Aton
INTERNATIONAL, INC.

1765 Scott Boulevard, Suite 119
Santa Clara, CA 95050 U.S.A.
Telephone (408) 554-9922

Release Notes

96K Bank Switching System (Level II) - Release 2.25

February 15, 1984

To activate the bank switched system, enter the command LEVEL2 after the A> prompt. The bank switching system will then be active. The bank switching system can be automatically activated by using the AUTOEXEC utility.

Auto Keys, Print Screen and Program Abort

Five user programmable function keys (auto keys) are available in Level II. They are accessed by using CTRL-1 through CTRL-5. These codes do not conflict with the normal control keys which generate the ASCII hex codes 00 through 1F. These auto keys cannot be reassigned via the keyboard mapping table.

After a cold boot, the function keys need to be programmed. Until they are, pressing an auto key will produce nothing. They can be programmed by either the manual technique or by invoking the GETAUTOK utility program which sets the keys up in the same way they were when the PUTAUTOK utility was used.

To program an auto key:

1. Enter CTRL-7. This may be done at any time, even in the middle of as program run. If the disk is active, a slight pause may occur. If a system status message such as "MOUNT DISK IN DRIVE A:" is displayed on the upper right side of the screen, then the CTRL-7 will not be active. At any other time, pressing CTRL-7 will produce a one line menu on the bottom line of the screen in reverse video. The next key stroke determines the action to be taken.

    a. ENTER means ignore the CTRL-7 and continue on.

    b. BREAK or CTRL-C means to unconditionally abort the task in progress, be it system or user. If your program gets into an infinite loop or hangs up on the serial printer, you can use this technique to get back to the operating system without using the RESET switch. Note that any modified track buffers will not be written to disk if this exit is used.

    c. CTRL-8 means to print a copy of the screen on the LST: device. The high order bit (reverse video) is stripped, control codes (graphics) are replaced with blanks, and trailing blanks are suppressed. The CTRL-7 key can be used to abort the print operation in progress if desired. Again, this "snapshot" can be taken at any time.

    d. CTRL-1 through CTRL-5 means to program the auto keys.

    e. Other keys are beeped and ignored.

2. When CTRL-1 through CTRL-5 are entered from the one line menu generated by CTRL-7, a new menu is shown.

This new menu shows the applicable control keys and the correct contents of the auto key. Each auto key can store up to 50 characters. After this new menu is shown, the following keys are active:

a. CTRL-8 means to backspace and erase the last character entered for the auto key.

b. CTRL-7 means to store the characters entered for this auto key.

c. CTRL-1 through CTRL-5 means to select another auto key for display. This will also ignore any changes just made to the auto key definition.

d. Any other key including BACKSPACE, BREAK, ENTER, etc. will go into the auto key definition. No more than 50 characters can be entered and control codes will display as their reverse video alpha equivalent. Thus CTRL-C will appear as a reverse video C.

3. To program an auto key:

   a. Press CTRL-7

   b. Press CTRL-1, or CTRL-2, or CTRL-3, or CTRL-4 or CTRL-5.

   c. Press the keys to be included in the auto key definition. Use CTRL-8 to backspace if an error is made.

   d. Press CTRL-7 to record the auto key definition in memory and exit back to the user or system program. If you press CTRL-1 through CTRL-5 at this point (instead of CTRL-7) you will <u>not</u> record the changes you just made.

   e. The auto key can be used at any time in a user or system program by simply pressing CTRL-1 through CTRL-5. If there is enough room in the key ahead buffer, all the key codes programmed into the auto key will go into the key ahead buffer (after any already there) and will be displayed on the status line as usual. These will subsequently be used by the user or system program. If the buffer is too full, then the auto key will be ignored.

4. The auto key definitions can be recorded to a disk file with the command PUTAUTOK. A file name may be given if desired, otherwise the file name AUTOKEY.BUF will be used.

5. The auto key definition thus saved can be recalled any tme later with the GETAUTOK command. A file name can be given (the one used with PUTAUTOK), otherwise the default file name AUTOKEY.BUF is used.

6. The assembly language source code for GETAUTOK and PUTAUTOK is included so that you may learn how to modify the auto key buffer from a user program if desired.

User Program Access to Alternate Memory Banks

User program may access alternate memory banks in either the Z-80 or 68000 memory. Study the assembly language program SETLPP.MAC for detail on how to correctly switch banks in a way that will allow interrupts to be enabled while the other bank is accessed. You should use the SYSDEF system options to declare Z-80 or 68000 memory banks for private use by your program so that the operating system won't use these banks for track buffers. Also, the stack and any disk or status buffers referenced by EIOS calls must be in the lower 32K.

BDOS Calls to the TPA BIOS Vector

The BDOS resides in another bank so BDOS usually calls the BIOS through a vector table that is not accessable to user programs. For some special purposes such as memory disk cards and non-Radio Shack hard disks, it may be necessary for the BDOS to call the BIOS vector in the TPA bank. This can be selected as an option by SYSDEF (system option) but will degrade throughput somewhat.

SYSDEF - Define System Parameters

This program only operates under the 64K unbanked (Level I) system. Any changes made to the 64K system are copied to the 96K system when it is bootstrapped.

BDOS Functions 27 and 31

These BDOS functions return the address of BIOS allocation and disk parameter tables for the currently selected drive. These are copies of the actual tables located in the system bank. So attempts by user programs to modify these tables are meaningless. Also, there is only room for one allocation table and disk parameter table in the TPA bank. Programs such as DU.COM that attempt to access tables for a previously selected drive will fail.

Available Space at Top of Memory

There is usually 256 bytes at the end of memory (0FF00H through 0FFFFH) for program parameter passing purposes. This area is a convenient place for user programs to pass parameters between job steps or links in a program chain. This last 256 bytes will be disturbed only if BDOS function 27 (Get allocation table address) or BDOS function 31 (Get disk paramter table address) or BIOS SELDSK function is called from the user program and the selected drive is a hard drive logical unit using 2K allocation blocks and the size of the drive is larger than 4 megabytes. The STAT program is one program that uses BDOS function 27 to determine the amount of available space.

Keyboard Buffer Display

The key ahead buffer (up to 64 characters) is displayed on the status line. Any control characters are displayed in reverse video alpha. For example, a CTRL-C code (03) is displayed as a reverse video C.

Thin Line Drives

Level II will purge its buffers whenever it detects the door is open on the drive for those buffers. There are two signals available that relate to the open door. One is called NR and is available on all drives. On full size drives used in the Model II, it indicates an open door. On the thin line drives used on Models 12 and 16, the NR line means that either the door is open or the motor has shut off. Thus, if the NR signal is used on a thin line drive, Level II will flush its buffers whenever the drive motor goes off, even if the door is not opened. This degrades throughput.

Another signal called DC is available on most thin line drives and some of the full sized drives. The CDC drives used in the early expansion bays did not have the DC signal. The DC signal goes on only if the door is open, whether or not the motor is running. So for thin line drives, it is the signal to use if it is available. Unfortunately, some Model 12s and 16s have been shipped where this signal is not active.

Level II must be informed when a disk is changed, or otherwise incorrect data can be written onto a diskette. Level II is shipped set up to use the NR signal for open door detection. This works fine on Model IIs but will cause loss of performance on Model 12/16s. SYSDEF can be used to specify the thin line DC option for thin line drives. After specifying thin line to SYSDEF, check out the operation of Level II as follows:

1. Do a DIR on the drive to be tested. The drive may or may not be accessed as indicated by the disk activity indicator on the right end of the status line.

2. Again enter the DIR command for the drive to be tested. The drive should not be accessed. If it is, the DC signal is locked on and needs repair.

3. Wait for the motor to shut off (20 seconds). Again enter the DIR command. The drive should not be accessed. If it is, either you have not specified the thin line option for the drive or the DC signal goes true when the motor shuts off - which shouldn't happen.

4. Now for the important test. Open, then close the drive latch after you shove the diskette back in. Enter the DIR command and note if the drive is accessed. If the drive is not accessed, then the DC signal is not active. Either get the drive fixed or go back to using the NR signal by turning off the thin line option for this drive. Do not use the system if it won't detect a disk change because you will probably end up writing the wrong directory on a diskette, making it useless.

Immediate Buffer Purge

Level II will normally purge modified buffers within ½ second of console activity occurring. This is done in anticipation of the operator possibly changing diskettes. On the hard drive, this is not needed and you can improve disk throughput by disabling the immediate modified buffer write. It can be disabled by using option E on the thin line disk systems parameter menu of SYSDEF. Modified buffers for hard disk will then be purged every minute, or when the user program issues an EIOS redetermine density call, or whenever the BREAK key is used to cause a warm boot. If you disable the immediate write, you must remember to use the BREAK key before hitting the RESET switch or powering down the computer. Otherwise, you may lose the last data to be written.

TRSDOS 4.2 on Model II

You can run TRSDOS 4.2 on a Model II if you have the extra 64K memory board (26-4105) installed and jumpered so that it is at banks 14 and 15. Set the bank jumper from pin 19 to pin 27. This is used instead of a jumper from pin 15 to 28 which selects banks 2 and 3. With the alternate jumper to banks 14 and 15, both TRSDOS 4.2 and ATON CP/M Level II can be used. The ARCNET and hard disk interface boards also provide the capability to operate TRSDOS 4.2. If either of these is used, then use the bank 2 and 3 jumper on the 64K memory board.

Model 12/16B Boot ROM

Some early releases of the Model 12 or 16B have a boot ROM with the characteristic that if the system includes a hard drive and you boot CP/M from floppy by using the BREAK or ESC key with REPEAT, Level II won't boot properly, even though Level I will boot up. If you press the BREAK key or ESC key exactly once after hitting the RESET switch (you will have to experiment to get the right time delay after hitting the RESET switch in order for the floppy to be used), Level II boots up properly.

This problem hasn't been seen on Model IIs or later Model 12/16Bs.

Improved Performance on Model 16

Performance can be improved on a system that has less than 96K of Z-80 memory by adding a 64K Z-80 memory board (26-4105). Without the board, the operating system will reside in 68000 RAM which, because of synchronizing wait states, will allow the operating system Z-80 instructions to be executed only half as fast as if the operating system was located in Z-80 RAM. Because the user program and critical portions of the operating system are always located in Z-80 RAM, the performance improvement will be about 10% to 30% with the addition of a Z-80 memory board.

Communication and Supervisory Programs

Certain types of programs will not function on the Level II system because they make assumptions about the structure of the CP/M system which are not valid for the Level II system. These programs are usually in two categories. Console command replacement programs such as Micro Shell, Supervyz and Power won't work and some data communication programs such as Lync won't function completely. These programs do function under the 64K Level I system which is included with the 96K Level II system. All of the commonly used languages, spread sheets, data base programs and word processor/text editor programs work correctly with the 96K Level II system. Also, some communication programs, such as MOVE-IT from Woolf Software, work correctly with the Level II system.

Model 16A and Model II with 16 Bit Upgrade

In the Models 16A and II there exists a design problem that prevents Level II from operating correctly in about one-third of these machines. This problem has not been reported on the Model 12 or 16B. (See the note above concerning the 12/16B boot ROM.) There is no way to predict, based on serial number, which of the Model 16As or Model IIs will not operate correctly.

The problem relates to the speed with which data is transferred from 68000 RAM to the Z-80 CPU. ATON CP/M Level II, if there is less than 96K of Z-80 RAM, will execute code located in 68000 RAM. ATON CP/M Level II with more than 96K of Z-80 RAM, as well as TRSDOS and Xenix, do not execute code from 68000 RAM but rather just store data in the 68000 RAM. Executing code from 68000 RAM requires that the data be available from RAM faster than if the RAM is just used to store data. The Models 12 and 16B use a faster series of bus buffer chips than the Models II or 16A. These faster chips allow Level II to operate correctly on the Models 12 and 16B. Individual part variations on Models II and 16A cause some of these machines to fail with Level II.

The symptoms of this problem are either that Level II will cease to operate after 10 to 30 seconds of operation or that bits will be dropped in directory entries or data files.

If you are experiencing this problem, there are a number of approaches to fix the problem.

a. Take the machine to Radio Shack and get them to bring the system up to the latest revision level. A Radio Shack service technician tells us that the "F" level revision to the Z-80 CPU board rectifies the problem, although we haven't verified this.

b. Swap Z-80 CPU boards with another Model II or 16A. Sometimes this is the easiest fix.

c. Check the Z-80 CPU chip and the DMA chip for corrosion on the pins and in their sockets.

d. Replace the buffer chips on the Z-80 CPU board with faster ones. The following chips on the Z-80 CPU board affect the memory read out speed. Replace a few at a time until the problems disappear.

1.	U17	From 7404 or 74LS04	to	74S04 or 74F04
2.	U34, U35	From 74LS240	to	74ALS04 or 74AS04
3.	U33	From 74LS244	to	74ALS244 or 74AS244
4.	U21	From 74LS32	to	74ALS32 or 74AS32
5.	U22	From 74LS10	to	74ALS10 or 74AS10

The Fairchild "F" series family can be used instead of the TI "AS" or "ALS" families.

e. Replace the buffer chips on the 68000 CPU board.

U8, U9	From 74LS245	to	74ALS245 or 74AS245

There is no guarantee that the above techniques will solve the problem, although combinations of these have solved the problem in many installations.

"Lost" Keystrokes

It sometimes may appear that keystrokes are lost from the key ahead buffer. Under certain circumstances, keystrokes can be rearranged, but not lost.

There are four different ways that keystrokes can be read by a user program:

a. Via BDOS function 10, Read Console Buffer

b. BDOS function 1, Console Input

c. BDOS function 6, Direct Console I/O

d. BIOS function CONIN

There are four corresponding ways that characters may be listed to the screen:

a. BDOS function 9, Print String

b. BDOS function 2, Console Output

c. BDOS function 6, Direct Console I/O

d. BIOS function CONOUT

If either technique "a" or "b" for listing characters to the screen is used, then BDOS will input a character from the keyboard if one is ready and check it to see if it is a CTRL-S. It if is, it will halt output to the screen until any other character arrives. If the character is not a CTRL-S, it is saved until a BDOS function 10 or 1 is used to read keyboard input, and subsequent CTRL-S characters just stack up in the keyboard buffer.

Techniques "c" and "d" for input and output bypass this console list halt mechanism, so mixing techniques "a" or "b" with "c" or "d" in the same program causes problems.

Many application programs (e.g. BASIC and others) use technique "c" for console input and technique "a" or "b" for console output. Thus one character gets sidetracked into BDOS storage until the program exits the job and the A> prompt occurs, then the sidetracked character shows up on the command line.

There is no fix for this problem at the operating system level; the application program must be changed to be consistent on input and output to the console.