



THE SOFTWARE LINK

Creating Unbeatable Solutions

" Does Your PC-MOS Machine..."

Troubleshooting Guide

We at TSL know that your time is very valuable. We have provided this troubleshooting guide as a tool to assist you in quickly resolving issues related to our PC-MOS product.

NOTE: This publication is distributed as a guide only. In no event will TSL be liable to you for any damages, including any lost profits, lost savings or other incidental or consequential damages, arising out of the use or inability to use any information contained in this publication.

**THE SOFTWARE LINK, INC.
3577 PARKWAY LANE
NORCROSS, GA 30092**

**(800)766-LINK * (404)448-5485
FAX: (404)263-6474
TWX 4996147 SWLINK**

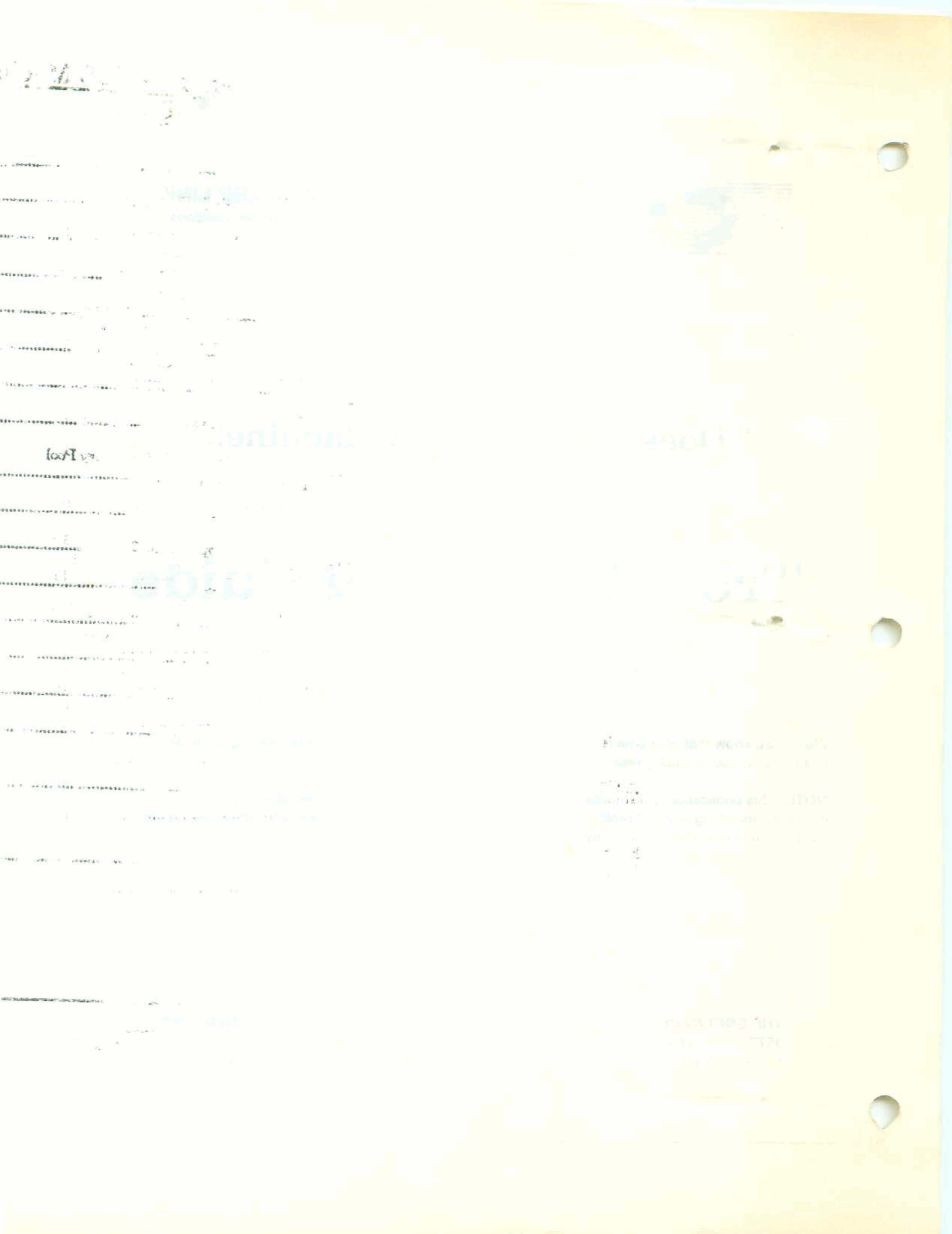


Table of Contents

"Does Your PC-MOS Machine..."

Halt during boot?	2
If it halts only when MEMDEV driver is loaded, then:	2
Experience an "error reading \$\$\$HELL.SYS "or "FILE ERROR 03"	3
Halt when a device driver is loading?	4
Halt on boot even with FREEMEM=N?	5
Experience Intermittent system failures?	6
Not have enough memory to run applications?	7
Not allow you to switch into graphics mode to run applications?	8
Experience "insufficient SMP" errors OR "Internal Error # 4 - System Memory Pool depleted"?	9
Experience an "invalid \$\$USER.SYS file!" error?	10
Produce erroneous calculated results on system with math coprocessor?	11
Not activate COM port(s)?	12
Not activate LIM (EMS)?	13
Not execute multiuser applications properly?	14
Run application(s) very slowly?.....	15
Depletes SMP when an application is executed?	16
Experience an "ERROR INITIALIZING WORKSTATION" with VNA?.....	17
Experience an "ERROR INITIALIZING WORKSTATION" on a MaXtation?	18

Table of Contents

"Does Your PC-MOS Machine..."

Halt when the VGNA driver is loaded or produce an "ERROR INITIALIZING WORKSTATION"?	19
Not print?	20
Print slowly?	21
Not allow you to print at workstation(s)?	22
Mouse not work?	23
Experience a "CANNOT START TASK - MOSITASK" error?	24
Experience an "Internal Error #0 Unexpected error during function 00"?	25
Experience an "Internal Error #0 Unexpected error during function 31"?	26
Experience an "Internal Error #0 Unexpected error during function 4b"?	27
Experience an "Internal Error #EE - re-grab overflow"?	28
Experience an "Internal Error #E1 - unable to remove task"?	29
Experience an "Internal Error #E3 - Unable to restart task"?	30
Experience an "Internal Error #E4 - Unable to return heap space"?	31
Experience an "Internal Error #E0 - terminal not in list" ?	32
Experience an "Internal Error # - Cannot start task - MOSITASK" error?	33
Experience an "Internal Error # - Command processor cannot be found"?	34
Experience an "Internal Error # - Error: closing \$\$\$HELL.SYS"?	35
Experience an "Internal Error # - SMP corruption during device driver loading"?	36

Table of Contents

"Does Your PC-MOS Machine..."

Experience an "Internal Error # - Error during INIT or open of standard drivers - error type X"?	37
Experience an "Internal Error # - 8 MEMDEV interrupt # XXXX conflict"?	38
Experience an "Internal Error #01 - Extended Service xx no longer supported"?	39
Experience an "Internal Error #0C - error resizing task"?	40
Experience an "Internal Error #59 - Invalid IRQ list"?	41
Experience an "Internal Error #58 - Invalid IRQ"?	42
Experience an "Internal Error #80 - error freeing heap in int 24"?	43
Experience a "DIVIDE BY ZERO error"?	44

"Does your PC-MOS machine..."

PC-MOS, being a multitasking / multiuser operating system, can be perceived as being complicated. It really is not. This guide is not designed to replace the PC-MOS user's guide, it is merely a companion product for quick solutions.

Most people, when having a problem with their PC-MOS machine, will say "MY PC-MOS MACHINE ...". That is why this book is called DOES YOUR PC-MOS MACHINE...? . It is organized by common errors we make in setting up a PC-MOS system and their solutions. We hope you find it useful.

Halt during boot?

Then boot with the original system diskette and try the following key items to determine what might be the problem.

If it halts only when MEMDEV driver is loaded, then:

- Make sure your system has at least 1 Meg of RAM with the RAM above 640 K configured as EXTENDED memory.
- Issue a FREEMEM=N in the CONFIG.SYS file and remove all other FREEMEM commands. If this prevents a lock up, then you have a memory conflict. Determine where all devices are addressed (within the 640K to 1 Meg range) and adjust your FREEMEM statements. Check the system's CMOS setup to see if SHADOW ROM is enabled. If it is enabled check the system documentation to determine the address location and adjust your FREEMEM statement accordingly.

If the problem still persist try the /C operand on the MEMDEV= command. This will prevent kernel 2 from moving into extended memory, where a conflict might be occurring.

Keep in mind that systems with EGA or VGA use the area between C0000 (C000:0) and C8000 (C800:0) for video BIOS.

NOTES:

Experience an "error reading \$\$SHELL.SYS" or "FILE ERROR 03"?

Try the following steps to solve this problem:

- Check to see if the \$\$SHELL.SYS file is located on the root directory of the boot drive. If it is not, then copy it from the system diskette or the PC-MOS sub-directory.
- If the "FILE ERROR 03" occurred while loading a device driver, then check the spelling of the file name and the location specified, to make sure the file is present.
- Check to see if Disk Manager was used to partition volumes greater than 32 Megabytes. If so, make sure the CACHE= command included the /BPS option (see the CONFIGURATION section in the PC-MOS USER GUIDE for the appropriate values).
- Check your hard disk or controller documentation to determine if they make use of DMA. If so, make sure one of the following parameters have been added to the MEMDEV= command:

/E - to set 64K buffer
/M - to set 32K buffer
/P - to set 16K buffer
/X - to set 8K buffer
/D=nn - to determine size of buffer (nn is size of buffer)

- Also, make sure the devices are not set to DMA channel 3.

NOTES:

Halt when a device driver is loading?

Try the following steps to solve this problem:

- Use a **FREEMEM=N** command in the **CONFIG.SYS** and remove all other **FREEMEM** commands. If this cures the problem, then draw a memory map showing where all devices (software and hardware) are being loaded within the space between 640K and 1 Meg. Place the addresses that are unused in a **FREEMEM** statement and remove the **FREEMEM=N**.

If the problem still isn't cured, rename the device driver and copy the original file. There might be some file corruption.

- Try the **/C** operand on the **MEMDEV=** command line if the problem persists.

NOTES:

Halt on boot even with FREEMEM=N?

Try the following steps to solve this problem:

- Make sure, if Disk Manager was used, that you have a /BPS option, with the appropriate values, specified on the CACHE= command line.
- Try placing a /C operand on the MEMDEV= command line, to prevent kernel 2 from loading into extended memory.
- Make sure, if a device is using DMA, that the MEMDEV= command contains one of the following switches:

/E
/M
/P
/X
/D=nn

- Also make sure that the device is not set to use DMA channel 3.

NOTES:

Experience Intermittent system Failures?

There are no quick and easy ways to determine what might be causing an intermittent lockup. Perform the following key steps to narrow the field:

- Make sure you are not using VTYPE with the F option if you have a system with an EGA or VGA graphics adapter.
- Draw a system memory map to determine where items are loaded. Include in this map the MOS INFO items showing where the operating system resides and any software drivers that require RAM locations (e.g. \$EMS.SYS, \$RAMDISK.SYS, ETC...). Make sure there are no overlaps.
- List all devices installed on the system with their I/O address(es) and interrupts, if any.
- If you don't see any obvious conflict from the system RAM map and I/O map, then try to narrow the lockup to a specific application. Some applications may try to make use of the RAM addresses above the 640K and may be corrupting the operating system.

NOTES:

Not have enough memory to run applications?

Try the following steps to solve this problem:

- If the task size is less than MAX (size of task 0), try a MOS RESIZE nnn command, where nnn is the desired task size.
- If the task size is already at its MAX (size of task 0), then we have to change the system configuration to maximize the available task RAM. This involves mapping the area of RAM between 640K and 1 Meg, listing all devices that are loaded into this area. If the devices can be configured to use alternate addresses within this region, you want to make sure that all devices are configured to use as contiguous block of addresses as possible. This will give you a defragmented FREE MEMORY area within this region and allow more of the PC-MOS operating system to move in, creating a larger Task 0, which will in turn allow larger background tasks.
- If your system does not have EGA or VGA on it (Host or workstations), then you may issue a VTYPE with the F option to increase the size of the host and subsequent tasks.
- You may also want to try splitting the SMP, because PC-MOS could be having difficulty finding a place, within the 640K to 1 Meg, large enough to store it. By splitting the SMP, PC-MOS can find pockets of space to fit it instead of one large chunk of memory.

NOTE: One of the SMP has to be sized to accommodate the largest device driver.

NOTES:

Not allow you to switch into graphics mode to run applications?

Consider the following:

Certain workstation environments do not support graphics (ie. PCTERM, ANTERM, TTTERM, ETC...). By default, workstations that do support graphics do not initialize in graphics capable mode. They must transition into graphics mode with the MOS VMODE command.

- If you get an error message saying "invalid terminal mode" when you use the MOS VMODE command, then you need to evaluate the type of environment you are running in and see if it supports the mode you are trying to set.
- The mono-graphics workstations (MaXtation, VNA, etc...) will only support graphics mode if the VTYPE=5 command is used in the CONFIG.SYS file.
- Make sure you have enough available extended memory to make the transition to graphics mode. If you don't, you may want to reduce the size of some of the tasks.

NOTES:

Experience "insufficient SMP" errors OR "Internal Error # 4 - System Memory Pool depleted"?

This error is very self explanatory, the System Memory Pool is too small, use the following steps to determine the desired SMPSIZE.

- **Issue a directory command on all device drivers that are being loaded in the CON-FIG.SYS file and SUM their sizes.**
- **Determine the amount of tasks that you will be invoking on the system and multiply that number by 4K.**
- **Determine the amount of RAM that will be used as buffers for the unintelligent serial ports. (These are ports that are defined using \$SERIAL.SYS, VNA.SYS or VGNA.SYS)**
- **Sum all of the above to get a subtotal.**
- **Take 30 % of the subtotal (RAM used for file handles) and add it to the subtotal for final total.**

NOTES:

...

Experience an "invalid \$\$USER.SYS file!" error?

This error only occurs when the \$\$USER.SYS file does not exist or is not found where it was supposed to be. To correct:

- Create a \$\$USER.SYS file on the ROOT directory of the boot drive.
- If you would like to store the \$\$USER.SYS file in a subdirectory other than the root, you must use the USERFILE= in the CONFIG.SYS file to tell PC-MOS where to find it.

NOTES:

...

Produce erroneous calculated results on system with math coprocessor?

This problem can be fixed with one command added to the CONFIG.SYS file, if you are using an Intel math coprocessor (8087, 80287 or 80387). To correct:

- Issue the command **8087=YES** in the CONFIG.SYS file.

NOTES:

Not activate COM port(s)?

PC-MOS does not automatically assign a logical name (COM1, COM2, etc...) to the standard or unintelligent COM ports. Because PC-MOS is multitasking / multiuser, any information that comes into a COM port must be buffered. The intelligent serial devices automatically setup a buffered interface, but the unintelligent ports don't. That is why PC-MOS will not assign a logical name to the port(s) until it knows how many buffers you want assigned to it. This allocation of buffers is done through the \$SERIAL.SYS, VNA.SYS and VGNA.SYS drivers. Please make sure you have done the following:

- Define the port(s) in \$SERIAL.SYS. If it is an IONA or VGNA serial port use the VNA.SYS or VGNA.sys drivers respectively.
- If you are using the port(s) for communication with a DOS communication program, do not define the port in the \$SERIAL.SYS driver. DOS communication packages communicate directly to the port(s) and won't function properly if the port(s) are buffered... Therefore, port(s) that are being used for this purpose will not be assigned a logical name by PC-MOS.

NOTES:

...

Not activate LIM (EMS)?

You need to check the following:

- Make sure the \$EMS.SYS device driver loaded properly.
- Verify that the 64K area specified in the CONFIG.SYS file for EMS bank swapping, does not conflict with any other device.
- If you are using PC-MOS 4.1 PL0 or less, then the \$EMS.SYS driver does not support LIM 4.0.
- If you are running PC-MOS 4.1 PL2 or greater make sure you specify the amount of EMS that is to be allocated to each task, by running the MOSADM EMSLIMIT nnnn in the tasks.

NOTES:

Not execute multiuser applications properly?

You need to know the following about the application:

- **Was it written to use a specific resource management protocol (ie. NETBIOS, Novell's IPX, etc...)? If so it can only run in the environment it was written for.**
- **Does it use the basic NETBIOS resource management and standard DOS SHARE calls for file and/or record locking? If so, then load the PC-MOS \$NETBIOS.SYS driver in the CONFIG.SYS file and activate NETBIOS in each task with the NETNAME command.**

NOTES:

Run application(s) very slowly?

You may try the following options to see if it cures the problem:

- Make sure MOS TSR is set to OFF in all tasks.
- Issue a MOS NODIS in the task(s) that is running the application(s).
- make sure you have specified enough RAM for disk caching in the CACHE= command in the CONFIG.SYS file.
- Make sure the application is not installed for a specific hardware LAN environment. When installing multiuser applications, if possible make a choice of OTHER when asked which LAN environment you are operating in. If a choice of OTHER is not given, choose IBM PC NET or IBM PC LAN, then load PC-MOS' \$NETBIOS.SYS device driver in the CONFIG.SYS file and run NETNAME in each task.
- If the problem still exists, then record all I/O addresses and interrupts that are in use on the system and make sure there are no conflicts.

NOTES:

Depletes SMP when an application is executed?

This one can be corrected by emulating the FILES=20 command of DOS by issuing the following command.

- In the task that is running the application issue a MOS FILES ON.

NOTES:

Experience an "ERROR INITIALIZING WORKSTATION" with VNA?

Since this is a hardware device, there are a lot of things that you need to look at to ensure correct configuration.

- Make sure the keyboards are set to XT or PC.
- Make sure the transformer is plugged and supplying power to the interface and translation units.
- Make sure all cables are connected properly.
- Make sure the VNA device driver correctly reflects the hardware settings (I/O address and interrupt).

If all of the above is correct, then do the following:

- Check the interrupt and I/O address assigned to the VNA card and make sure it doesn't conflict with any other device on the system.

NOTES:

..

Experience an "ERROR INITIALIZING WORKSTATION" on a MaXtation?

When this occurs check the following:

- Check the MaXtation port number on the ADDTASK line and make sure it is a valid port number.

NOTE: In a non-coresident MaXtation environment the top port of the first MaXtation board is PORT #0. In a coresident environment this is port #1.

MaXtation keyboards must be set to the AT mode.

If the error still occurs after checking the above notes, then this indicates that you have older equipment. Please contact your TSL sales account manager.

NOTES:

.

.

.

Halt when the VGNA driver is loaded or produce an "ERROR INITIALIZING WORKSTATION"?

Check the following points:

- If this is a system with a co-existing VGA card, make sure the host VGA card is a PARADISE VGA up to 800 x 600 resolution.
- If the motherboard has built in VGA that is non PARADISE, make sure you disable the video on the motherboard including all of its BIOS shadow areas.
- Make sure the keyboards are set to AT.
- Make sure transformers are plugged and supplying power to the interface units.
- Make sure all cables are connected properly.
- Make sure the VGNA device driver correctly reflects the hardware settings (I/O address and interrupt).

If all of the above is correct, then do the following:

- Check the interrupt and I/O address assigned to the VGNA card and make sure it doesn't conflict with any other device on the system.

NOTES:

Not print?

To determine the cause, do the following:

- Check cabling to make sure everything is connected correctly.
- Make sure the printer is connected to the correct port.
- If spooling is activated then do a MOS MAP and make sure there is at least one task running the PRINT processor PRINT.COM. If there is not, then add a task in which you will execute the PRINT processor.
- Make sure PRINT.COM was not executed in a task that has SPOOL.COM activated.

If you are using a serial printer, in addition to the above:

- make sure the port being used for printing is defined with a PC-MOS serial driver (\$SERIAL.SYS, VGNA.SYS, MAX.SYS, SH.SYS, ETC...) setting the correct handshaking protocol (DTR, XON/XOFF, XPC, ETC...).
- Once the port is defined, obtain the port number by issuing a MOS INFO.
- Issue a MOS SERINTT n,b,p,d,s,h in the print processor task, where n is the port number, b is baud rate, p is parity, d is data bits, s is stop bits and h is handshaking.
- Issue a MOS ROUTE LPTx to COMn in the print processor task.

NOTES:

...

Print slowly?

To cure this problem try the following:

- If you have used the MOSADM TMF nn command, try reducing the value specified. To determine what the current value is issue the command MOSADM TMF.
- Issue a MOS NODIS.

NOTES:

Not allow you to print at workstation(s)?

To cure this problem try the following:

- Make sure all cabling is connected properly and to the correct ports.
- Issue a MOS ROUTE LPTx TO TERM in the ORIGINAL task that is associated with the workstation that has the printer attached.
- If it is a terminal, enter setup mode and make sure the printer port is configured correctly.

NOTE: The printer ports of terminals are considered light duty printer ports.
If you desire to use the AUX serial port of a terminal for printing,
you must set that port and the host computer port to use hardware
handshaking.

NOTES:

...

Mouse not work?

The PC-MOS environment supports only a Microsoft or compatible mouse. To activate please do the following:

- Make sure the port that the mouse is attached to is defined using one of the serial drivers. To check do a MOS INFO.
- Make sure the \$MOUSE.SYS driver is loaded as the last driver in the CONFIG.SYS file.
- Determine the number of port that the mouse is attached to by doing a MOS INFO.
- Issue a MOS MOUSE {port#},1200,2 or MOS MOUSE {port#},1200,3 in the task that will be using the mouse. Use the 2 for two button mouse and the 3 for three button mouse.
- Load the mouse driver MOUSE.COM in the task that will be running the mouse.

NOTE: The task should be in MOS NODIS and MOS TSR ON modes.

NOTES:

Experience a "CANNOT START TASK - MOSITASK" error?

This error should only occur during the boot process and is a result of PC-MOS not finding a resource or file that it needs during the initialization procedure.

- Check to see if the `$$$HELL.SYS` file is located on the root directory of the boot drive. If it is not, then copy it from the system diskette or the PC-MOS sub-directory.
- Check to see if Disk Manager was used to partition volumes greater than 32 Megabytes. If so, make sure the `CACHE=` command included the `/BPS` option (see the CONFIGURATION section in the PC-MOS USER GUIDE for the appropriate values).
- Check your hard disk or controller documentation to determine if they make use of DMA. If so, make sure one of the following parameters have been added to the `MEMDEV=` command:
 - `/E` - to set 64K buffer
 - `/M` - to set 32K buffer
 - `/P` - to set 16K buffer
 - `/X` - to set 8K buffer
 - `/D=nn` - to specify size of buffer (nn is size of buffer)
- Also, make sure the devices are not set to DMA channel 3.

NOTES:

Experience an "Internal Error #0 Unexpected error during function 00"?

This error occurs during the termination process of an application (INT 21h, function 4ch), when PC-MOS discovers that the application corrupted the areas of RAM that it was using to keep track of the task's memory allocation. This could have been caused from an application, TSR or device driver's incorrect use of RAM.

To correct:

- Cold boot the system and execute the application again. If the problem reoccurs, the application that you are executing might be corrupted.
- Rename the application and copy or reinstall from the original diskettes.
- If the problem still occurs, then the application is poorly written in the way it handles an out of memory situation and would probably cause a problem under DOS if there were less than 640K of RAM available on the computer. The problem will manifest quicker under PC-MOS, because it's easy to create a virtual machine with less than 640 K of RAM, where under DOS, this requires pulling chips out in most cases.

Most computers sold today come with 640K of RAM, therefore, the logic within applications which verifies that there is enough memory doesn't get exercised (if there is any such testing). If an application designed to expect a minimum of 450K of RAM is executed under PC-MOS in a 380k task, it might appear to execute properly, but it could very well leave PC-MOS' memory control blocks corrupted, which would then produce an internal error when the application terminated.

NOTES:

Experience an 'Internal Error #0 Unexpected error during function 31'?

This error is similar to the one previously mentioned with function 00 , but it only happens on termination of terminate and stay resident (TSR) programs (function 31).

An error occurs during the termination process of a TSR application (INT 21h, function 31h), when PC-MOS discovers that the application corrupted the areas of RAM that it was using to keep track of the task's memory allocation. This could have been caused from a MOS RESIZE command or an application that has been corrupted. To correct:

- Cold boot the system and try the application again. If the problem reoccurs, the application that you are executing might be corrupted.
- Rename the application and copy or reinstall from the original diskettes.
- If the problem still occurs, then the application is poorly written in the way it handles an out of memory situation and would probably cause a problem under DOS if there were less than 640K of RAM available on the computer. The problem will manifest quicker under PC-MOS, because it's easy to create a virtual machine with less than 640 K of RAM, where under DOS, this requires pulling chips out in most cases.

Most computers sold today come with 640K of RAM, therefore, the logic within applications which verifies that there is enough memory doesn't get exercised (if there is any such testing). If an application designed to expect a minimum of 450K of RAM is executed under PC-MOS in a 380k task, it might appear to execute properly, but it could very well leave PC-MOS' memory control blocks corrupted, which would then produce an internal error when the application terminated.

NOTES:

Experience an "Internal Error #0 Unexpected error during function 4b"?

This error normally occurs when an application is being executed and PC-MOS finds that it has problems in one of the following areas:

Freeing the environment
Freeing the program space
Closing the file handle

These errors can be caused from a corrupted application or incorrect FREEMEM configuration. To correct:

- Cold boot the system and execute the application. If the error still occurs,
- Check your CONFIG.SYS file to make sure the FREEMEM is not overlapping other devices, which can cause a corruption of the SMP.

NOTES:

•
•
•

Experience an "Internal Error #EE - re-grab overflow"?

PC-MOS will use one of two different methods to manage the interrupt vector table: the re-grab method or the load interrupt data table (LIDT) method. This error can only occur when the re-grab method is being used.

It occurs when the storage space that PC-MOS uses to track changes to certain interrupt vectors becomes full and more space is required.

The re-grab method is in effect when:

1. There is no memory management driver available.
2. A memory management driver is being used but a /I switch is included after it (e.g. memdev=\$386.sys /I).

To correct:

- Determine which application, TSR or device driver is responsible and discontinue its use.
- Allow PC-MOS to use the LIDT interrupt management method by loading the MEMDEV= command , without the /I operand , in the CONFIG.SYS file .

NOTES:

Experience an "Internal Error #E1 - unable to remove task"?

This error indicates that PC-MOS was unable to remove a task during a task restart logic (a CTL-ALT-DEL from a workstation, a carrier drop on a MODEM task that has CN=T in the \$SERIAL.SYS command or a restart from the MONITOR.COM program). This can be caused from a free memory conflict which has somehow corrupted the SMP.

To correct:

- Consult the documentation of hardware devices, such as LAN cards, EMS, intelligent serial cards, etc..., to determine where, between the 640K and 1 MEG space they are addressed. Make sure no conflicts exist among each of these devices or with the PC-MOS FREEMEM. If it does, you can correct the problem by selecting different addresses for the hardware devices or adjusting the FREEMEM command in the CONFIG.SYS file.
- Check the application that you are running to make sure it does not make direct writes to the areas that PC-MOS uses to keep track of task data.

NOTES:

Experience an "Internal Error #E3 - Unable to restart task"?

This can be caused from a malfunctioning hardware such as a locked keyboard on VNA or VGNA.

To correct:

- Power the system off and then back on to re initialize all hardware devices.
- Power off each workstation individually and then back on.

NOTES:

Experience an "Internal Error #E4 - Unable to return heap space"?

This error indicates that PC-MOS, during the task restart logic, was unable to free SMP resources allocated to a task. This can be caused from a free memory conflict that has corrupted the SMP or an application, TSR or device driver that is not handling memory correctly.

To correct:

- Consult the documentation of hardware devices, such as LAN cards, EMS, intelligent serial cards, etc..., to determine where, between the 640K and 1 MEG space they are addressed. Make sure no conflicts exist among each of these devices or with the PC-MOS FREEMEM. If it does, you can correct the problem by selecting different addresses for the hardware devices or adjusting the FREEMEM command in the CONFIG.SYS file.
- If not, check all applications, TSRs and device drivers to find which is causing the memory conflicts.

NOTES:

Experience an "Internal Error #E0 - terminal not in list" ?

This indicates that PC-MOS, during a terminal pam switch, could not find a specified terminal on its internal link list. This is very indicative of a system memory corruption.

To correct:

- Consult the documentation of hardware devices, such as LAN cards, EMS, intelligent serial cards, etc..., to determine where, between the 640K and 1 MEG space they are addressed. Make sure no conflicts exist among each of these devices or with the PC-MOS FREEMEM. If it does, you can correct the problem by selecting different addresses for the hardware devices or adjusting the FREEMEM command in the CONFIG.SYS file.
- If not check all applications, TSRs and device drivers to find which is causing the memory conflicts.

NOTES:

Experience an "Internal Error # - Cannot start task - MOSITASK" error?

This indicates that PC-MOS encountered a problem during the initialization of a task. A possible cause is a lock on the COMMAND.COM file resource.

To correct:

- Reinitialize the system, which will clear the file lock on COMMAND.COM.

NOTES:

Experience an "Internal Error # - Command processor cannot be found"?

This indicates that PC-MOS could not find COMMAND.COM..

To correct:

- Copy COMMAND.COM from the system disk to the root directory of the boot drive or the location specified in environment variable COMSPEC.

NOTES:

...

Experience an "Internal Error # - Error: closing \$\$\$HELL.SYS"?

This indicates that PC-MOS encountered an error while processing \$\$\$HELL.SYS. The most likely cause is a bad FREEMEM.

To correct:

- Consult the documentation of hardware devices, such as LAN cards, EMS, intelligent serial cards, etc..., to determine where, between the 640K and 1 MEG space they are addressed. Make sure no conflicts exist among each of these devices or with the PC-MOS FREEMEM. If it does, you can correct the problem by selecting different addresses for the hardware devices or adjusting the FREEMEM command in the CONFIG.SYS file.

NOTES:

Experience an 'Internal Error # - SMP corruption during device driver loading'?

This indicates that PC-MOS' SMP got corrupted while device driver is loading.

To correct:

- Make sure the SMP size is large enough to accommodate the device drivers.
- Make sure the device driver is not attempting to utilize more memory than it is allocated.

NOTES:

Experience an "Internal Error # - Error during INIT or open of standard drivers - error type X"?

This error normally occurs during the initialization of a standard system device (CON, PRN and AUX devices). This may indicate a corruption in the PC-MOS kernel file (\$\$MOS.SYS).

To correct:

- Rename the existing \$\$MOS.SYS on the root directory of the boot drive, then copy the \$\$MOS.SYS file from the original system diskette.

NOTES:

Experience an "Internal Error # - 3 MEMDEV interrupt # XXXX conflict"?

This error occurs when PC-MOS finds a vector set to Memory management which MEMMAN did not set. This indicates a corrupted kernel file (\$\$MOS.SYS) or MEMDEV driver (\$386.SYS, \$GIZMO.SYS, etc...).

To correct:

- Rename the existing \$\$MOS.SYS on the root directory of the boot drive, then copy the \$\$MOS.SYS file from the original system diskette.
- Rename the existing MEMDEV driver (\$386.SYS, \$GIZMO.SYS, etc...) on the boot disk and copy the original driver from the original diskettes.

NOTES:

Experience an "Internal Error #01 - Extended Service xx no longer supported"?

This error indicates that an application, TSR or device driver made an invalid extended services function call.

To correct

- Modify application so this call is not requested.

NOTES:

...

Experience an "Internal Error #0C - error resizing task"?

This error normally means that PC-MOS' SMP was somehow corrupted.

It might be difficult to pinpoint the exact cause of this problem, because the offending application, TSR or device driver might have done its memory littering long before the effects are noticed. When this error message occurs:

- Try to identify what processes you've been running lately that can reliably reproduce this failure. Introduce more frequent MOS RESIZE commands to try to identify the exact time when the memory corruption occurs.**

NOTES:

Experience an "Internal Error #59 - Invalid IRQ list"?

This error normally means that PC-MOS' SMP was somehow corrupted.

It might be difficult to pinpoint the exact cause of this of problem, because the offending application, TSR or device driver might have done its memory littering long before the effects are noticed. When this error message occurs:

- Try to identify what processes you've been running lately that can reliably reproduce this failure.

NOTES:

Experience an "Internal Error #58 - Invalid IRQ"?

This error normally means that PC-MOS' SMP was somehow corrupted.

It might be difficult to pinpoint the exact cause of this problem, because the offending application, TSR or device driver might have done its memory littering long before the effects are noticed. When this error message occurs:

- Try to identify what processes you've been running lately that can reliably reproduce this failure.

~~CONFIDENTIAL~~

• **Prevalence** = the proportion of a population that has a disease at a particular point in time

Journal of Management Education

Experience an "Internal Error #80 - error freeing heap in int 24"?

This error normally means that PC-MOS' SMP was somehow corrupted.

It might be difficult to pinpoint the exact cause of this problem, because the offending application, TSR or device driver might have done its memory littering long before the effects are noticed. When this error message occurs:

- Try to identify what processes you've been running lately that can reliably reproduce this failure.

NOTES:

1. The first part of the report is a summary of the work done during the last year.

2. The second part is a detailed account of the experiments carried out.

3. The third part is a discussion of the results obtained.

4. The fourth part is a conclusion.

5. The fifth part is a list of references.

6. The sixth part is a list of acknowledgments.

7. The seventh part is a list of appendices.

8. The eighth part is a list of figures.

9. The ninth part is a list of tables.

10. The tenth part is a list of symbols.

Experience a "DIVIDE BY ZERO error"?

This error indicates that during a PC-MOS calculation, a zero was the divisor. The most likely cause would be a corrupted application, TSR or device driver or invalid application configuration.

To correct:

- Try to isolate the process that was running at the time of the error and determine how the invalid calculation arose.
- Obtain a corrected version of the application, TSR or device driver and correct the configuration information which might be causing the erroneous calculations.

NOTES:

•
•
•
