you had one on order and haven't received it, you might consider checking on it.

That's all for now....   KRP

HARD DISK

CBM 9060/9090

Hard Disk Error Diagnostics

Number of Flashes	Possible IC's	Pcb Location
1	6532	7f or 7g
2	ROM	7d
3	ROM	7c
4	6502	4a
	2114	5e
	741s157	6a,b,c,d
	721s42	3d or 4a
5	6810	4d
6	741s157	6a,b,c,d
	2114	5a,5e
7	2114	5b,5f
8	2114	5c,5g
9	2114	5d,5h
10	ROM	4c
11	Power on Detection	
12	741s157	6a,b,c,d
13	741s157	6a,b,c,d
14	741s157	6a,b,c,d

NOTE: The 11 flash sequence means that the controller has detected to
many bad sectors for proper operation.

8050 Problems

If the On Power Sequence Unit sometimes flashes very quickly or a faster than normal flash sequence, check the clock input on 6502 (UN1). If there is a difference from the normal power up clock frequency to the clock pulse during flashing, possible defective UA6-74LS04.

Drive doesn't spin when trying to load, displays loading but never actually loads program, possible UA2-7414 defective.

Flashed four times-replaced UD3-74LS157. Try UC4 & UC5-2114 RAM first.

Drive 1 okay but Drive 0 had intermittent directory and loading. Checked 12V reg. and found ripple, replaced & checked okay.

Drive would load all programs but when running performance test, it came back with read error 23-38-03. Replaced RAM UF4 and checked okay.

If LED's flash and you have changed the possible defective component(s) listed in tech topics, try changing the RAM starting with UC4 and UC5-2114 RAM.

Five flashes and have already changed UK3 and/or UH3 and still flashes, change RAM UD4 or UD5-2114 RAM and check.

Unit would load DOS and performance but head would not step when trying to run performance, replaced UJ6-pins 8 and 9 shorted.

After heated up if unit motor still spins but no stepper motor operation, check UK6 (GCR ROM) couldn't find by cooling chip. Had to change, couldn't load anything.

After warm up unit (Drive Motor) still ran but no stepper. Turned off and back on unit flashed continuously, check 6502's. No directory or can't read directory-check phase lock loop. UN4-4066. No directory or loading a program-check phase lock loop ckt.

Seven Flashes-UF4 or UF5-2114 RAM.

On Test Disk (8050) #86 "Fintest"-return to print soft error. Test will only work with 901482-06 & 07 ROMS and will not work with 04 & 03 ROMS.

8050 Digital Card on power up only flashed once (red to green on center led, not twice) changed UH1 and checked okay.

8050 Digital Card-power up okay wouldn't load anything. Red led (error) flashed. Replaced UK6-GCR ROM and checked okay.

98

2040, 4040 Problems: 5 flashes and have already changed K3 and/or H3 and still flashes, change RAM C4 and C5. First change C4 and check. Same Symptom-UB4-74LS00

8 Flashes (4040) UE5.

Check C23, 24, etc. for 5 Volts on one side and ground on the other.

Data file error on performance test. Newed okay. Check GCR ROM – 901467-01 UK6.

Loading DOS – comes up searching, loading ready, type in run and comes up ready, check socketed 6532's PIA.

Comes up device not present, check UB2 Pin 4 or 12. Should go from low to high. Could be either UA4-7414 or UL2-74LS04.

Try to get a directory – no motor start up or head movement – check 3446 (UD2, UB2, UB1). Also UJ2-74LS00.

All lights on and stay on after power up, if problem is not 6502 check UA3-74LS42. Also check DOS ROMS for defect or in wrong position. Same Symptom-2 bad rams E4 & F4

Loaded performance test ok-typed in run and stepper went dead-Replaced UK3-40 pin DOS rom(901466-04)

Loaded ok but would not"format"when running soft error test-Replaced UM6-74LS165 (write cktry)

When checking 12 & 5 volt supplies you find AC ripple-Check 23000 mfd power supply filter capacitors(2), usally screws are loose. No reset pulse and have already replaced 555 and 2 capacitors in reset circuit. Check for bad buffer UA4-7414 pins 3 should be at a high level state and 1.6K ohm approximate value across R4. Also check UN2-7406 pin 6 to ground (pin 7) for 10K ohms approx.

As a 2040 board, the board checked okay; when configured to a 4040 board, it flashed pointing to RAM location D4 or D5. Replaced D4 and checked okay.

4040 flashed 8 times. Dealer had replaced specified RAM locations E4 and E5, replaced F5 and checked okay.

Trying to load DOS. file not found error-replaced 6522 and checked okay (motor RAN and head stepped).

Directory would come up incorrect or not at all-wouldn't load at all:check UA4

2040/4040 ANALOG PCB: Wont read or load-stepper and motor work-check 592's & LM311. Same Symptom-found open coil L4 Same Symptom-found UA3-7486

Hardware/Mechanical
Alignment - read errors and load problems.
Belt Problems - read errors on sync. track. (21 . . . .)
Motor Problems- read errors on sync. track. (21 . . . .)
Speed Off -   read errors on sync. track. (21 . . . .)

2031 SINGLE DRIVE

ON POWER UP

4 Flashes-Replaced U3C,2114 RAM.

Drive light stays on & motor runs-Replaced U3J,6522.
Same Symptom: Found solder short on 555 timer pin 5 to C57.
Same Symptom: Found U4D,74LS193 in xtal ckt defective.

Drive spins and LED flashes-Used NO-OP 6502 & found address 9 (A9)
signal missing on last two RAM's-found open foil run under second RAM.

POWERS UP OKAY

No directory-press shift/run stop,error file not found: Found one lead
of C38 not soldered in.(pin 2 of U2P & gnd)

Shift/run stop,error device not present: No signals present on pins
10-17 on 6522 - U3H. Replaced 75161 interface mux/decoder chip.

Loads okay-tried to run performance test came back with read error 21.
?ds$ command-write protect on: Replaced 74LS04-U6F.

Flashes then file not found error-?ds$-74 drive not ready: Both
592's-U5R & U3R bad.(look for signals at C47 & C46)
Same Symptom: Found shorted capacitor C68.(47 pfd.)

1540 and 1541 DISK DRIVES

Visually check drive for all possible defects-broken head wires-jammed heads-belt in place-dirty or improperly seated load pad-does clamp assy keep diskette from slipping-clean head slide rails...LOOK FOR THE OBVIOUS

On power up drive flashes 4 times-found UA2-2114 RAM bad

READ/write problems-try demagnetizing head-also check head alignment

The biggest problem with the Hard Disks seems to be in getting it headered. Here are a few suggestions:
     Upon receiving the drive, remove the top cover and be sure ALL SOCKETED IC's and ALL CONNECTORS are seated properly ! Visually look over the drive to see that there is nothing loose (xmfr,pcb's,mounting screws). Power up drive and let run for five (5) minuits. There are two different header commands that can be tried: 1st-header"fn",d0,icm (RETURN) and y for yes, or try-header"fn,icm",d0 (RETURN) and y for yes. When formatting is completed the cursor will be present on the screen. cAd0 and nothing happens? Try these commands - open1,8,15,"uj (RETURN) and try directory again, still nothing ? Try the same command with "i0 to initilize the drive. Sometimes powering the drive off and then back on will also work. I strongly suggest you have an extra set of pcb's for troubleshooting the hard drive. Once you've narrowed it down to a specific pcb its easier to t-shoot one board than the complete drive! This will also allow you to find out if the drive itself is bad,which can happen usally due to rough handling! ANY HARD DISK DRIVE SENT BACK IN ANYTHING OTHER THAN THE ORIGNIAL PACKING CARTON OR CARTONS WILL RESULT IN YOU RECEIVING A BILL FOR REPAIR REGUARDLESS OF WARRANTY COVERAGE!!

Calling up a directory you get 74 read error:drive not ready-try open command. If still in error try changing second pcb from the drive(TM600-cont.&data). I found an open diode-CR8.

Communication problems-check top board! Be sure all IC's are seated & in correctly.

102

NUMBER OF FLASHES OF LEDS IN REPETITIVE SEQUENCE IF FAILURE

Number Flashes	8050 Error Cause	Component Location	2040 Error Cause	Component Location	4040 Component Location
1	Zero Page	6532, C1, E1	Zero Page		6532, C1, E1
2	ROM 901482-03 MIC (901482-07 TAN & MIC)	2364, L1	ROM (901468-07)		H1 (901468-13 Old) (901468-16 New)
3	ROM (901482-04 MIC) (901482-06 TAN & MIC)	2364, H1	ROM (901468-06)		L1 (901468-12 Old) (901468-15 New)
4	RAM (C4 or D4)	C4 or C5-2114 (901483-03 MIC) (901483-04 TAN)	ROM		J1 (4040 Only) (901468-11 Old) (901468-14 New)
5	RAM or Zero Page	6530-K3 or D4 or D5 6502-H3	Zero Page 901466-02-028 6530-028 (2040)		6530, K3 901466-04-034 (4040) 6504, H3
6	RAM	E4 or E5	Illegal		
7	RAM	F4 or F5	RAM		2114, D4, D5
8	Illegal (Could be RAM)		RAM		2114, E4, E5
9	Illegal (Could be RAM)		RAM		2114, F4, F5
10	ROM	6530, K3 6502, H3	ROM		6530, K3 6504, H3

DOS Problems and Fixes

```
1000 rem fix four bugs
1010 rem need 8050 6,7 (4040 14,15,16).
1020 rem buffer write (fixed).
1030 rem timing problem of controller(rel files)(no update of record).
1040 rem chance of controller working on a buffer and dos will release
1050 rem buffer to another job.
1060 rem copy d0 to d1 had bugs fixed (9 th file id mismatch)
2000 rem tandon 8050 problems.
2010 rem track 0 allen screw is above stop gets off(may put in track zero sw).
3000 rem super dos (901887-01,901888-01) 8050,8250
3010 rem change controller (????????)
```