3

Troubleshooting Techniques

Verifying a real problem, analyzing symptoms, and isolating and correcting a failure in the IBM PC can be a stress-filled hassle, or an enjoyable, rewarding experience depending on your understanding of the machine and your ability to properly troubleshoot computer circuitry. Whether you're a hobbyist, or a technician, this chapter will guide you through the troubleshooting process and provide you with analytical tools and tips that will help you analyze a problem, identify the failed part, and step toward the correct repair.

INTRODUCTION TO TROUBLESHOOTING

Imagine for a moment that you're in the midst of printing a lengthy analysis report when suddenly the printer halts, the screen display goes blank and your IBM PC ceases to function. What do you do? What failed?

This chapter is devoted to a subject we often wish we could pass off or ignore–trouble. Trouble is like a flat tire: no one wants one, but when it occurs we all wish we could fix it quickly and get the experience behind us. Knowledge and action are required to overcome trouble.

Integrated circuit technology is advancing rapidly. Logic gates on tiny chips of silicon are getting smaller and faster. This has been welcomed by all, but coming with these advances in microelectronics are more challenges to overcome in determining whether a chip, board, or computer system is functioning correctly and has been properly maintained. Faults can occur that are difficult to locate.

A fault is any physical condition that causes incorrect output when a circuit is exercised to perform a function. Faults can be classified as static or dynamic. Static failures include the stuck-at problems associated with open or short data paths in circuitry. These failures are typically catastrophic causing a system operation to end. Shorts can be described as electrical conduction in the wrong place. Shorts are typically caused by mechanical failure in a device or a solder bridge during improper repair. Opens are cases of no electrical conduction when it should be present. Electrically open inputs can affect the switching speed of a device. They can also degrade the noise immunity of the component. Other catastrophic faults include the wrong component installed on a board, an improperly installed component, a missing component, and dead or partially dead devices.

Dynamic failures include time-dependent errors such as the loss of signal quality causing a circuit output to reach steady state too late to be properly used by another part of the system. The symptoms of dynamic faults include devices operating too slow. This failure is seen in setup and hold problems, data and addressing problems, machine cycle-time instability, and interactive problems between components. In the logic gate, dynamic faults are seen in propagation delay problems–time delay in getting a signal from the input to the output. Flip-flops typically experience dynamic faults in their setup and hold ability to capture and hold data after the inputs have dissipated. Dynamic faults in memory occur in the data/address relationships where timing problems occur between the occurrence of valid data and address information. Even the 8088 central processing unit can experience dynamic faults in the cycle-time stability of the microinstruction function cycle. Component dynamic failures are more difficult to find than the static catastrophic faults. Locating static and dynamic faults will be covered in this chapter.

The most effective way to locate a failure in the IBM PC is to think the problem through just as the machine operates–logically. Imagine the computer system as a human body. The timing and the timing circuitry represent the heart. The CPU and related ciruitry are like the brain. Without the heart and brain, nothing works in the body. The keyboard and drives represent the eyes and ears. The display and printer act like the mouth. By viewing the computer system as a functioning body system, you can quickly determine which area is not working properly and home in on the malfunctioning part. Understand what should happen and compare the "shoulds," one by one, with what is really happening.

There are typically two ways to analyze electronic circuit failures: classical troubleshooting which incorporates localizing and isolating a failure using deductive reasoning and mental intuition; and brute force troubleshooting which uses flow charts and replacement of all suspected components. Both of these techniques will be addressed in this chapter.

CLASSICAL STEPS TO SUCCESSFUL TROUBLESHOOTING

Solving computer system problems requires application of the deductive technique called "troubleshooting". Effective and efficient troubleshooting requires gathering clues and applying deductive reasoning to isolate the problem. Once you know the cause of the problem, you can follow a process of analyzing, testing, and substituting good components for each suspected bad component to find the particular part that has failed.

The use of special test equipment such as logic probes, logic clips, digital multimeters, oscilloscopes, and logic analyzers are the technician's tools-of-the-trade to help speed the process. Good, deductive reasoning is used to isolate a failure to a particular group of chips, then circuit analysis is used to reduce the problem to a specific component. When a suspected circuit network is found, there are two ways to test the board: in-circuit testing and functional testing.

In-circuit testing treats the PC system board as a collection of parts. Testing is accomplished on each individual part as though it were all alone. In-circuit testing relies on the ability of the tester to isolate and test the board components separately. Test patterns are automatically applied to the inputs to each component on a board, and the responses are measured directly at the outputs. This is a good diagnostic fault isolation technique for locating catastrophic faults, but if not done properly, in-circuit testing can cause catastrophic failure or degradation of component performance. Improper application of this technique can produce high current density, local hot spots, and excessive voltages in certain parts of the circuitry. These problems can be minimized using dedicated layout strategies and design-for-test techniques.

A specific form of in-circuit testing in which every electrical network on the PC system board is accessed at the same time is the bed of nails technique. A bed of nails fixture is typically a box with rubber gaskets that form a tight seal with the printed-circuit board being tested. It has hundreds (or thousands) of prongs (nails) with spring-loaded probe tips. When the circuit board is closed inside the vacuum of the bed of nails box, the spring-loaded probe tips make solid contact with the solder side of the board. While this is a very useful test technique for printed-circuit boards, the development of the nail configuration, construction of the box, and generation of the test program that monitors all test points is very expensive and generally out of reach of most service centers. A subset of in-circuit testing is accomplished by most service technicians using logic probes, pulsers, and other test tools that probe various parts of the board circuitry.

Functional testing treats the PC system board as a single functioning entity. This test technique evaluates the board in an environment that closely emulates the system for which it was targeted. A subset of functional testing is found in the short programs that can be written to exercise certain functions in the system. When these programs are executed, test equipment monitor specific test points to determine the proper (or improper) operation at those nodes.

Another form of functional testing occurs when test patterns are applied at the inputs to a circuit and the outputs are observed and measured. The expected results are derived by simulating the test patterns against a good machine which serves as a model. This was the technique used to produce the voltages and waveforms noted on the COMPUTERFACTS schematics. When the same circuit stimulation criteria are used on a suspected bad circuit, the measured results can be compared with the results derived from the known good model.

The key here is that the test node must be observable and vitally involved in the function being done. A related requirement is that the measuring equipment must not introduce adverse reaction by the circuit under test. It must not load down or change the signal content at the test point.

For example, the output of a gate can be considered the output of a transistor configuration. The input of a gate can be considered the input to a transistor configuration. Transistor inputs characteristically have very high impedance; the outputs have very low impedance. Touching a point on the circuit board causes the tester to source or sink current. Most of the current drives toward the points of low impedance. This causes the output transistor to be back driven producing heat in the chip. The temperature of the output junction increases. As long as the thermal threshold of the transistor is not violated, no component damage occurs. The accepted temperature threshold for silicon transistors is typically 120 degrees Celsius. Therefore, probe tests in an active circuit should not be overused.

In general, there are some favorable steps that you can follow to achieve successful computer troubleshooting and repair.

1. Don't panic.
2. Observe the conditions.
3. Use your senses.
4. Retry.
5. Document.
6. Assume one problem.
7. Use correct service data.
8. Use the right test equipment.
9. Diagnose to a section (fault identification).
10. Localize to a stage (fault localization).
11. Isolate to a failed part (fault isolation).
12. Repair.
13. Test and verify.

The following pages discuss the steps to troubleshooting success in detail.

Every computer is composed of functional sections, as shown in Fig. 3-1. Any of these sections can fail.

When something functionally goes wrong in the computer, the first step is to determine whether the trouble results from an actual failure or from a loose connection or human error. To do this, you need to understand how the IBM PC works and how it interacts with the other parts of the system. Chapter 2 was written to fill this requirement.

Once you're convinced a true component failure has occurred, the next step is to determine

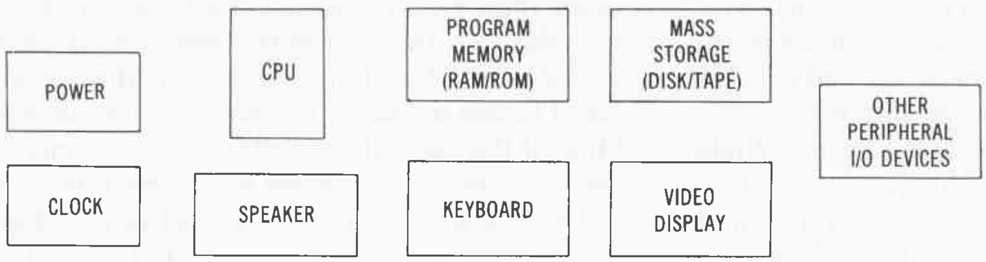


Fig. 3-1. The functional units of the IBM personal computer.

which functional section of the system is not operating–disk drive, keyboard, display, or some other part. To do this, break each section into stages and trace the trouble to a circuit stage within the section. If a display isn't working, for example, the problem could be in the display monitor itself, in the video cable, or in the video circuitry of the computer. Each of these can be considered a stage of the video display functional section. Then, analyze the circuit to isolate the failed part.

When troubleshooting a computer you must discipline yourself to check that the power is what you need it to be at that time (usually off). Make it a practice to always place your hand over the power switch whenever you first start thinking about doing something inside the computer.

Visual Inspection

There are specific steps you should take when troubleshooting an IBM PC. First, search out all the symptoms, the clues, that point toward the location of the failure. Make a visual and operational check of everything that is active during normal operation of the function that is failing. Look for misplaced or unconnected cables, power and other switches incorrectly set, disk drive doors inadvertently left open, unplugged wall sockets and bad disks. Look for anything that appears out of place.

Cleaning Connections and Inside the Chassis

Turn off the computer and clean all the edge connectors on the plug-in cards. Reseat the associated cables, making sure to look for bent pins. Examine the system board and related interface cards for discolored components or loose debris. You'd be amazed at what has been found inside computers, printers, and disk drives. During analysis, we've found pieces of bread, cigarette ashes, coffee spills on the keyboard, and even sticky soda pop all over the motherboard. Inside drives we've found everything from pencils to carrots to strange creatures. In one drive that was being repaired for a school district, we found a dead mouse! Don't be surprised by the things that somehow find their way into these machines.

After checking the internals of the computer, disk drive, and any other associated equipment, close the system up and reboot. Check to see if the same failure occurs. If it does, shift to symptom analysis.

Symptom Analysis

In this step in the troubleshooting process you examine symptoms. Carefully evaluate the problem to determine what area of the system is failing. If the screen on the display console is black and shows no sign of video life, then check the monitor and video circuitry and interconnecting cables. If the program locks up in the middle of an operation, the failure is most likely in the CPU, ROM, RAM, or related circuitry. If the drive doesn't boot, then the drive, the drive circuitry, the disk drive adapter board and the related interface cables become suspect.

When checking symptoms, remember to check for all symptoms, not just ones that seem directly related to the problem. Many times there are other clues available.

Diagnose to a Section

After you've considered all the symptoms, narrow the failure to the section in which the problem seems to be centralized. For example, lets say you have a PC in front of you whose disk won't boot in the disk drive. You completed the visual check and found nothing, so you analyze the failure symptoms. You notice that when power is applied to the system the computer turns on, and the self-diagnostic runs and passes.

Keep in mind that, even though the diagnostics passes, there can still be a problem in the circuitry that was tested. The PC diagnostic does not test all types of stuck at or improper signal conditions. In our example, the diagnostics run fine but when the drive tries to boot it just locks up and doesn't do anything and no failure code is displayed on the screen.

You turn the computer off and, several seconds later back on, closely watching the action of drive A. When the BIOS reaches the step to boot the disk, the drive light turns on and you can hear head movement inside. However, about 40 seconds later, the drive motor stops. The drive light is on, but no activity occurs. A curious fact. You try another known good disk. Same results. You swap disk drives and retest. Again, no change. You clean and reseat the drive adapter board. You also clean and reconnect the cable from the drive to the adapter board. You retest. The problem remains.

In our example, you remove the disk from the drive and power down and back up trying to boot the system up in BASIC. It does. Writing a short program verifies that the software works fine.

Now that the system has been proven to work without the drives, you conclude that the CPU, the system board, the screen, and the keyboard sections work properly. The problem has been reduced to the disk drive portion of the system circuitry. Closing in on the failure, you decide that you have a choice to make. If you have spare cables and adapter boards you can swap out one at a time until the problem goes away. However, if you don't have spares, you must investigate further. Look at the magnetic head inside the failing drive as you reapply power to the system. It should immediately move to the "home" position furthest from the center of the hub. This is the location of Track 0. Disk unique information such as where the directory can be found is located on this track. In this example, the head goes to the home position. Then the head moves across the disk just like it was reading data off a track cylinder, but 40 seconds later head movement stops with the head positioned half way across the disk. It's now time to break out the scope to take technical measurements.

You check the basic power and timing circuitry. Power and a good ground are present at the drive and adapter board. You check the speed and tracking of the malfunctioning drive. (Chapter 4 Preliminary Service Checks provides a detailed description of speed and tracking adjustments.) These are verified within specification so you shift into detailed troubleshooting.

Localize to a Stage

You check for open or shorted pins in the cable. Then you check the disk drive controller circuitry on the adapter board. The problem localizes to a circuit stage on the adapter board. Normally, you would have found the problem by now. (Chapter 5 covers detailed troubleshooting.)

Isolate to a Failed Part

You isolate and locate a failed part on the adapter board. Replacing the part, you verify proper system operation has been restored.

The specific problem described in this example is not important. The value of the preceding analysis is the troubleshooting steps that have been described. The first step is a visual check (including swapping disks with a known good disk). Next, the problem is reduced to a sub-system by checking symptoms. Then a preliminary check is conducted on the circuitry in question. This is followed by specific circuit analysis to isolate and identify the failed component.

UNDERSTANDING HOW COMPONENTS FAIL

While the use of troubleshooting equipment makes it easier to analyze and isolate different computer problems, many failures can be found using deductive reasoning and understanding. In fact, troubleshooting and repair can be relatively simple if you know how the system should operate and understand how electronic components fail.

Failures generally occur in the circuits that are used or stressed the most. These include the RAM and ROM memory chips, the 8088 CPU, and the input/output (I/O) chips between the motherboard and the peripherals. The CPU is a highly reliable device and seldom fails. Most failures involve the other chips. Except for the ROM chips which are programmed by IBM, most of these other chips are standard, off-the-shelf devices and are so common they've earned the nickname "jelly beans"–inexpensive, easy-to-replace products. They can be obtained at a low cost at almost any electronic store or from an IC distributor.

Integrated Circuits—Chips

A chip or integrated circuit is constructed out of silicon with some other tiny particles of metal (impurities) imbedded in specific positions in the silicon. By positioning the metals in certain ways, tiny transistors can be formed. Applying a voltage to specific locations on the chip allows the device to invert a voltage level (+5 volts, logic 1, to 0 volt, logic 0), and enable all sorts of logic gates (AND, NAND, OR, NOR, and so forth) to function. It turns out that these chips can be made with silicon/metal junctions so tiny that today thousands of transistors can be placed on one chip. A memory chip the size of a fingernail can hold over 470,000 transistors.

The problem for chip manufacturers is how to get voltages and signals in and off such a tiny chip. Very thin wires are used as inputs and outputs to the chip. These wires are glued or bonded to tiny pads on the chip. The other end of each wire is bonded to a larger pad on a supporting material (the big part of what we call the integrated circuit, as shown in Fig. 3-2).

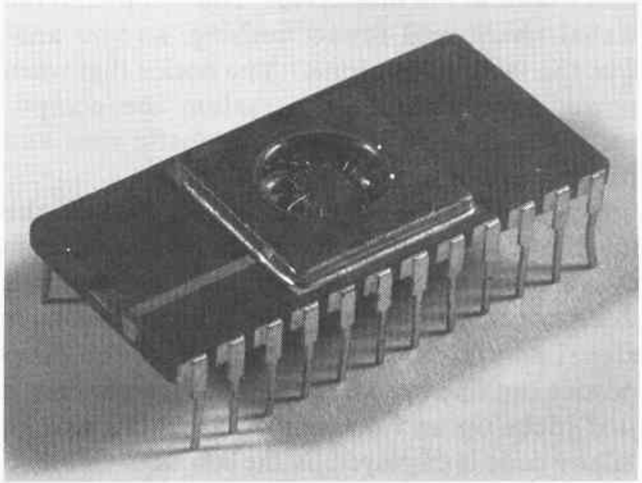


Fig. 3-2. Photo shows the chip pad to package interconnect leads.

The supporting structure includes the pins we plug into the sockets on our printed-circuit boards.

Integrated circuits are designed to operate for many hours, as shown in Table 3-1.

Table 3-1. Life Test Data for Various IC Technologies

Technology	Failures Per Device Hours
Bipolar	1 per 6 million
CMOS	1 per 55 million
NMOS	1 per 100 million
NMOS μPs	1 per 40 million
PMOS	1 per 10 million
PMOS μPs	1 per 35 million
8080 μP	1 per 16 million
4K SRAM	1 per 10 million

These tiny silicon and metal chips are placed in environments that put them under thermal stress during normal operation. Thermal stress affects those tiny strands of wire, or leads, going between the chip and the supporting structure including the large pins that are inserted into sockets. After a period of time, the thermal stress can cause the bonding of the wire lead to break away from the pad on the chip.

This disconnect causes an input or output to become an "open" circuit, and chip replacement is required. These interconnection failures are principally caused by electromigration. Silicon aluminum and copper aluminum were invented to replace the older pure aluminum interconnection wiring because they can tolerate higher current.

Another failure in these chips is caused by a phenomenon called "metal migration." The chip can be compared with an ocean of atoms. Some tiny particles of metal float about in this sea, migrating in directions perpendicular to electrical current flowing through the chip. Problems occur when these metal particles begin to collect in parts of the chip. If they concentrate in the middle of one of those microelectronic transistors, they cause the transistor to operate differently or not at all. If the resistance of these collected metals gets high enough, it causes the device to operate intermittently or to simply refuse to work. Since a transistor is part of a logic gate, the gate malfunctions and the output may become "stuck at 1" or "stuck at 0," no matter what the input signal is. Theoretically, a wearout failure won't occur until after several hundred years of use. We shorten the life span of our chips by placing them in high temperature, high voltage, or power cycling environments. These cause the devices to fail sooner.

Any event that acts to break down the oxide on a silicon chip can become a failure mechanism. Hot electron effects caused by injecting electrons into the oxide weakens the breakdown strength of the chip oxide. This also occurs over time (time-dependent breakdown).

In bipolar and MOS technologies, the most important failure mechanisms are metal corrosion and surface inversion. Half of the failed parts returned from the field exhibit these failure mechanisms. These problems can generally be attributed to poor assembly procedures and poor chip packages.

The second most important failure mechanism in ICs is caused by electrical overstress. Approximately one-fourth of the returned parts tested showed failures caused by operating the device in high temperature or high humidity. This was known to be a problem with MOS circuits, but now with the use of oxide isolation in bipolar circuits, these devices are also being adversely affected.

The remaining fourth of the failures found in returned parts showed physical malfunction caused by pinholes in the oxide and broken interconnections from electromigration.

In the manufacturing process, the major contributing aspect to failure are dust particles that cause mask and oxide defects. In fact, dust particles are believed responsible for 80 percent of all fab-line failures in today's MOS chips.

The I/O buffers have a higher failure rate than the internal logic cells because they experience higher currents and consume more power. On many bipolar parts, making them TTL compatible causes high power dissipation at the chip I/O. Other problems occur outside the chip, between the chip leads and the support structure pin leads, the device inputs or outputs.

These types of failures include: inputs or outputs shorted to ground, pins shorted to the +5 volt supply, pins shorted together, open pins, and connectors with intermittent defects. The most common IC trouble (assuming power is available) are opens or shorts to ground. Under normal use, chips finally fail with an input or output shorted to ground.

I/O pins and the bonding wires that connect the package pins to the pads on the IC die bonded inside the package have a higher failure rate than the devices on the chip because these interconnections are susceptible to environmental stress such as temperature variations, vibration, and improper handling and operation. Historically, only two percent of all IC failures occur in the silicon of the chip. Ninety-eight percent of all IC failures occur in the interconnects–the bonding wires and the leads coming off the package.

The classical stuck-at faults that came out of the diode-transistor logic era have been joined by nonclassical faults such as shorts between adjacent signal paths changing functions in an IC circuit. A stuck-open in a CMOS transmission gate or a reduction in transistor gain caused by ESD damage can cause slow gate switching.

Excess charge leakage or charge injection can cause slow independent change in a logic level. A high simply drifts down to a low.

Typical IC failure mechanisms are summarized in Table 3-2.

Table 3-2. Typical IC Failure Mechanisms

Chip Failure Mechanism	Package Failure Mechanism
Oxide faults	Broken wires
Oxide/junction contaminants	Lifted bonds
Diffusion defects	Gross deformation
Mechanical defects	Particulate contaminants
Metallization defects	Chip die separation
	Loss of hermeticity

Many failure mechanisms in ICs cause intermittent malfunction. These effects are typically caused by design specification margins that are too tight, fabrication line variations and contaminants, and random events such as alpha particle impact inside a package. Test methods that seek out classical stuck-at faults are not useful for chasing down intermittents.

Diodes and Transistors

The diodes and transistors on your computer's system board and adapter cards are made of solid material and act much alike. In fact, the transistor can be considered as partly constructed of two diodes.

Diodes are one-way valves for electric current, allowing current flow in only one direction. Diodes are usually made of either silicon or germanium. They are used in power supplies as rectifiers and in some circuits to maintain a constant voltage level. Other diodes are made of gallium arsenide and react by giving off light when biased in a certain way. These are called light emitting diodes or LEDs.

Transistors are used in various places in your computer circuitry as amplifiers or electronic switches.

Transistors and diodes fail by disconnecting inside, which causes an open or break in the circuitry, or by having their output short. Either kind of failure causes total loss of signal.

Diodes and transistors fail in the same ways and for the same reasons as chips, but chips fail more often than diodes or transistors. One reason is that there are many more tiny transistors on a chip the same size as a single (discrete) diode or transistor. This produces more heat and hence more thermal wear in the chip.

Capacitors

An understanding of the way a standard capacitor is constructed will aid in your understanding how these devices fail.

There are several types of capacitors on the PC system board and adapter cards. The capacitor is constructed of two separated plates. A voltage is placed across the plates and for a short instant, current flows across the gap. But soon electrons build up on one plate and cause the current flow to stop. The capacitor is then considered charged to some voltage potential. Capacitors are used to store charge and to filter unwanted signal spikes (sharp, quick peaks of voltage) to ground.

The electrolytic capacitor is constructed as shown in Fig. 3-3.

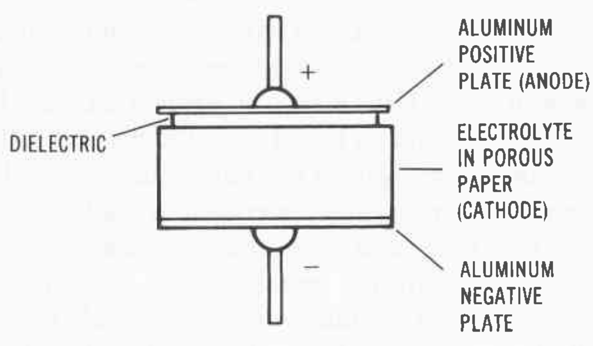


Fig. 3-3. A block diagram of the electrolytic capacitor.

Two aluminum foils or plates are separated by a layer of porous paper soaked with electrolyte, a conductive liquid. On one plate (the positive plate) a thin layer of aluminum oxide is deposited. This is called the dielectric. A capacitor has an anode (the positive plate) and a cathode (the electrolyte). Electrons build up on one plate causing it to become so negative that it prevents further current flow (remember that electrons have a negative charge).

Another type of capacitor found in the PC is the "film" capacitor. It is constructed of alternating layers of aluminum foil and a plastic (usually polystyrene) insulation. The metal foil acts as the plates and the plastic insulation acts as the dielectric between the plates. Film capacitors are coated with epoxy and have tinned copper leads.

Capacitors open or short depending on the operating conditions and on their age. Capacitors fail when they short internally or when one of the leads disconnects, causing an open. Again there is a loss of signal.

Electrolytic capacitors are especially susceptible to the aging process. One effect of aging is drying out of the electrolyte insulator. The capacitance value increases, and circuit performance decreases. Finally the capacitance value drops dramatically as the plates fold toward each other, and shorting of the plates can occur.

Another kind of failure occurs when some of the dielectric oxide dissolves into the moist electrolyte, causing the thickness of the dielectric to shrink. This deforming usually occurs when the electrolytic capacitor sits for a long time without voltage applied. Here, the capacitance value increases but a high leakage of electrons occurs across the plates, making the capacitor useless.

The leads of the capacitor can physically detach from its plate causing an open in the circuit.

Also, the plates can short together when a large area of one plate is stripped of its dielectric oxide layer by the application of too much voltage.

Resistors

These current-limiting, voltage-dropping devices are quite reliable and should function properly for the life of your computer. However, the features that shorten the useful life of the chips also act to reduce the operational life of resistors. High temperatures, high voltage, and power cycling all affect the materials of which the resistors are made. These stresses cause breaks in the carbon, resistive paste, or resistive layers producing an open conduction path in the circuit. Excessively high voltages can produce electrical current so large that it actually chars resistors to burnt ash. This is rare, especially in a digital circuit where the highest voltage seen is 12 volts (usually 5 volts) and the currents are very tiny indeed (milliamperes).

Resistor failures are almost always associated with catastrophic failure of some other circuit component. Resistor failures, when they occur, are usually located in printer electronics rather than in the IBM personal computer. Like capacitors, however, resistors in the PC digital logic circuitry will seldom fail. A momentary power surge through the power supply can "fry" a capacitor on your PC system board. Excessive temperatures can also damage resistors.

Resistors can absorb too much current and actually bake in the circuit. The result is usually an open circuit with shorting during the "meltdown."

All the devices mentioned so far are solid state. These components are built to rigid specifications and are constructed of materials (metals, plastics, oxide, and so forth) whose electronic performance changes as the components age. We can accelerate this process by producing excessive voltages in the circuitry, or allowing the system to operate without adequate cooling. Severe temperatures or high voltages can cause the device and the circuit or system to behave strangely. Fortunately, IBM PC motherboards are not exposed to high voltages. But they can get hot (especially if you plug a lot of cards in the expansion slots), and this will affect the operation of the components.

Anytime a computer is operated, the circuit components (especially the chips) are subjected to thermal stress. First they heat up when the machine is energized. Then they cool down when the PC is turned off, and they heat up when we turn the machine on again. This heating up and cooling off weakens the ICs and eventually causes failure. Thermal stress can cause a break in the connection of a wire leading from inside the chip to a pin, producing an "open" circuit, which requires chip replacement.

Even if no break in the chip or lead connection occurs, lengthy exposure to high voltages or temperatures can change the operating characteristics of a device. When the performance of these devices falls far enough below specs, the system starts to fail. A chip may work intermittently or simply refuse to work at all. An output can become stuck at 1 or stuck at 0, no matter what input signal is applied. Theoretically, a wearout failure like this won't occur until after several hundred years of use, but we shorten the life span of the chips by placing them in high-temperature, high-voltage, or power-cycling environments that can cause early failure. This is the reason that the computer finally fails after so many years of faithful operation. Its parts simply wear out.

Problems between the chip leads and the support structure pins which connect the device to the rest of the computer can cause failures such as inputs or outputs shorted to ground, pins shorted to the +5-volt supply, pins shorted together, open pins, and connectors with intermittent defects. Most failures result from opens or shorts to ground. Chips fail far more often than diodes or transistors, because the chips that are the same size as single (discrete) diodes or transistors contain hundreds or thousands of tiny circuits that produce more heat and therefore more thermal wear.

HOW DISK DRIVES FAIL

Disk drives give us the ability to save and load software at almost unbelievable speeds. These "boxes" are some of the most complex collections of electronics and mechanical hardware ever constructed. Thousands of tiny magnetic signals are stored on each disk inserted into one of these drives. We expect disk drives to save all our programs and data accurately and quickly and to accurately load the information back into our IBM PC without a single lost number or letter.

And they do. Disk drives will give you months of faultless service if you do your part, operating them carefully and providing tuning and periodic cleaning.

But sometimes users operate drives while puffing on a cigarette, tapping ashes onto a tray at the side of the drive. They smile as they jam a disk into the drive and then slam the drive door closed.

Then one day, that horrible DOS ERROR message appears and the drive "gives up the ghost." Now what? What kinds of failures can occur with disk drives?

The consistently heavy use of these electromechanical machines is the reason that the most common computer system problem involves the disk drives. Not only is this circuitry active much of the time, but any machine with mechanical movement, by definition, will require periodic alignment. The moving parts in disk drives gradually drift out of very strict operating specifications.

The typical drive failure is a change in the drive rotation speed. This affects the reading and writing of information on the disks. The speed is adjusted for approximately 300 revolutions per minute–200 milliseconds per revolutions. As the speed varies from this, disk read and write errors begin to occur. Drive speed and tracking must be within prescribed limits for proper disk storage and retrieval. Drive speed and tracking must be periodically checked and adjusted because age and movement can cause the settings to drift off. You shouldn't worry about relocating your system. However, when you do, lift and set down all the computer equipment carefully.

Even when the drive sits in one place and simply operates day in and day out, the drive and tracking can shift outside proper write/read specifications. Every time you energize the PC, the system runs a diagnostic test and then commands the magnetic head in the drive to seek the home (track 0) position. When it senses home, it moves back and forth from track to track writing or reading data. The movement of the mechanical parts in the drive can affect the tracking a tiny bit with each disk access. After a year of use, the tracking can be barely within limits. So we must periodically check and adjust the tracking and speed to keep them in "tune." The procedures for doing tracking and speed adjustments can be found in Chapter 4.

Rough handling in disk insertion and removal can cause misalignment of the read head. Misalignment is not an easy thing to fix. It usually requires special software and head alignment tools possibly including an oscilloscope.

HOW DISPLAYS FAIL

Most of us don't anticipate failure of a display monitor. Although monitors are like television sets, and we know from experience that sooner or later a TV will develop a problem and need repair.

Part of the reason displays still fail is that displays are the only new electronic device that still uses a vacuum tube. The cathode ray tube (CRT) is the screen you look at when you work with your computer–it displays video information. The CRT is probably the only modern electronic component that is guaranteed to wear out.

In Chapter 2, you learned how the letters and numbers you see on your screen are produced by electrons striking the back side of the screen. The electron streams get weaker as the CRT ages. You can correct some of the effects of age, but this requires knowledge and experience in television and monitor repair. Unless your're so trained, it's better not to open the display unit and expose yourself to those dangerous high voltages.

Here are some possible video display failures:

Short inside the CRT–can result in a "hum" noise and a bar across the screen, very poor contrast, a bright beam on the screen, or even diagonal lines on the screen.

Open or disconnect inside the CRT–no characters are displayed on the screen.

Bright "bloomy" letters; poor intensity control–caused by tube age. The center of the CRT has worn so that you can get normal brightness with the intensity turned down as far as possible, but black is really black, and grey shades are poor or not displayed.

Screen edge won't display; picture fuzzy–a deposit has formed on the inside of the screen causing reduced brightness and fuzzy display. The deposit is thicker at the outer edge of the CRT.

No picture–brightness and intensity controls have no effect.

Marginal performance–display monitor performance less than optimal. Monitors, like computers, printers, and other electronic equipment, are affected by dust and dirt. These pollutants coat the components inside the chassis and cause heat to build up. You know (now) what heat can do to your equipment.

In general, CRT failures cannot be corrected by anyone other than a trained service technician. The voltages inside the chassis of your monitor reach as high as 25,000 volts. These levels can be lethal if you make a mistake.

Unless you also repair displays, the only adjustments you should attempt are those that can be accomplished from outside the chassis. If you see holes in the back of the chassis for alignment, you'd be better off keeping out of these, too; but if you feel experimental, be sure you use a plastic alignment tool (it looks like a thin pen with screw-driver-shaped ends).

OTHER FAILURES

Liquid "Fry." This occurs when someone holds or sets a liquid on top of or too close to the computer and then accidentally spills the liquid into the top of the keyboard while the computer is running. It's a real mess to clean up, and you also get to replace many components.

Component Failure by Asphyxiation. This is caused by blocking the IBM PC vent openings or stuffing your computer with piggyback expansion boards that produce lots of heat without considering additional cooling. It "kills" components.

The Interface that Doesn't. This can be caused by improper connection of cables. Plugging cables in one pin off blows many chips. If cable connectors are badly corroded, no signal can get through the cable.

RFI Wipeout. Ribbon cables don't have much protection from radio frequency interference or magnetic fields produced around high voltage machines or even power cords. Printers may print garbage or not at all if the ribbon cable connecting the computer to the printer runs alongside or through a loop in a power cord.

So much for "other failures." If it can be done, someone has probably done it.

REPAIR GENERATED FAILURES

Some people simply have a knack for fouling up the works every time they try to "repair" something. They should take up reading instead of repair. Overzealous or under trained repair technicians, and technicians in a hurry or not understanding the system being analyzed can introduce more trouble than they can correct.

The following paragraphs describe some possible repair-generated "failures."

Bent or Broken Pins. Watch the way you put those chips in. You can only straighten those pins so many times before they break off completely.

All too often, pins on ICs that you are inserting into the board during repair bend. One of the most frustrating things to have happen is to desolder and replace an IC only to discover much later that you bent a pin under the chip shorting out etches on the board beneath the chip. This creates symptoms that cause you to believe something else is wrong in the system. Think about what would happen if you replaced a chip causing a speaker failure and now the drive won't boot. You could spend hours searching for a failure that you created yourself! To avoid this, be alert and careful when conducting repair. Above all, don't rush fixing the problem. Take your time and make sure the job is done right. Check your work after you have replaced something.

Electrostatic Discharge

Devices can be "blown-up" by improper handling. This problem occurs when someone picks up ROM, or CPU chips without first grounding any static electricity that a person might be carrying.

Many technicians improperly ground static electricity. People and objects such as desks and benches can accumulate a substantial electrical charge. Your body can actually accumulate static charges up to 25,000 volts. It's not unusual to build up and carry charges of 500 to 1500 volts. When you touch a computer or component inside the machine, the potential on you will discharge to ground. The electrostatic discharge will find the shortest path to ground. If it's through a chip, it can damage or destroy the IC. Many types of ICs are very sensitive to static electricity. A discharge of only three volts into a chip can cause malfunction and cause wild screen displays.

Each IC in your IBM was designed to withstand a certain amount of low voltage discharge. Latch-up and the destruction of the transistors in the chip can occur when excessive ESD is passed through it. To prevent ESD problems from affecting electronic circuitry, you should discharge any potential on your body to ground before touching anything inside the machine. This can be accomplished by touching a grounded area of the system such as the power supply case. The best technique is to use a ground strap attached around your wrist and connected at the other end to system ground. This frees you to move about and touch components without fear of zapping something. Some chips, such as the bipolar TTL ICs, are more suseptible to static than others. The chips produced in metal oxide semiconductor (MOS) technology are the most susceptible. These chips include:

- The 8088 CPU
- The ROM memory chips
- The 8237 DMA Controller
- The 8253 Programmable Interval Timer
- The 8255 Programmable Peripheral Interface
- The 8259 Programmable Interrupt Controller
- The 8284 Clock Generator
- The 8288 Bus Controller

Improper Soldering/Desoldering

Another self-generated failure can be caused by improper desoldering and soldering techniques.

> **Caution: never attempt to desolder or solder a PC board if you don't know how!**

Soldering and desoldering components is a skill that every technician should know. But, if you're a novice, for goodness sake, don't learn by practicing on a $500 system board. Practice on a board that won't cost much to replace.

Typical solder-related problems include leaving the solder pencil, or gun on the board so long it melts the etches and pads. This can introduce many more hours of repair work than you expected.

Solder "splashes" can also raise havoc on a component board. These are caused by using too much solder to fuse a connection. When you remove the pencil, a tiny ball of solder drops from the end of the soldering pencil right on top of the board, shorting out some of the circuit. Sometimes, a solder ball is so tiny, you may not even notice it falling into the circuitry.

Another potential problem with using too much solder occurs when you hold the soldering pencil to the pin being soldered and over-feed solder to the connection. The solder can flow through space in the hole around the pin and start building up on the other side of the board. It will flow through the board and then start climbing the pin. When the pin is all covered with solder, a big solder ball builds under the chip shorting the pin to other pins and board etches around it. This is a sure indication that the technician doesn't yet understand how to solder properly.

During one repair, a resistance test from one pin in the circuit to ground that should measure very low instead read nearly infinity. There was continuity across the component, but one side to the next component in the data path tested open. A visual inspection showed a bad solder connection at the component wire lead. The wire end was cleaned, retinned, and resoldered eliminating the problem and completing the repair.

In another case, many hours were spent alternately applying canned coolant and hot air to various suspected areas of the suspected circuit. The failure symptom remained. More time was expended pulling on wires and prodding joints with a plastic screwdriver blade, but to no avail. Finally, while moving one wire, the problem disappeared. The pins were resoldered correctly removing the cold solder joints and eliminating the problem.

If a component is desoldered during a test, or a new part is installed, make sure it is placed in the correct board holes and soldered carefully. Careless or improper desoldering and soldering can create troubles and multiply the difficulties converting a typical repair into a tough dog. Soldering and desoldering are covered extensively later in this chapter.

Installing the Wrong Replacement Part

I was recently told of an unusual problem that was observed by a service technician repairing a broken machine. During the visual inspection, the service technician noticed fresh solder flux around a diode and a nearby transistor on a circuit board. Someone had replaced the original diode and transistor in some vain attempt to correct the malfunction. Both the diode and transistor tested good by the technician. Then the technician noticed that the previous repair technician had installed a universal transistor in place of the device on the schematic and in the parts list. Replacing the universal transistor with the original required part didn't change the failure symptom. Since the transistor replacement was improper, the technician inspected the replacement diode carefully and discovered that this component was also incorrect. The correct diode was installed and the system was restored to correct operation.

Max Goodstein, in an article appearing in the April 1987 issue of "Electronics Servicing & Technology," described a difficult troubleshooting incident in which a malfunction occurred about 30 minutes after circuit activation. He sprayed the components in the suspected circuit with cooling spray, hoping to identify a heat-sensitive component. Cooling one IC seemed to help, so he replaced the IC, but the problem was unchanged. Mister Goodstein tested all the components in the malfunctioning stage both in and out of circuit. No bad components were found. Then he noticed two parts in a related stage that were connected to the suspected stage. Conducting ohmmeter tests of the resistors, capacitors, and diodes in this stage revealed a shorted diode. The diode was replaced, but the symptom remained. Removing a transistor from its mounting, he measured the forward and reverse resistances with a meter. He discovered base-to-emitter leakage in the transistor. Not having a direct replacement in stock, Mister Goodstein substituted a "similar" transistor into the circuit. The malfunction was not corrected. When he installed the proper type replacement transistor, the problem went away.

Be careful to use the correct replacement parts for failed components.

Improper Cable Hookup

I don't know of any technician who at one time or another hasn't incorrectly mated cables and plugs. How many of you have hooked up a printer to the wrong adapter card and later realized the error.

Another typical error is not making a tight cable connection. The interface looked fine, but only part of the signal got through. This also happens if you haven't kept the connectors clean and corrosion blocks signal flow.

Noise Interference

Signal noise can be caused when you place cables near sources of RF interference. This noise comes from being too close to CRTs, and power and interface cables. Don't place cables near the CRT or pass them through loops of a power cable. Interface cables are insulated from most noise, but you can defeat their shielding by placing them in an EMI or RFI field. This will produce symptoms of system failure. Noise interference is a major cause of intermittent failures.

DOCUMENTING YOUR PROGRESS

Technicians are often interrupted during their analysis. Any interruption that breaks your deductive thought process and pulls you temporarily away from the problem you are troubleshooting can cause the loss of hours of crucial analysis. By the time you return to the system, you've forgotten where you were in the process. To help you quickly refocus and again concentrate on the failure, write down key points as you proceed. Then when you're interrupted, you can return later and quickly refresh where you've been and continue where you left off.

HOW TO LOCALIZE FAILURES

There are two ways to localize failures and determine which computer part is broken: the software approach, and the hardware approach. Each approach is important in its own right, so each will be covered separately in the following paragraphs.

Software Approach

The software approach is a troubleshooting method used widely by most IBM PC repair technicians. As long as the disk drive will boot up properly, we can often find the failure using diagnostic software. As you know, your PC has a built-in diagnostic software program that checks out the machine each time you apply power. This program is well written and does much to ease your mind that all is well inside. More on this later.

Diagnostic Software

Watching strange things happen to a computer system can be frustrating. Often you can't be sure if you caused those weird characters on the screen or if your IBM PC is truly sick. You'd rather not start taking the system apart for failure analysis if the machine isn't really broken.

There is a way to gain confidence that the system is healthy and that the errors are probably in the software program you're trying to run. If the error is repeatable and the system drive still boots up, you can insert a diagnostic disk into your PC system and run a series of programs that test the condition of the computer. These self-test routines can give you a 95 percent or greater confidence indicator that your PC is working properly and that you need to check your software.

Diagnostic programs can also show possible faults before they become hardware problems. For example, some diagnostic software tells if the disk speed is too fast, too slow, or within a speed range where reading and writing data can occur without errors. These diagnostics measure the mechanical operation of your disk drives and are helpful in periodic preventive maintenance.

The success of self-test packages is measured by the level of confidence one can have that the component identified as bad by the software is indeed faulty. Some diagnostics are advertised as only 60 percent accurate; other companies say that their software test packages have an 85 percent confidence factor.

Most minicomputer diagnostics only identify faults to the board or module level. That's because customers in the large companies that own most minicomputers usually depend on the computer manufacturer's field service representatives for repair support. Here, the diagnostic is used as an improved user interface. The user can relay to the computer service center what the diagnostic tests have determined giving the field service technicians a quicker troubleshooting and repair visit. This is exactly the situation with your IBM PC. If it fails during diagnostic testing, a number is printed on the screen. This number is a key number to help you identify the bad part. (As Chapter 4 will describe, you can use this number to your advantage also).

Fortunately, most of the IBM PC microcomputer diagnostics can call-out faults to the chip level (especially faults in memory).

About 30 percent of all PC failures can be detected by diagnostic programs. Diagnostic programs can be bought from many of the popular IBM computer stores. The IBM advanced diagnostics is the best documented of them all. It comes complete with a hardware maintenance and service manual that contains a complete list of all the power-on self test error codes that could be displayed when energizing your IBM. There is a similar list in Chapter 4 of this book.

The IBM advanced diagnostic package also contains a switch configuration chart and a problem isolation chart to help you quickly get to the failing module. The diagnostics don't contain maximum stress tests for the floppy disk drive, the memory, or any hard disk drives. Therefore, if any marginal systems problems exist, this diagnostics will probably not catch it. The diagnostics software that comes with each IBM system is much like the Advanced version, but doesn't contain the option to format the hard disk drive, or the wrap plugs to test the asynchronous and printer adapter. In addition, little documentation is provided with the diagnostics that comes with the basic system.

Besides the diagnostic software provided with the machine, several companies provide diagnostic programs for IBM PC. These programs test main memory, system read-only memory (ROM), the CPU, the monitor, the keyboard, the disk drive speed, and many peripherals.

The most common diagnostic programs check the system random-access memory (RAM) and some of the input/output. Some routines check the operation of the CPU itself, but these usually locate only minor errors. It's difficult for a CPU like the 8088 to run a test on itself. Most diagnostics assume that the CPU is working properly.

Testing a microcomputer must begin with thorough analysis and test development related to each component in the design. The logic of this approach is that verification of satisfactory performance of each component must be assured before a test is conducted on the whole system.

There are three widely used methods for testing a microprocessor like the 8088 CPU. These are:

1. Actual use
2. Stored response from a known good system board
3. Algorithmic pattern generation

In actual use testing, a very limited subset of the CPU's capability is tested under generally ideal conditions. This test technique is satisfactory for noncritical applications, but is not suitable for critical applications such as braking systems and medical monitoring and controlling systems.

Storing a set of response vectors derived from testing a known good board also has some limitations. How was the reference system board verified good? Another limitation is the large amount of memory required to store simulation and response patterns.

Generating an algorithmic pattern gets around much of the large storage requirement. A test vector compaction technique such as signature analysis further reduces the memory requirements of the tester.

Without automatic test equipment, the CPU is usually tested by executing a software program that exercises the nonmemory parts of the chip such as the data path, the peripheral I/O logic, and the sequencer. During the execution of the program, the test results are compared to a pattern of expected results after every stimulus is applied to these functional blocks. Typically, an external tester monitors the output pins and compares the readings measured with stored expected values.

Testing the I/O Logic

The I/O logic in the IBM PC consists of several latches and transceivers that enable 8-bit data and 20-bit addresses to move about on the system board. A test program can be written that exercises individual ports in a predetermined manner. By monitoring the port, and knowing what information should be present on the pins, the output signals can be validated.

Testing the Interrupt Logic

The interrupt flag and interrupt enable latches are controllable and observable. Therefore, functional test patterns can be generated and applied to this circuitry via a test program written specifically for this purpose. Verification of the proper interrupt sequence can be easily made, but to check the priority logic, an external tester will likely be required so the output signals can be indirectly verified by checking whether or not a proper interrupt service routine was executed.

Bus Testing

The bus connects many devices on common interconnect lines. The first step in functionally testing a system board is to make sure the bus structure is free of defects.

Two types of problems are associated with a bus. First, problems occur with devices exhibiting leaky outputs associated with weak internal diodes. Second, input problems occur in connected devices that have internal shorts that overdraw current from the bus.

The IBM PC employs many buses for data communication, including the control bus, the operand and result buses in the data path, the memory address and data buses and several buffered buses on the system board and each of the peripheral boards. By controlling and observing a bus, all the logic connected to it can be easily accessed, improving their testability.

No special test points are needed to access the devices connected to a bus. In addition, all registers connected to a bus can be accessed by accessing the bus. One good access place on the system board is the expansion slot backplane. By inserting an extender card into one of these slots, many bus signals are readily accessible. By accessing one bus line at a time and observing the results, a bad bus line can be confirmed.

One technique for isolating bus output problems is to get the bus into a tristate condition with all bus-connected devices disabled. Then use a tester to pull each bus line both high and low. If the bus passes this test, the bus is good. If the bus does not pass this test, check each device connected to the faulty bus. Run that device to a pin state opposite the state in which the bus line failed. By measuring the difference in the amount of current required to bring the bus line back to the opposite pin state, a faulty device can be detected.

If the bus problem is an input problem, inject current into each device connected to the bus and detect the current flow on the bus. In this manner, input faults on bus-connected devices can be detected.

Memory Tests

Some memory diagnostics test to see if the computer is properly setting and clearing individual bits in memory and also if store or write operations are affecting more than one memory address location at one time. Other diagnostics test the permanent memory (ROM) by reading every location and then computing a final signature such as a checksum or a cyclic redundancy check code.

Both the read only and read and write memories on the IBM PC system board are tested during the boot-up process. Each type of memory test is a part of the ROM BIOS power-up program.

ROM Diagnostics–In the IBM PC, each ROM is tested using the checksum technique. Initially, the 8K ROM containing the BIOS is read and its contents summed. The final summation is compared to a stored value. If an error occurs, the system halts with an error message displayed on the screen.

The routine begins by disabling the NMI interrupts, initializing the DMA page register, disabling black and white and color video, setting the 8255 PPI A, B, and C ports for A and C as input and B as output, writing the 8255 command code register, disabling the parity check, setting up the data segment register in the 8088 to point to the ROM address, setting up the ROM starting address at E0000H, setting up the return address, and jumping into a ROS CHECKSUM routine.

Later in the start-up routine, a check is made for an optional ROM at address C8000H through F4000H. If one is found, a checksum routine is conducted on this ROM. Otherwise the program jumps to the next 8K ROM module and a checksum is performed on this chip. After all the remaining ROM modules are tested, the power-up routine continues (checks to see if a disk drive is attached).

RAM Diagnostics–Testing RAM memory is more involved than testing ROM because RAM requires the write operation besides the memory cell read.

The main memory tests assume the CPU is fine and go on to do some fancy tests on the RAM. This form of testing finds out if test data can be correctly loaded into one and only one location in memory. If a "storage error" occurs–that is, the test data stored is not the same as the test data–a message is printed on the screen. If the correct data gets stored but into several different memory locations at the same time, an "addressing error" has occurred and this too is noted on the screen.

There are many algorithms (routines) for testing memories. Typical RAM tests are machine language programs that carry out RAM write-read algorithms. RAM is tested by a conventional memory testing algorithm carried out as a software program that writes a pattern of data into memory, then reads the value out and compares it with an expected value. Because the test results are compared after each write-read operation, a RAM test takes time to complete. In addition, by carrying out a machine instruction write-read algorithm means that the RAM cannot be exercised at its full speed capability. Therefore, the test, while verifying the ability to write and read properly, does not guarantee that the same performance is assured at full operating speed. The following is a list of the most common memory tests:

Common Memory Tests
Simple Store and Read
Sequential Numbers Test
Rotating Bit Test
Walking Bit Test
Dual Address Test
Butterfield Test
Sum Test

A "simple store and read test" preserves a known value in every location in a selected block of memory. Then it reads the contents of each location to ensure that the value was correctly stored. It is a quick and easy rough test.

A "sequential numbers test" involves loading all the binary number combinations for an 8-bit word sequentially into a block of 256 memory locations. Then it starts at the first address location and reads out the data word stored, comparing it to the value that should be there. If the data is correct, the routine displays the words "all O.K." and the test moves on to the second location. If an error is found, the program displays an "error" symbol on the screen and the test starts over at the next (third) address location. The test repeats until you reset your system.

A better memory test, the "rotating bit test" checks each address location to see if a binary bit stored in any one of the eight positions in a binary 8-bit data word will falsely set another bit in the same word. This test starts by loading the binary number 0000 0001 in the lowest RAM address. The contents of this address are then read back out and verified. If the 0000 0001 was correctly stored, the bit is shifted left one place to 0000 0010 and the test is repeated. After the set bit (the "1") is shifted through all the binary combinations, stored in that same address location, read out, and verified, the entire test starts over at the next memory address location.

The "walking bit test" improves on the rotating bit test slightly. All 8 bits in a starting location are set to 0, or "cleared." Then the first bit is set to "1" (0000 0001) as in the rotating bit test. The program tests all seven other bits to see if they have changed from 0 to 1. Then the second bit position is set to 1 and all other positions to 0 (0000 0010). Again all seven other bit positions are tested. This process walks through each bit in that memory location setting each bit to 1 and testing all seven other positions.

Then the values are all reversed; all the cleared bits are set to 1 and the set bits are cleared to 0, and the entire process begins once more, but now as a rotating zero test.

This test is quite time consuming. Apparently, it can take over 13 hours to check a 16K-byte area of RAM. And it can take over 52 hours to test 32K bytes of memory! You can just imagine how long it would take to test a fully packed IBM PC.

A "dual-address test" provides a more thorough addressing check. Starting with the lowest memory address in a selected block of memory, the program stores all zeroes into the area (clears it to zero). It then stores all ones (1111 1111) into the first location and checks all other locations to see if any other memory address falsely received any ones. If all other locations are still "zero-loaded," the test location is cleared (written into with all zeroes) and the test shifts to the next higher address, storing all ones in this location and then testing all other memory locations. This test repeats until the program reaches the end of the selected memory area.

Jim Butterfield wrote a program that is a variation of the dual-address test and is in the public domain. In the "Butterfield test" program, all ones are stored in every location of the selected memory area. Then all zeroes are stored in every third address location starting with the first address. The algorithm then checks the contents of every memory address to make sure the values have been stored correctly.

Next, the program shifts the position of the "all zeroes" word twice using the second and then third locations in the memory as starting points. After the three-pass test using 0s in a memory field of all 1s, the bits are reversed and all 1s are stored in every third location of an all 0s memory field.

If an error is found, the program stops and the address of the error is displayed. If no error is detected, the program ends and the top address plus one is displayed on the monitor.

The "sum test" is probably the most sophisticated memory diagnostic test. It generates a unique data word for storing in each location of memory to be checked. The data word is the sum of the two bytes that comprise that memory address (recall that it takes 16 bits to address 64K bytes of memory; 16 bits is two 8-bit bytes). Since each succeeding address is one location higher, the value stored increases and each value is unique to an address. A variation on this scheme can be used with the 20-bit address word in the PC.

The algorithm then checks for correct value storage. If an error is found, the program displays the error and its location on the screen.

This diagnostic test is also time consuming. It's a good idea to run these types of dual-address tests on small blocks of memory rather than testing all the RAM. It has been determined that the testing time quadruples for each doubling of the amount of memory tested.

The RAM memory test in the IBM PC is a write/read/verify operation in which the patterns FF, 55, AA, 01, and 00 are written into the first bank of memory. After each write, a read is executed and the value fetched is compared to verify proper storage and retrieval.

Self-Diagnosis

There is a trend toward building diagnostic capability into peripheral equipment like printers and plotters. A strong incentive exists to place diagnostics in CRT displays, disk drives, and the personal computers, because so many of these devices are being sold.

Disk drives and printers function both electronically and mechanically. The electronic controller portion of these machines can contain their own diagnostics, and many controllers now do some form of self-diagnosis each time the system is powered up. These tests check for faults in the electronics.

Mechanical components are inherently less reliable than electronics, so peripherals containing mechanical parts need diagnostics that regularly check their internal operation. Most of the conditions monitored are operator related; for example, "paper out" or "ribbon out." Disk drive diagnostics measure mechanical parameters like speed and head alignment. We cover disk speed adjustments and head alignment in Chapter 4.

All the "canned" diagnostic packages use some version of the seven test algorithms described previously. Each diagnostic program is a valuable addition to your "troubleshooting toolbox," but no software diagnostic can help if your system won't boot or display. The message is: "There are many ways to skin a computer cat. Know them all."

Hardware Approach

In the hardware approach, troubleshooting tools are used to measure voltage (logic) levels in the circuitry of the IBM PC. These tools include the logic probe, the logic pulser, the current probe, the oscilloscope, the multimeter, the logic analyzer, and the signature analyzer. This approach requires a knowledge of electronics and test equipment.

Usually when a chip comes to the end of its useful life, a catastrophic failure occurs–it cooks itself internally. While your eye can't always see the chip defect, you can find the problem without much effort. (But, don't think that every time your IBM PC quits working, you've just had a catastrophic failure.) For those problems that are not easy to identify, let's refer again to our guidelines for success.

1. *Don't Panic.* You now have a manual that will help.
2. *Observe.* What are the symptoms? What conditions existed at the time of failure? What actions were in progress? What program was running? What was on

the display screen? Was there an error message?

3. *Sense.* Is there any odor present from overheated components? Does any part of the system feel overly hot? Do any of the components look charred or broken?
4. *Retry.* If the display is dark, check the brightness control, the power plug, and the power cord. Is the plug snug in the back of the computer? Is the other end of the power cord plugged into a wall socket? Is the wall socket working? If any of these isn't right, correct the problem and try again.

   If the problem involves an external display, the printer, or other I/O peripheral equipment connected to the IBM PC by cable, make sure the power to the system is off, disconnect the power plug from the computer, and then reseat all the connector cables associated with the failure. Cables have a habit of working loose if they aren't clamped down. Once you've checked the cable connections, reconnect the power plug, power up, and retry.

   If a disk didn't boot, try booting the disk in the other drive or try booting another copy of the program disk. You could also try booting the disk in another IBM PC computer. If you always use a copy of the program disk, any failure of a disk drive won't cause as much frustration if it destroys data on the backup disk as it would if the disk were the program master. If data is altered by a malfunctioning drive, the disk can be recopied again from the program master once the drive problem is resolved.

   If it still won't work, disconnect all the external equipment connected to the computer, and try to operate the system alone. Sometimes, the failure in a peripheral device appears in another functional part of the computer. If the computer works by itself, the problem is probably in the external device or in the connecting interface.
5. *Write.* Document all that you see and sense. Write down all the conditions that you observed at the time of failure, or when you verified a reported failure. Write down what conditions exist now that failure has occurred.

   What is the PC doing?
   What is it not doing?
   What is being displayed?
   Is there an error message?
   What is still operating?
   Is power still indicated on each part of the system?
6. *Assume one problem.* In digital circuitry, the likelihood of multiple simultaneous failures is low. Usually, a single chip malfunctions, causing one or more symptoms; however, if you've shorted something in the circuitry, all bets are off.
7. *Use correct service data.* You'd be amazed at how many small service centers are trying to run repair operations with little or no technical information on the equipment they claim to support. In one case, a major service center covering an entire state was conducting repair activities on a myriad of personal computers with a 20-page "technical manual" and one of my trouble-shooting and repair guides. The manufacturer's "technical manual" was so poor it was essentially useless. Because the micro maintenance series of trouble-repair guides are high level descriptions of personal computer equipment, the service technicians had no source for accurate measurements, waveforms, DC voltages, and in-depth technical theory of operations for the machines.

   Don't attempt repair without proper service data. Make sure you have the SAMS COMPUTERFACTS, this manual, and anything the manufacturer provides at your bench. The use of correct and complete service information can prevent moderately difficult repair jobs from becoming tough dogs. If you value your time, prepare before you repair.

8. *Use the right test equipment.* Just like the proper technical documentation, the right test equipment can change difficult repair jobs (dogs) into routine activities. Unfortunately, all too many small service centers are trying to conduct a repair business with old, substandard, uncalibrated equipment. How many of you have worked (or are now working) in a repair center that has little more than a VTVM and some hand tools. Going after a failure in electronic circuitry inadequately prepared is like tracking a rabbit through your carrot garden with your eyes blindfolded. Your actions make little difference and all the while that rabbit eats more and more of your garden away.

   The lack of essential test equipment such as logic probes, pulsers, and a good triggered scope makes a world of difference in computer repair. If good troubleshooting equipment is called for, nothing less will do. Visual examination of the condition of the machine and its operation can have great value (when they're successful). However, this won't locate serious defects. Therefore, limit the time for simple tests to a few minutes before you change to using a DMM, scope, or generator. Be careful you don't apply inappropriate troubleshooting resources to the problem.

9. *Diagnose to section.* If the system worked when the peripherals were disconnected, turn the power off and reconnect one of the peripherals. Power up and test. If the unit still works, turn the power off, and reconnect another peripheral. Power up and test. Follow this procedure until the unit fails. The built-in diagnostic tests are a big help here. Once failure occurs, you know what device and what interface section has the problem.

   If you disconnect all the peripherals and test the computer alone, and it still won't work, try to determine what section or division of the machine failed. Describe the failure in simple terms–drive B won't read a disk.

10. *Consult the symptom index.* Chapter 5 includes an index of the most common troubles with the IBM PC. It includes a section on system error displays. If any error codes are displayed, these self-diagnostic results can guide you to the correct area of the problem. If the symptoms that you see match a problem described in the "Troubleshooting Index," turn to the referenced page and follow the instructions under "Troubleshooting Procedure."

Caution: Any time you open the computer, ensure the power is off, and touch a metal lamp, or other grounded object, to remove any stray static electricity.

11. *Localize to a stage.* Turn off the power to the computer, and disconnect the power plug. Disassemble the computer as shown in Appendix D. Follow the troubleshooting steps and procedures in Chapter 5 to localize the failed stage.

12. *Isolate to failed part.* Closely following the procedures in Chapter 5 should guide you to the failed part.

Many things get in the way of proper system operation. Chips have a tendency to work themselves out of their sockets under normal operation. A loose RAM chip could be your whole problem. "Loose chips sink MIPS" (MIPS stands for millions of instructions per second–a measure of computer capability).

Replacement of chips that are in sockets may look easy, but there are some pitfalls you should be aware of. Those fragile pins on your chips bend easily, and it doesn't take very many straightening actions to break a pin completely off.

Removing and reinstalling chips that are soldered into the motherboard are actions that require more than a passing knowledge of soldering techniques. Only attempt this part of the test-repair procedure if you have experience soldering and desoldering multilevel printed-circuit boards. If you don't have the experience, get it–your job requires it.

Sometimes a problem is caused by noise. Not audible noise, but electrical noise, the kind that produces "static" on your radio. This noise also affects computers. Noise in the computer system can cause data to be lost or wrong data to be stored or displayed.

> Note: To avoid noise problems, keep cables clear and away from power cords, especially coiled power cords.

And it's appropriate to add, don't try out your new drill set next to your computer while computing the effects of your recent pay raise. Your calculations might prove unbelievable.

Intermittent Failures

Sooner or later you're going to be confronted with those once-in-a-while failures called "intermittents." These can be really frustrating. It's no coincidence that most "tough dog" problems are associated with intermittent symptoms. Erratic operation multiplies the difficulties.

Unlike a hard (constant) failure, an intermittent problem shows up randomly, or only at certain times (usually when you expect it least). Intermittent failures are difficult to handle using standard troubleshooting methods to obtain dependable, repeatable and easily interpreted results.

Since intermittent failures can be caused by shock, vibration, or temperature change, these conditions can be used to find and correct them. Here are some helpful hints regarding intermittent failures:

> **Caution: The following steps are conducted with the computer open and operating.**

Be careful not to short out any connectors or pin leads. Use only a nonmetallic or wood object to probe components inside an energized IBM computer.

a. Check, clean, and reseat all connector boards and cable plugs.

b. Tap gently at specific components on the suspected board using a nonmetallic rod or plastic screwdriver.

c. Heat the suspected area with an infrared lamp or hair dryer. Don't overheat it.

d. Spray coolant on a suspected component when investigating an intermittent failure. Several companies sell pressurized cans of coolant spray that have long plastic extender nozzles for pinpoint application By cooling the device with the computer energized, and operating the system, you can identify heat sensitive chips on the verge of total failure. A system with a heat-sensitive chip will begin working for a few moments after cooling until the chip heats back up and starts causing problems again.

e. After you've found the area where the problem is located, make sure the power is off, and use a strong light and a magnifying glass to look for small cracks in the wiring or solder connections.

If the problem is a marginal chip, replace the pesky rascal. Be sure your replacement chip is of the same logic family as the original (that is, replace 74LS74 with another 74LS74).

Good cleaning, pin and board reseating, and inside-the-case temperature control will prevent the occurrence of most random failures. Board reseating is not a problem on the PC since the boards can be secured down with screws.

The final method for fault isolation to a component is signal tracing. This technique will be covered shortly.

13. *Repair.* A disassembly and reassembly guide is located in Appendixes D and E.

It takes a little practice before you can remove a socketed chip without it jumping out, flipping in midair and sticking you right in the thumb or index finger with that double row of tooth-like pins. Fortunately, there are several devices that make the job much easier. These are the tiny screwdriver, or "tweaker," and the

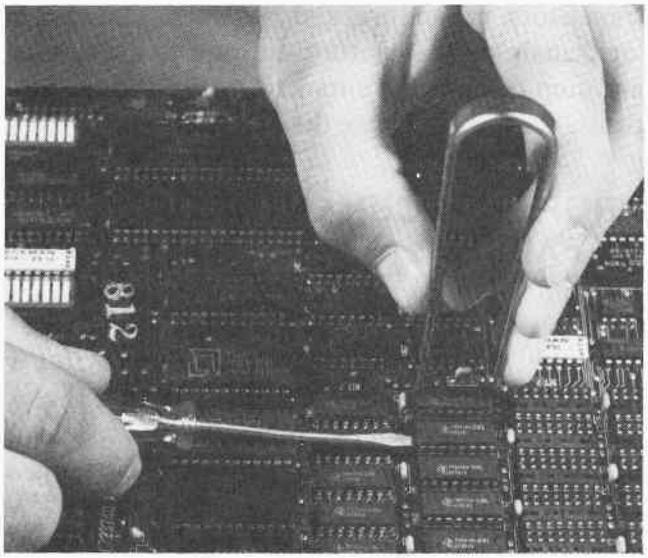


Fig. 3-4. ICs can be removed with a chip extractor or by gently prying up with a tiny screwdriver (tweaker).

IC extractor tool. Fig. 3-4 shows how each tool can aid in removing stubborn chips from their sockets.

Getting the chip out is only part of the repair challenge. Now you have to put the new chip in the socket. Here's how to do it:

a. Line up the pin-1 end (with the notch or dot) with pin 1 on the socket. (Notice how all the other chips around this socket are mounted.)

b. Place the chip over the socket, lining up one row of pins with its socket holes, as shown in Fig. 3-5.

c. With the chip at a slight angle, press down gently, causing the row of pins in contact with the socket to bend slightly, which lets the other row of pins slip easily into their sockets, as shown in Fig. 3-6.

d. Press the top of the chip down firmly to seat the chip completely into the socket. Be careful not to flex the board too much. If necessary, support the motherboard with the fingers of your other hand as you press the chip into place.

Now, that wasn't too bad was it? Well, it is pretty easy to make mistakes in chip replacement.

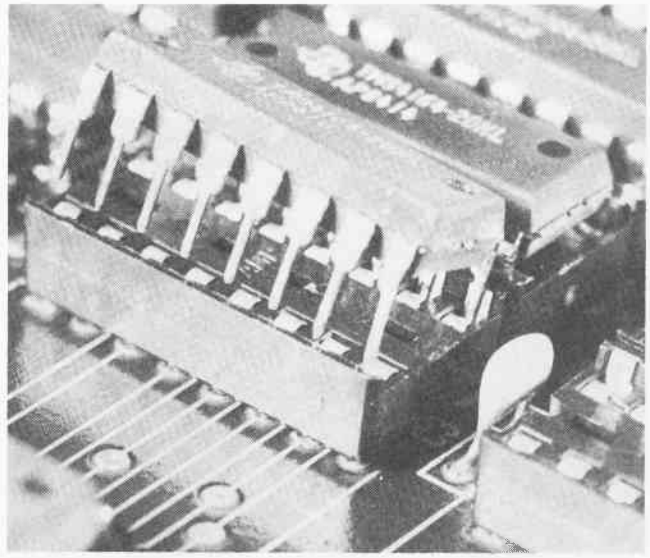


Fig. 3-5. Place the chip over the socket as shown.

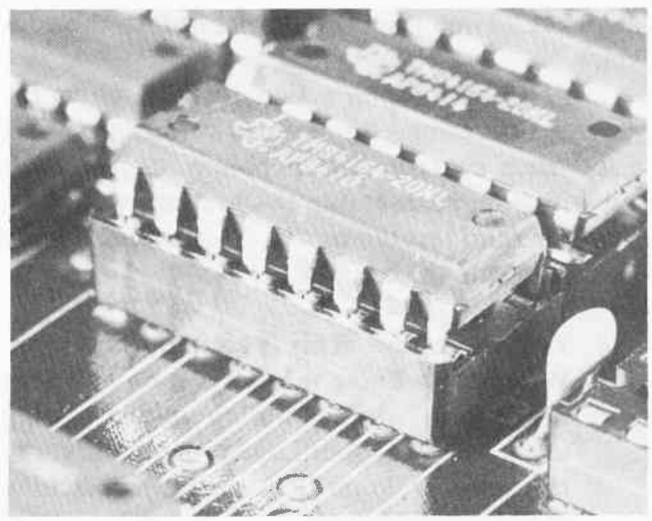


Fig. 3-6. Once each row of pins has been started into the socket, press down gently to complete the chip insertion.

- Make sure you don't put the chip in backwards. The notch or dot marking the pin-1 end of the chip is intended to help you correctly line up pin 1 on the chip with pin 1 on the socket.
- Don't offset the chip over the socket by one pin, as shown in Fig. 3-7.
- Don't force the chip down so one of the pins hangs out over the socket or is bent up under the chip.

If a chip to be replaced is soldered into the motherboard, always replace the chip with

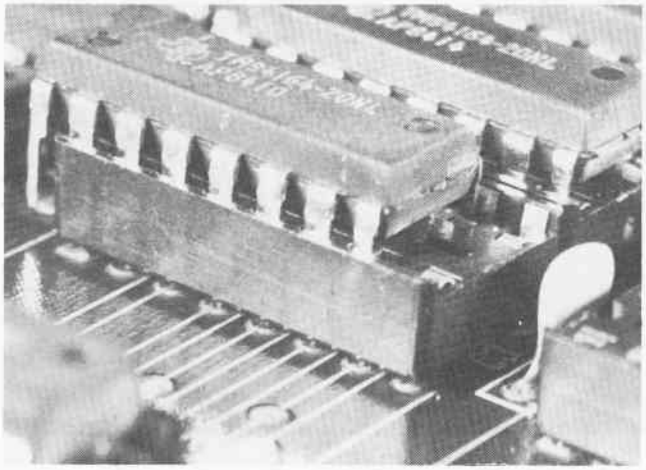


Fig. 3-7. Be careful not to offset the chip by one pin.

a socket. Then plug the new chip into the socket.

After each repair action, test the system for correct operation. Sometimes substitution of a good chip corrects the problem. After each substitution, reassemble the system enough to power up and test the repair. This process is very likely to locate the trouble.

14. *Test and verify.* This is an important step. We need to know that all is now well with the system. After booting up and testing using a copy of your DOS program disk, run the same program that was in the machine at the time of failure.

> Note: It's a good idea to log the repair action in a record book to develop a history of the maintenance conducted on the machine. Record sheets are included at the back of this book.

If these troubleshooting steps still don't help you find the failed component, hook up some test equipment, open the schematics, and start localizing the problem to a stage and then isolate the problem to a failed component. Try signal tracing with a logic probe and a logic clip. Use an oscilloscope and a digital voltmeter (DVM) to test the discrete components such as transistors, capacitors, and resistors. Connect a logic analyzer or signature analyzer to the system and step through the circuitry. Make voltage and resistance tests to locate the bad part.

PROPER DOCUMENTATION OF FAULTS

Of great value to the technician is a historical record of failures and repair actions. After a period of successful repairs, you will find that a pattern develops that can be referenced to speed up future repairs of systems with similar failure symptoms.

The following checklist is a guide to be used during and after repair. Each step is expanded here for clarity.

VALIDATING THE PROBLEM

1. What is affected? Find out if the problem is "catastrophic" and affects the operation of everything. If it affects only a part of the system (such as a disk drive), you may be able to swap drives and work on the bad drive away from the system so computer work can continue during repair.
2. Is the problem in software? Be certain it isn't. Try to run a program that you know is good.
3. Was the problem caused by operator error? Try a different operation that uses the same hardware or function.
4. Is it an intermittent failure? If your problem is intermittent, it could take several days for the problem to reoccur, and be found and fixed. You may just want to live with the problem until the intermittent becomes permanent (a hard failure). At least then you will have something concrete to troubleshoot.
5. Completely document the problem. Write down a complete description of the problem.
   - What was the system doing at the time of failure?

- What were you doing at the time of failure?
- What is the system doing now?
- What isn't it doing now?
- Is there an error code?

Maintain a written description for future reference and to build a fault history on your equipment.

6. Log the serial numbers of all the peripherals and the computer system.
7. Make a detailed listing of what was repaired or replaced.
8. Test run the system before placing it back in service.

RECOMMENDED SAFETY PRECAUTIONS

As with all devices that use or operate on electrical power, you must observe certain precautions to prevent damage to yourself or to the IBM PC system.

- Keep out of the display chassis.
- Be very careful when troubleshooting the power supply. If you're not experienced in high voltage circuitry, stay out.
- Turn the power off, ground yourself against static electricity, and pull the plug when doing anything inside the chassis except energized system troubleshooting.
- Handle diskettes carefully.
- Don't cycle the power quickly.
- Use a power strip to apply power to all components except for hard disk drives.
- Keep liquids away from the computer.
- Handle components with care.

Observing these precautions can save you time, money, and frustration. For your benefit, each point is expanded in the following paragraphs.

Keep out of the display monitor chassis. The voltages inside a monitor or television are dangerous, and only trained technicians should ever troubleshoot and repair a display unit. Voltages as high as 25,000 volts hide in there, so, unless you're experienced in high-voltage circuits, stay out!

Be very careful when troubleshooting the power supply. These circuits convert lethal 115-volt line power to the 5 and 12 volts used by the motherboard. IBM uses some special screws to secure the metal shield over the power supply. If you're not experienced in power supply circuits, stay out!

Always turn the power off, touch a grounded metal object like a desk lamp, and then pull out the power cord before touching anything inside. Before connecting test equipment to a circuit under test, turn off the power to the system. Failures are caused by people who don't follow this rule.

Handle diskettes (often called disks) carefully. Don't write on a label once it's attached to the disk jacket. Don't lay disks on a dusty, dirty surface. Keep cigarette ash away from disks and the computer system. Don't touch the disk surface. Don't try to see how flexible a floppy disk is. Don't set your disks on, or in front of, a TV or color monitor. Keep them out of strong magnetic fields.

Don't cycle the power on and off quickly. Wait 7 to 10 seconds to let the capacitors in the power supply discharge fully and the circuits to return to a stable (quiescent) condition.

Use a power strip. This saves wear and tear on the PC's on/off switch. Most power strips also have a built-in overload protection for voltage spikes. Voltage spikes can harm your computer system. (Don't connect a hard disk drive to the same power strip if it must be energized and up to speed before the computer gets turned on.)

Keep liquids away from the keyboard. I once had the opportunity to help a friend who's son had spilled a soda on the keyboard. It's amazing how sticky soda becomes after frying components all over the inside of the keyboard.

Handle components with care. Don't let chips lie around. The pins will get bent. If you lay a chip on the bench and forget about it, you may later lean over and stick all the pins on the package in your arm. Not dangerous, just hurts. What is dangerous (to the chip that is) are indiscriminate handling of MOS or CMOS parts. Watch out for static electricity–chips may need "special handling."

SPECIAL HANDLING

Some logic devices require extra care when you touch or handle them. You have no problem removing or inserting TTL (74xx series) chips into circuit boards. But the metal oxide semiconductor (MOS) chip family (MOS, CMOS, NMOS, and so forth) need extra care because they're more susceptible to static electricity than TTL. There are some 74HCxx chips being sold today that are CMOS, but these aren't currently used in the IBM PC computer.

Don't be afraid to touch the chips in your computer. Most guides for handling MOS chips lean far toward the super-safe zone and sometimes cause more problems than they prevent. However, these chips can be damaged by the static charge you can build up by scuffing your feet across a carpet; so be sure to ground yourself by touching a metal lamp or grounded object before you reach for a chip inside the IBM PC chassis. In addition, conductive foam provides static charge protection during storing or transporting of MOS-type chips.

Additional precautions should be observed when you use test equipment with your IBM computer. Turn off the system power before connecting or disconnecting test equipment such as oscilloscopes or logic analyzers.

ADVANCED TROUBLE-SHOOTING TECHNIQUES

In this section you will learn advanced troubleshooting techniques and become more familiar with the repair technician's "tools of the trade."

Tools of the Trade

When the problem can't be solved using flowcharts and pictures, repair technicians reach for help–they reach for their "tools." These tools are not only the tiny screwdrivers (tweakers), the diagonal cutters (dykes), and the soldering pencil. They also include some electronic test equipment–the various measurement meters (VOM, DVM, DMM), logic probes, logic pulsers, current tracers, clips, oscilloscopes, and logic signature analyzers.

Meters

Electronic measurement equipment has improved a great deal over the years, markedly improving your ability to test and locate circuit troubles. Twenty years ago, a meter called a VOM (volt-ohm-millimeter) was used to measure the three parameters of an electric circuit: voltage, resistance, and current. Then came the VTVM (vacuum-tube voltmeter). It wasn't long before electric circuits made room for electronic circuits, where digital was replaced and new meters appeared for troubleshooting using some of this new capability in their design. The DVM (digital voltmeter) and DMM (digital multimeter) quickly became the preferred measurement devices for digital technicians because they offered capabilities better suited for electronic circuit testing, including increased accuracy. These meters have characteristically high input impedances (resistances) so don't load down or draw down a digital circuit where the voltages and currents are far lower than those found in analog circuits.

Two changes affected the types of tools used in troubleshooting and repair. First, vacuum tubes were replaced by solid-state devices such as transistors and the integrated circuit (IC), or chip. Second, circuits themselves became smaller with more components packed compactly into less board area. One need only compare the early radios and televisions (standing 4 feet tall and weighing 40 pounds) with the wrist radios and now the wrist televisions of today to recognize that electronic circuits are smaller, more complex, and more difficult for test-probe access.

Electronic advances always lead to electronic opportunities, and clever test equipment designers soon came up with devices that enabled digital circuit testing without fear of inaccurate readings caused by circuit overload, or circuit failure caused by bulky test probes shorting two pins or wires on a packed printed-circuit board.

Logic Clip

One digital circuit testing device is the logic clip shown in Fig. 3-8.

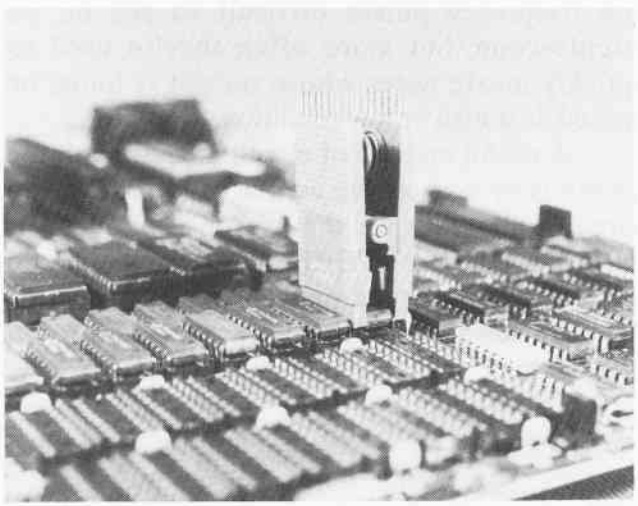


Fig. 3-8. A popular type of logic clip. *(Courtesy Pomona Electronics Division of International Telephone and Telegraph Corporation)*

> **Caution: When using a logic clip, turn power to the circuit off, attach the clip, and then turn power on. (This helps prevent accidentally shorting out the chip.)**

This handy tool fits over an IC and has exposed pins at the top. Measuring or monitoring probes or tiny clips can be attached to the pins to determine the logic level on any pin of the device under test.

Another type of logic clip has a built-in monitoring capability (Fig. 3 9). Instead of exposed pins, the top of the clip is lined with two rows of light emitting diodes (LEDs) which continuously display the logic condition of each pin on the chip. The LEDs are turned on (showing a Logic 1) by power from the circuit under test. All the pins are electrically buffered so the clip doesn't load down the circuit being tested.

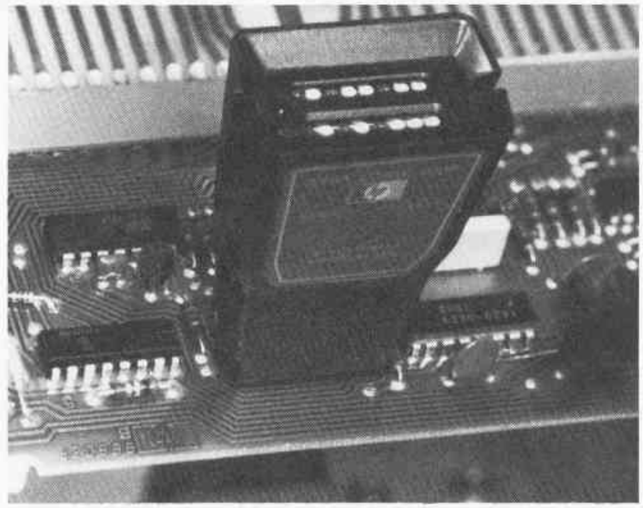


Fig. 3-9. A logic clip that gives visual indication of the logic condition at each pin. *(Courtesy Hewlett-Packard)*

Logic clips can be obtained in several varieties–to work with almost all logic families, including TTL and CMOS–and in voltages up to 30 volts DC.

To use the clip, squeeze the top (LED) end to spread the pin contacts, and slip the clip over the top of the chip to be tested. When power is applied to the circuit, the LEDs will show the logic level at each pin on the chip.

Logic clips can be used on ICs with up to 16 pins, or 80 percent of the ICs on your IBM PC system board.

Logic Probe

When you want to really "get into" your circuit, you can use a logic probe. A blown chip can't be repaired, but the logic probe can tell you which chip has failed so you can replace it.

The logic probe shown in Fig. 3-10 is the most widely used tool for this type of analysis. It can't do many of the things complex test equipment such as logic analyzers can do; however, the high frequency of chip failures in electronic circuits, the simplicity of the probe, and the ability to rapidly troubleshoot in an energized circuit make this tool ideal for 90 percent of your fault isolation needs. The SAMS COMPUTERFACTS on the IBM PC contains extensive logic probe values for use in analyzing section and stage circuitry.

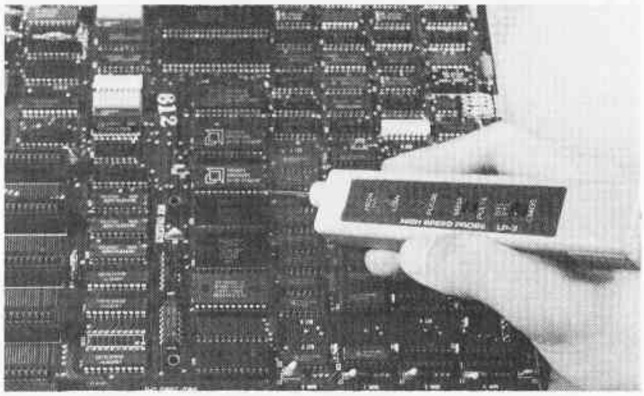


Fig. 3-10. The logic probe is the most widely used tool for circuit board analysis.

When the tiny tip of the probe is placed against a pin on a suspected bad chip, a test point, or even a trace on a circuit board, an indicator light near the tip of the probe tells you the logic state (level) at that point. The metal tip on most logic probes sold today is protected against damage from accidentally touching a source of higher voltage (up to 120 volts AC for 30 seconds) than that of logic gates (+5 volts).

Some probes have two lights built in near their tips–one for logic high and the other for logic low. The better probes can also tell you whether the test point has a pulsing signal present. They can also store a short pulse burst to tell you if a glitch or spike has occurred at that point. If you're planning to buy a logic probe, be sure it will work with the logic families you plan to analyze.

The ability to touch a point with the probe tip and directly determine the condition at the point for diagnostic analysis, and the ability to store pulses make this device easy to use and universally accepted as the proper diagnostic tool for all but the most complex digital troubleshooting. Other tools force you to attach the measurement probe and then look away at some display to read the condition. The logic probe displays the condition near the tip of the probe itself.

The logic probe in Fig. 3-10 provides four indications:

- Lamp off for logic low (logic 0)
- Lamp on bright for high (logic 1)
- Lamp dim for floating or tristate
- Lamp flashing for pulsing signals

Power for the probe comes from a clip attached to a voltage point on the circuit under test. Another clip attaches to ground, providing improved sensitivity and noise immunity.

Probes are ideal for finding short-duration, low-frequency pulses difficult to see on an oscilloscope, but more often they're used to quickly locate gates whose output is hung, or locked, in a high or low condition.

A useful method of circuit analysis with the probe is to start at the center of the suspected circuit and check for the presence of a signal. Using the COMPUTERFACTS schematic and Chapter 2 as a roadmap, move backward or forward toward the failed output, as shown in Fig. 3-11.

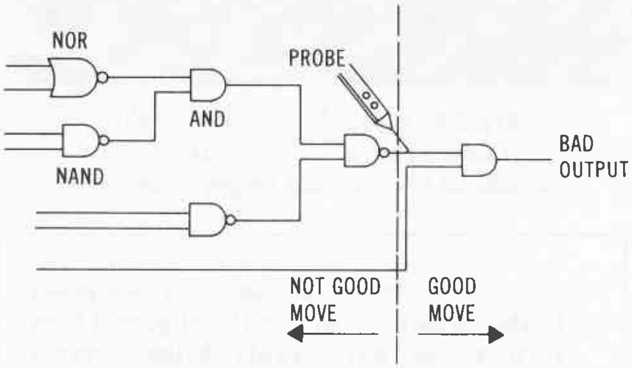


Fig. 3-11. Problem analysis starting at the center of a suspected circuit.

It doesn't take long to find the faulty chip whose output isn't changing.

The only limitation of logic probes is their inability to monitor more than one line.

Logic Pulser

If the circuit under test doesn't have a pulsing or changing signal, you can inject controlled pulses into the circuit using a logic pulser (Fig. 3-12). These handy devices are portable logic generators.

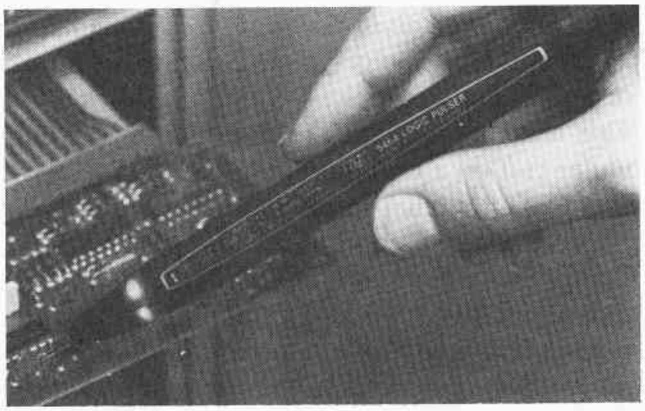


Fig. 3-12. A logic pulser can be used to inject a signal into a circuit. *(Courtesy Hewlett-Packard)*

When activated by a push-button or slide switch, the pulser will sense the logic level at the point touched by the tip and automatically generate a pulse or series of pulses of the opposite logic level. The pulses can be seen on an LED lamp built into the handle of the pulser.

The ability to introduce a changing signal into a circuit without desoldering or cutting wires makes the logic pulser an ideal companion to the logic probe and logic clip. These two tools used together permit step-by-step stimulus/response evaluation of sections of a circuit.

Figure 3-13 shows several ways to test logic gates using the probe and pulser. Assume the output of the NAND gate remains high. Testing inputs 1, 2, and 3, you find them all high. This condition should cause the AND gate output to go high, producing a low out of the NAND gate. Something is wrong. Placing a probe at the AND gate output, you discover the level is low. It should be high. Now, which gate is bad?

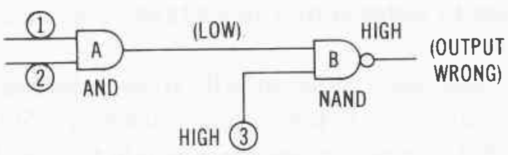


Fig. 3-13. There are several ways to test logic gates like these.

To find out, place the probe on the NAND (gate B) output and the pulser on the AND (gate A) output (NAND gate input), as shown in Fig. 3-14. Pulse this line.

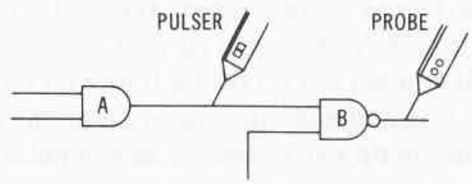


Fig. 3-14. Place the probe on the NAND gate output and the pulser on the AND gate output.

The probe should blink, indicating a change at the input to the NAND. If it doesn't blink a change, the NAND may be bad. But what if the low was caused by a short to ground at the AND output or the NAND input?

Place the probe and the pulser on the AND output trace, as shown in Fig. 3-15, and pulse this line. If the pulse blinks, the NAND is bad; its input changed state, so its output should have changed state also.

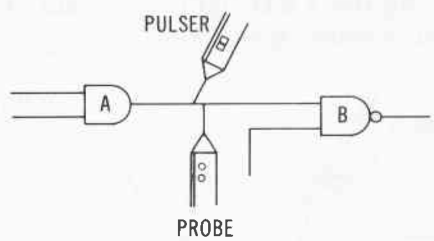


Fig. 3-15. Place the probe and the pulser on the AND gate output.

If the probe doesn't blink, you know this line is shorted to ground. One way you can determine which chip is shorted is by touching the chip case. A shorted chip gets hot, while a chip hung at one level seems to be normal but just won't change state.

Current Tracer

A fourth handy troubleshooting aid is the current tracer probe. This portable device lets you precisely locate shorts on the computer's system board (or peripheral card). The current tracer senses the magnetic field produced by the flow of electrical current in the circuitry. The logic pulser can be used to generate a pulsing signal that will make the current tracer LED blink, indicating the presence of current.

If you set the tip of the tracer on a printed-circuit line and slide the tracer along the line, an LED in the tip end of the tracer will pulse as long as there is a current present. When you slide past a shorted point, the lamp will go dim or out, and you've found the short.

Figure 3-16 shows an easy way to determine which gate has the short to ground in a logic circuit. Assume gate B has a shorted input. Place the pulser and the tracer midway between the two gates. Adjust the LED in the current tracer so that it just lights. Pulse the line as you place the tracer on the output of A and then on the input to B. The gate with the short to ground will pulse brightly because most of the current is going to ground here. Therefore, the input to B causes the tracer lamp to pulse brightly, while the A side of the line doesn't cause the LED to light. Following the LED light with your tracer will lead you to where the current is going.

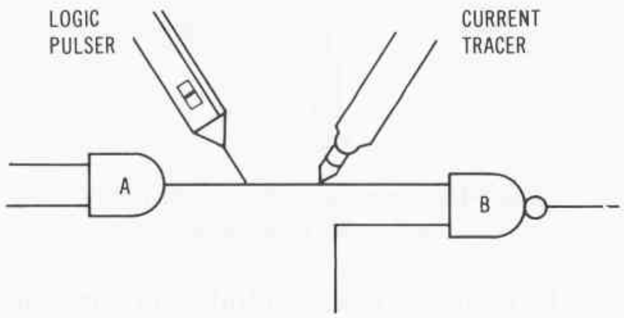


Fig. 3-16. Following the LED light with your tracer will lead you to the short.

IC Testers

Advanced troubleshooting equipment is becoming very sophisticated (and expensive). Today, you can buy equipment that tests almost every chip in your system for between $1000 and $2000. For $10,000 you can even conduct your tests from a remote location.

Micro Sciences, Inc., in Dallas, Texas makes an IC tester that can test over a hundred 7400 TTL and 4000 CMOS series devices. Options for this tester include RAM and ROM tests.

Microtek Lab in Gardena, California makes a tester that can do complete functional pin tests of all 900 devices in the 54/74 TTL series chips. This test tool displays the condition of the chip under test on a liquid crystal display (LCD). It uses LEDs to signal go/no-go test results.

VuData Corporation in San Diego, California markets an in-circuit component tester that's actually a 50 MHz CRT to display the voltage versus current characteristics for virtually all circuit components including capacitors, diodes, integrated circuits, resistors, and transistors. With their tester, the condition of the component under test is determined by the shape of the CRT display. Using this test machine, you can easily pick out open circuits, shorts, leaky diodes, leaky transistors, and marginal ICs. The tool is valuable because it can test a wide selection of components while they're still mounted in the circuit.

Oscilloscope

The oscilloscope has been called the "eyes of the technician." Scopes have been with us for years, although recent advances in the state of the art have added a great many capabilities to the instrument.

Simply put, an oscilloscope (Fig. 3-17) is an electronic display device that draws a graph of signal voltage amplitude versus time or frequency on a CRT screen. By graphically capturing the features of a signal, a scope can be used to analyze the quality and characteristic of an electronic signal. Its interface to the printed circuit board is a probe that touches a test point in a circuit. It can also be used as a measuring device to determine the voltage level of certain signals.

Scopes come in all sizes, shapes, and capabilities. Prices vary between $500 and $20,000. Some scopes use a single test probe for displaying and analyzing a single trace signal. Others have two probes and display two different

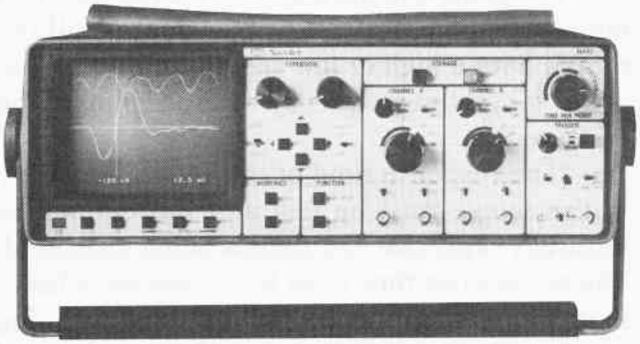


Fig. 3-17. A digital oscilloscope suitable for troubleshooting IBM PC failures.
(Courtesy Nicolet Oscilloscope Division)

signals (dual trace) at the same time. As many as eight traces can be analyzed at the same time on some oscilloscopes. In fact, in 1983 the first seven-color digital scope was introduced by Test & Measurement Systems Company. Colors make it possible to rapidly compare signals at different locations in the circuitry. Some scopes even have built-in memories to let the machine store a signal of interest for future evaluation.

Besides sensitivity and trace display, one of the major distinguishing characteristics of oscilloscopes is that they allow a great range of frequencies to be observed on the CRT screen as frozen images. This range is called bandwidth. Bandwidths vary between 5 MHz and 300 MHz, and price is proportional to frequency.

Oscilloscopes are useful tools for freezing an analog or varying signal and displaying this static waveform on the face of a CRT screen covered with a measurement grid. While it is time consuming to learn how to use an oscilloscope, the analytical rewards are substantial. Not only can you measure voltage amplitudes and frequencies of test signals, you can also accurately resolve small time intervals and measure delay times, signal rise and fall times, and even locate the intermittent glitch.

Graticule waveform analysis requires careful positioning to help identify the vertical and horizontal intervals of the signal, interpolation of the readings, and calculating the resultant values. Most scopes today have overcome the limitations of graticule measurements by providing a digital readout of the signal based on the position of special screen cursors. These digital scopes have the same basic measurement accuracy of analog scopes with crystal-controlled clocks, but cannot match the analog scope's time resolution and bandwidth (especially for intervals less than 1 microsecond).

One of the most exciting test products introduced recently is the digital storage oscilloscope (DSO). The primary advantage of a DSO is that it can store waveform signals being observed for later analysis and parametric measurements such as rise and fall times, frequency, and period. It can also be used to compare stored waveform images from another unit with the signals present in the unit under test.

Some DSOs can interface with a PC so a service technician can transfer recorded information from a known good machine onto floppy disk magnetic storage media. This capability can also be used to store failure symptoms to develop a magnetic file of past PC failures. Then if a technician is later working on a failed unit, he or she can access a record of past failure symptoms seeking a match for the problem being analyzed. By storing the waveforms from a known good machine and past failures, the DSO/PC configuration can be used to help and stimulate the troubleshooting process.

Most DSOs have more than one available channel. Many standard scopes measure only events that are displaced from the main sweep triggering event. This excludes the trigger event itself from measurement. A scope that includes the main-sweep triggering event in the possible delay range with the run-immediately-after-delay mode may not include the main-sweep triggering event in the trigger-after-delay mode. Most digitizing scopes lack the timing resolution and bandwidth required for analyzing really fast signals. Fortunately, the signals found in the IBM PC circuitry can be easily handled by a DSO.

Of particular interest is the capability of a scope to provide multiple channel analysis at the same instant in time. If you're monitoring an intermittent event and have a suspicion that the problem is related to or caused by another condition, you can use a multichannel DSO to tackle the malfunction. Putting one set of probes on the test point of the unit under test and another set of probes on the other suspected signal, you can observe both waveforms at the moment the intermittent occurs.

A limitation of the multichannel analysis can be seen when small mismatches in probe- and channel-delay times exceed the allowable error of a measurement. Better quality scopes provide a front panel adjustment to correct for this mismatch. Typically this is called "Channel 2 Delay Matching." Another problem that has occurred using multichannel scopes is that the vertical bandwidth and the transient response errors can obscure the true 50 percent (or other) reference points on the signal displayed.

However, the nice thing about dual trace, quad trace, and even eight trace is the ability to look at different signal paths or different signals simultaneously. For example, you could look at the input and output of a gate and actually see and be able to measure the delay time for the signal passing from input to output of the chip. Another useful technique is simultaneously displaying all or parts of the data bus, or part of the address bus to see what the logic level (high = +5 V, low = 0 V) is and what binary number it represents.

One way to use a dual trace scope to locate the point where a signal ceases erratically is to place the probes at the input and output of a suspected stage. You could also start with the probes several stages apart. Monitor each end of the circuit and start moving the scope probes toward one another until one probe reaches a point where the signal goes haywire. At this point, one probe will be touching the last point of the signal path that has a signal, while the other probe is on a point where some signal amplitude loss occurs at the next point of the signal path. Only one or two components separate the two test points. The problem will be found in the circuitry between the two scope probes, greatly reducing the number of suspected components.

Selecting a scope for the service center is much like buying an automobile. You must first know what you intend to do with the machine. When selecting an oscilloscope, determine what type of signals will normally be measured. Will the signals be repetitive or transient. Will the signals have a high or low repetition rate. What is the maximum frequency of the signals to be measured?

The response time of the vertical circuitry in the scope must be fast and flat enough to faithfully represent the signals being measured. The scope's rise time must be considerably faster than the transition times of the signals being monitored. The step response of the probes and the probe-ground connections should be clean. Coaxial connections may be required to obtain good time measurements.

Once the types of signals to be captured have been determined, you should decide the level of signal detail that must be observed on the screen. This is reflected in the accuracy, amplitude, DC offset, probe effects, range, sensitivity, vertical resolution, and whether single-ended or differential measurements will be made. DC offset is a way to obtain higher vertical resolution without resorting to AC coupling of the trigger signal.

If the signals will be repetitive, real time or equivalent time sampling can be used. A scope's real time scale represents the actual time at which succeeding samples are taken. An equivalent time scale represents the distance between two samples. With this sampling method, the sampling rate must be at least twice that of the highest frequency being measured (the so-called Nyquist rate). Repetitive signals above 100 MHz are better captured using equivalent time sampling.

If the most complex equipment that you'll be troubleshooting is the IBM PC, you can probably get along fine with a dual trace, 25-30 MHz scope. Investment in an oscilloscope is quite cost effective for a service center. A nice scope, recently advertised under $1400, boasts 100 MHz capability, eight trace, two time bases, alternate triggering, three-channel inputs, 0.5 millivolt sensitivity, 0.3 microsecond per division to 0.5 second per division time scale, and five millivolt per division to 5 volts per division voltage scale capability. This is an economical way to guarantee capability to attack the most difficult microcomputer failures.

After you fully understand how the scope works (you've read the owners manual and have followed by tutorial in the reference material) you're ready to make the plunge. When using an oscilloscope to troubleshoot a board for a bad component, the first step to take is to check the scope itself. Make sure the scope and the probes have been recently calibrated. If they aren't, you can't be sure whether the signal you are measuring is accurately represented. Consider powering up the scope and leaving it on for about 30 minutes to let its electronics reach ambient operating temperature. In this way, it won't give you problems that sometimes occur when you try to "read" a circuit in the midst of the warming up process.

Do preliminary checks first (see Chapter 4). Then try to get the problem to loop. Usually, you need the problem to occur during some program loop operation before the failure can be seen on a scope. In some cases the computer is always looping on the problem (for example, video, keyboard, memory refresh, and so forth). In other cases a program can be written that will loop on the problem. In those cases where you can't get the problem to loop, the use of a logic analyzer may be appropriate.

Start checking the control signals to see what is being activated or not activated. Attempt to determine what the PC thinks it's doing.

You need to collect as many clues as possible to help you decide what path through the circuitry you should take. Keep an open mind. Don't look for a particular type failure such as a short or open connection in a component or on the board. Assume that the problem could be any of many things–from a simple short to a capacitance noise problem on the main data bus.

Follow every signal in the circuitry affecting or being affected by the problem. Look for any clue that could provide more insight into the problem and lead you to the failed part, or another clue. Check voltages, data, and control signals. Compare related signals. Also compare signals that are remotely related (for example, data bus signals). If you have another PC system, try to loop on the same thing in that system and compare the measurements from each system.

Many technicians who've worked in this field for a time sometimes feel overwhelmed by the complexity of faults, such as capacitor and resistor failures. Don't become overwhelmed by failures that rarely occur. Capacitors and resistors seldom fail, but, when they do, the failure is usually quite visible. A big chunk blown out of a cap, or a discolored resistor whose value coding you can no longer read are visible indications of a short in part of the circuitry –something shorted with the power on. One cause can be pulling out or plugging in an adapter card while the board is energized. This can also happen if you short leads or traces on the circuit board with a scope probe. Use the utmost care while troubleshooting an energized circuit.

There isn't one scope setting that is appropriate every time you use the scope to troubleshoot PC failures. Many different types of failures can be scoped, and not every one will require the same scope settings.

Logic Analyzers

The logic analyzer is a multichannel oscilloscope with a memory. It captures and stores several digital signals, letting you view the signals simultaneously. If each signal is a bit on the data bus, you can see the entire data bus at one time. This means you can analyze the logic level for each bit on the bus for any instant in time. The bus signals are frozen for your display and analysis. The ability to freeze a single event or data pattern so you can determine the information present on a digital bus during a specific sequence in time is a distinct advantage for troubleshooting.

Logic analyzers, like oscilloscopes, cost between $500 and $20,000, and they come in a range of frequency bandwidths between 2 MHz and 200 MHz.

These analyzers can display many signals (channels of input) simultaneously. Arium Corporation in Anaheim, California sells an analyzer that handles 32 channels of input data at frequencies up to 100 MHz. Nicolet offers a 48 channel, 200 MHz analyzer with built-in microcomputer and dual, double density floppy disk drives. Each channel has associated with it a probe clip for connecting to some test point in the circuitry. Fortunately, the clip probes are tiny and easy to install.

A sampling of the capabilities available in logic analyzers reveals one configuration that provides 104 channels at 25 MHz, another with 32 channels at 100 MHz, and yet another with 16 channels at 330 MHz. Another configuration has 8 channels of input and can operate at 600 MHz! (Recall that your IBM PC clock is 4.77 MHz.)

Where would logic analyzers be useful? One place is in debugging software. You can read the data in machine code and trace its flow through the circuit. You could analyze the input and output of each memory bit on the data bus simultaneously for locating a bad RAM chip. Or you could uncover intermittent glitches, those phantom spikes that can raise havoc with your system. There are many more uses for logic analyzers, including analysis of disk I/O operation.

There are advantages and disadvantages associated with using a logic analyzer. One advantage is that you needn't loop on the problem. The analyzer will develop a historical visual record of the signals occurring at the points being measured. Logic analyzers are also great for intermittent failures or heat problems. However, this equipment is very expensive, doesn't measure voltage levels, is time consuming, and is difficult to connect to the circuit under test.

When troubleshooting with a logic analyzer, the first step is to determine what to analyze. You must look at the symptoms and decide which signals to check. If you've already checked the power and timing, and are ready to apply a logic analyzer to the problem, begin by connecting the analyzer probe clips to the control signal pins in the suspected area of circuitry. Connect the remaining probe clips to signals that are affected by the control signals being monitored.

Often technicians will get overwhelmed investigating problems on the large address and data buses. If you take the time to learn how to read these buses, you'll find them valuable when analyzing system board problems. With a logic analyzer you can visually observe conditions up to and during the time of failure. This can be done by connecting the analyzer to the data bus and the address bus at a point where they are active into the RAM memory. In this configuration, you can "read" exactly what's going on in the system's "mind." To do this, you'll need a logic analyzer with at least 32 channels. Then you can simultaneously monitor both the 8-bit data bus and the 16-bit address bus, and still have enough probes left to trigger on other signals.

You must understand the operation of the analyzer so you can set proper triggering and be able to recognize the screen display. Read the logic analyzer's operating manual to become comfortable with the tester. Logic analyzers can be a blessing in disguise, and they can help isolate almost any problem. But when hooking up 32 channels of tiny probe clips, there is ample opportunity for mistakes.

To save time and make troubleshooting easier, you can build a kludge that lets you connect all the necessary leads from the analyzer's pod to several regular chip clips at one time. By plugging the clips to the appropriate IC (labelled on each chip), with the correct pin 1 alignment, you can have the address and data lines ready for analyzing in less than 2 minutes. Individual clips can take up to 15 minutes to connect without the kludge. Several magazine articles have described how to construct such a kludge.

The logic analyzer has been called the oscilloscope of the digital domain. It can be a valuable tool for the software or hardware designer, but the investment can be large. Fortunately, most IBM PC failures can be found and corrected without the need for a logic analyzer.

Signature Analyzer

Logic probes can be effective in detecting logic levels and pulses at single points. Oscilloscopes can extend the number of points to be monitored even though the data pulses all tend to look alike.

And logic analyzers extend the number of test points even further to include buses the size of the data and address buses. However, as the sophistication and capability of the measurement device increases, so does the expertise required to operate the test tool. Logic analyzers, in particular, can be very capable but they can also be difficult to understand and operate. The signature analyzer was developed to allow easy detection of hardware failures.

Signature analysis is a comparison method of troubleshooting. It works by running a diagnostic program in the system being tested, and evaluating a coded signal at specific test points in the circuitry. If the coded signal matches the code observed when the system was running properly, the malfunction is not in that part of the circuitry. When a test point signature fails to match the baseline correct code, this suggests that you have located the faulty area. Then you can probe backwards or forwards from this point to isolate and locate the component that has failed.

The key to the success of this test technique is in the signature code. The first codes were developed by Hewlett-Packard and with slight modification are still being used today. A test code is a 16- or 24-bit repeatable value that represents a stream of data passing a test point during an interval of time. This known stream of data, when sampled at different places on a good circuit board by a signature analyzer, produces a unique 16- or 24-bit code at each test point.

These codes can be documented or stored in a programmable read only memory (PROM) and recalled later for comparison during troubleshooting. The PROM then becomes a custom memory module containing every signature sampled from a properly working system that was being stimulated or pulsed with a known data stream.

Signature analysis has not been a popular troubleshooting tool because it takes lots of time to identify the test points or nodes, probe the nodes, produce a signature, and then document the code. Once this task is completed, however, the task of locating a failure becomes a breeze. And the introduction of PROM modules has made the setup task much easier.

More improvements in this analysis technique can be expected soon. One analyzer on the market uses a mode called "backtrace" to prompt the troubleshooter through a series of test points, guiding the tester to trace bad signatures back to the failed part.

The investment for a signature analyzer is between \$400 and \$10,000. Signature analysis uses a simple, nontechnical approach to troubleshooting, so even untrained people can use the equipment and the technique.

Now that you understand what kinds of tools are available, let's look at how you can use these tools to find failed components.

USING TOOLS TO FIND FAILED COMPONENTS

Most components on the IBM PC motherboard are TTL (transistor-transistor logic) chips. If you know the logic gates in a chip to be tested (NAND, NOR, OR, AND, and so forth) you can test for opens or shorts by applying a known logic level to the inputs while monitoring the output. For example, if you were to place a slowly pulsing 0 volt to +5 volt signal on the input to the AND gate in Fig. 3-18 with both inputs shorted, you should see the output voltage level change (pulse) along with the input. Whenever the input is a logic high, the output becomes a logic high (between +5 volts and +2.4 volts).

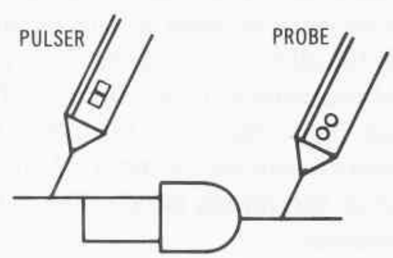


Fig. 3-18. Pulsing the input should cause a change in the output as indicated by the logic probe.

The tool you use on the input is a logic pulser. The monitor tool on the output is a logic probe. The pulser places a cyclical logic level on the input to the device and the probe measures the presence or absence of a logic signal on the output of the chip.

If the input to the AND gate becomes shorted to ground, the pulser cannot cause the gate to react to its signal and the output remains at a logic 0, or low (about 0 volt). Even if just one of the inputs shorts to ground, the output cannot change and remains low.

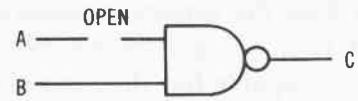
A short to the gate supply voltage (+5 volts) will have the effect of qualifying or enabling one input to the gate all the time. This means that each time the other input receives a logic high, the input set is correct and causes the output to change to a logic high even though only one input signal was actually correct. This produces incorrect circuit operation and strange results. This kind of problem shows up in memory circuits. Only one of the inputs to a particular gate is shorted or opened. Whenever this gate is used, the resulting output may be correct–a difficult problem to trace down.

Shorting an input pin to +5 volts can have potentially disastrous results. When the previous gate tries to deliver a logic 0, or low, a huge current is produced which usually causes catastrophic failure of the driving chip. The same result occurs when the input pin is shorted to ground and the previous gate tries to deliver a logic 1, or high. The +5-volt logic high is shorted directly to ground, producing an unusually high current with equally disastrous results.

Open connections prevent logic levels from being transferred and prevent the affected gate from being able to respond. If one input of a two-input NAND gate is open at the input as shown in Fig. 3-19, all but one of the four possible input combinations will be correct. This means that with this type of failure, the system could operate correctly most of the time with only half of the inputs good. The failure would be intermittent.

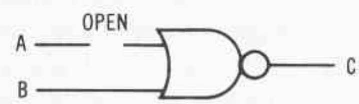
As shown in Fig. 3-20, if the device being tested is a NOR logic gate, the output would be a logic 1 only when both inputs are at logic 0. Should one of the inputs become open, it would float to logic 1 and cause none of the input conditions to produce a logic 1 output. Thus, the output would be low all the time–just as though the output were shorted to ground.

If the chip has an open pin at its output, it cannot deliver any logic 1 or 0 to the next gate.



INPUT A	INPUT B	OUTPUT C
L	L	GOOD
L	H	BAD
H	L	GOOD
H	H	GOOD

Fig. 3-19. An open at the input to a NAND gate is only a problem in one of four logic state cases.



INPUT A	INPUT B	OUTPUT C
L	L	L
L	H	L
H	L	L
H	H	L

Fig. 3-20. An open at the input to a NOR gate will prevent the output from ever changing state or going high.

You can measure a voltage at the input to the next gate since it is providing the potential, a logic 1 or high level (something around +4 volts). The key here is that any time an input to a TTL gate opens (a condition we call "floating"), the gate will act as though a logic 1 were constantly applied to that input. The voltage on this floating input will drift between the high supply voltage of +5 volts and a level (about 1.5 volts) somewhere between a valid high and a valid low. (A valid high is usually above +2.4 volts; a valid low is below +0.4 volt.)

A voltmeter reading of about +1.7 volts at the output pin of a gate on a chip is a clue that the output is floating open and the voltage is actually being provided by the next chip or following gate.

All these kinds of failures can be located using a logic pulser and a logic probe with backup from a VTVM for voltage measurement.

Since the PC system board is flexible at certain points, replacing chips by depressing the board without supporting it from beneath could cause a break to occur opening a trace on the circuit board. A hairline crack such as this is often difficult to find, but looking at the board with a magnifying glass and a strong light (or a magnifying lamp) can sometimes reveal a suspected failure. A resistance test can be conducted with a VOM or VTVM by placing a probe at either side of the suspected bad trace as shown in Fig. 3-21 and observing whether a zero ohm reading is measured. Another way to ascertain if an open trace is present is to compare the logic states at either end of the trace.

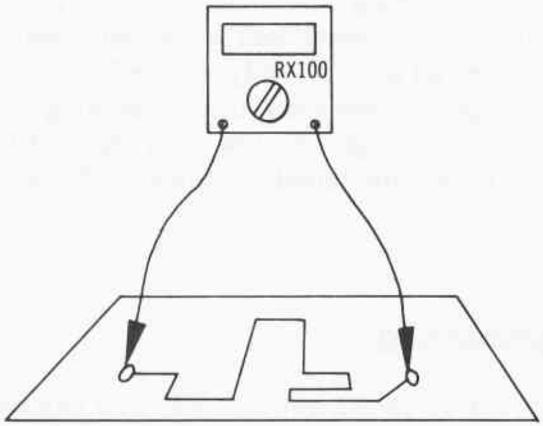


Fig. 3-21. A trace can be tested for an open using an ohmmeter to test the logic states at either end of the trace.

An important fact to keep in mind when testing for individual shorted or open gates in the IBM PC system board circuitry is that more than one gate may use the same input or output lines to or from another gate. This is called "fan-in" or "fan-out." When studying the gate circuitry, remember, the failure could be located at the other end of the board. One long trace from it to the chip you are looking at may be shorted or open at that end. Use of the schematics in Chapter 2 and the SAMS COMPUTERFACT for IBM PC will be of value here.

OTHER TROUBLE-SHOOTING TECHNIQUES

There are some interesting tricks you can use to aid in finding chip failures.

Use Your Senses

Look, smell, and feel. Sometimes failed components become discolored or develop bubbles or charred spots. Blown devices can produce some distinctive smells–the smell of a ruptured electrolytic capacitor, for example. Finally, shorted chips can get very hot. By using a "calibrated finger," you can pick out the hot spots on your board.

Heat It, Cool It

Heating and cooling is a fast technique for locating the cause of intermittent failures. Frequently, as an aging device warms up under normal operation, it becomes marginal and then intermittently quits working. If you heat the energized area where a suspected bad chip is located until the intermittent failures begin, and then methodically cool each device with a short blast of canned coolant spray, you can quickly cause a marginally defective chip to function again. By alternately heating, cooling, heating, and cooling, you can pinpoint the trouble in short order.

You can heat the area with a hair dryer or a focused warm air blower designed for electronic testing. Be careful using this technique since the thermal stress you place on the chips being tested can shorten the life of good components. A 1- or 2-second spray of freeze coolant is all you should ever need to get a heat-sensitive component working again.

Most coolant sprays come with a focus applicator tube–use this to pinpoint the spray. Avoid spraying electrolytic capacitors, because the spray soaks into the cap, destroying the electrolyte in some aluminum capacitors. Also be careful not to spray your own skin. You could get a severe frost burn.

The problem with hot air and freeze spray techniques is that the heating and cooling can weaken good chips. A logic analyzer can replace temperature stress techniques to find "weak" components by triggering on the failure.

Some technicians use both test methods together. In this technique, the analyzer is connected to the suspected circuit and a suspected IC is heated (if the system is temporarily working), to cause a failure, or cooled (if the system has already failed) until the circuitry starts working again.

Each part is not heated or cooled until the failure occurs because it may not be the bad component and you may force a new failure. Only the part that is suspected bad should be tested this way.

Assume that you are searching for a heat sensitive component. Your visual and preliminary checks don't turn up a thing. You analyze the symptoms looking for clues that might lead to the area of failure. The program runs and then the system locks up. So you open the equipment. You suspect a bad RAM IC on the system board. After all, memory chips fail often, don't they? So you spray coolant on each of the RAMs starting with U69. While spraying U72, the system suddenly starts working again. Thinking you've found the bad part, you replace U72. Buttoning the system back up, you rerun the program that was failing earlier. Ten minutes later, it locks up. Notice that, if you had buttoned the system up and only tested for one or two minutes, you would have assumed the computer repair complete and placed the system back in service. Without a 30-minute operational test to confirm problem correction, you may find yourself embarrassed and losing credibility fast when you're called on later to "fix that darn thing" again.

The retest failed. If you assume that this is one of those rare double chip failures, you reopen the unit and start spraying coolant on the RAMs again. This time, just as you are about to spray RAM IC U89, the system starts up again. Being a "thorough" technician, you let the system lock up again, and then you spray U88 again (this was the last chip you sprayed before moving to U89). Bingo! The system starts to run. Smiling with confidence that you've found the culprit component, you replace U88 and rerun the program. To your dismay, the machine locks up again! Irritated and frustrated, you start wildly spraying again. After an hour and a half (and three cans of coolant), you find the problem is the data buffer U12. The data buffer not a RAM! If you had used a logic analyzer to monitor the address and data buses while operating the PC to failure and cooling the system back into operation, you would have seen the failure and could quickly follow it to the bad buffer chip. Many technicians have used coolant spray hoping to identify heat-sensitive components only to replace a part that seemed to be helped by the spray and discover the problem unchanged.

The important thing to remember here is that when troubleshooting any failed part or failing circuitry, always keep your mind and eyes open for anything that could represent a clue to the problem. Sometimes a clue will lead you back over an area already covered. Don't be discouraged. Remember, troubleshooting can be detailed and tough. Follow each clue until you finally isolate the failed component. Never give up.

Piggybacking

Figure 3-22 shows another way you can chase down intermittents caused by a break (open) in a chip bond (wire) inside the chip housing that allows good contact only when the chip is cool.

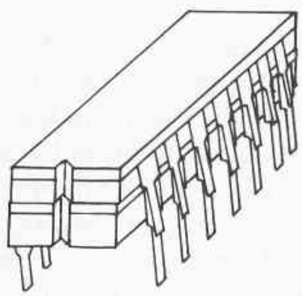


Fig. 3-22. Open-type failures can be found by piggybacking a good IC on a suspect IC.

Place a good chip over the top of a suspected chip and energize the circuitry. You may need to squeeze the pins of the good chip in slightly so they make good contact with the pins of the suspected device.

If the intermittent failure is caused by an open connection, the new chip will react to input data and cause its output to act accordingly. Use your stock of spare parts as your piggyback source. A major caveat here is that if the failure was caused by a short on one of an IC's pins, piggybacking a good chip over the shorted chip will blow out the good chip. This technique must be used with caution.

The Easter Egg Approach

Often we can quickly locate a fault to a couple of chips but need further testing to determine which chip is the culprit.

When time is of essence, take an "Easter Egg" approach. Just as a youngster used to pick up and examine Easter eggs one at a time to see if its name was marked on the egg, you can try replacing the chips one at a time to determine if the chip replaced was causing the problem. You have a 50-50 chance of selecting the right chip the first time. If it didn't work, replace the other chip.

If the chips involved are inexpensive "jelly beans" (7400 series TTL), why not replace them both? For 30 cents more, go ahead and splurge. If the problem's gone, but you're still curious, you can always go back later and test each chip individually.

MICROVOLT MEASURING A PIECE OF WIRE

If you have a meter with microvolt sensitivity and have isolated a "stuck low" problem to two chips, you can try the technique shown in Fig. 3-23. Measure the voltage drop between input pin 1 of gate B and output pin 3 of gate A.

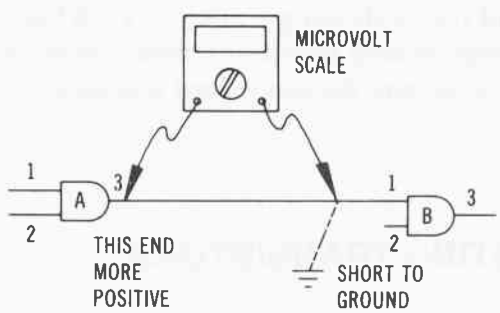


Fig. 3-23. A short circuit that is sinking current can be found using a meter with microvolt sensitivity.

This means measuring the opposite ends of the same trace or piece of wire! You're interested in determining which end of that trace is the more negative. The end nearest a bad chip will be more negative because the defective chip will short the trace voltage to ground causing a smaller microvolt reading at this point.

TESTING CAPACITORS

How do you check out a capacitor that you believe has failed? If the device has shorted, resulting in severe leakage of current, you can spot this easily by placing an ohmmeter across the capacitor and reading the resistance. At first you'll notice a low reading, because the capacitor acts as a short until it charges; but then, if the capacitor is working properly, it will charge, and the resistance will rise to a nominally high value. If the device is shorted, the initial low resistance reading continues and the capacitor won't charge.

Should the component be open, you'll not see the instantaneous short at T0, the moment charge starts to build. An open circuit has infinite resistance. An in-circuit capacitance tester is helpful here.

Total failure as a short or open is easy to find. But how about the device which its leakage depends on temperature or its dielectric has weakened, changing the capacitance value? To test this capacitor requires a different level of analysis.

CAPACITANCE MEASURING

If you have an ohmmeter which has the number 10 in the middle of the scale, you can easily use it to approximate the capacitance of a device. Using the time constant formula T=RC, where the time in seconds for a capacitor to charge to 63.2% of supply voltage is equal to the resistance in ohms times the capacitance in farads. Using a 22 microfarad (or 0.000022 farad) capacitor and a 1 megohm (1,000,000 ohm) resistor, the charge time for one time constant is 0.000022 x 1,000,000 = 22 seconds.

Transposing the formula to read:

$$C = \frac{T}{R}$$

where

C is capacitance in farads,
T is time in seconds,
R is resistance in ohms.

you can determine the value of capacitance by knowing the resistance and counting the seconds required for the charge to cause the ohmmeter needle to reach 63.2 percent of full scale (infinite resistance). This point is at about 17 on the meter's scale.

To do this, disconnect one end of a capacitor from the circuit, turn on your meter, and let it warm up for a minute. Zero adjust the ohms scale reading. Then estimate the ohms scale multiplier needed to let the capacitor charge in some acceptable time period. For microfarad capacitors use the x 100K scale because this will let the capacitor charge in less than a minute. The 17 on the scale represents 1.7 M ohms on the x 100K scale.

Short a low-ohm-value resistor across the two capacitor leads for several seconds to thoroughly drain off any charge. Then connect the meter's ground lead to the negative side of the capacitor (either side if the capacitor is not an electrolytic), and touch the positive meter probe to the other side of the capacitor. Using a stop watch to count seconds and tenths of a second, watch the face of the ohmmeter as the capacitor charges, moving the resistance needle up. When the needle gets to 17 on the scale, stop the clock and read the time. This will give you the value of capacitance in microfarads.

This technique will give you a close enough approximation of the capacitance value to determine if the device is good or should be replaced.

REPLACING CAPACITORS

Always try to use the same type and value capacitor as the one being replaced. Keep the leads as short as possible and solder the capacitor into the solder connector holes with the proper iron. The solder process should not exceed 1.5 seconds per lead, heat damage to the component may result.

A good technique to use is to tin the capacitor leads just before poking them through the circuit board holes. This speeds the solder bond process.

TESTING DIODES

If you have a digital multimeter (DMM) with a diode test capability, you can quickly determine whether a suspected diode is bad or good. Placing the meter on the ohmmeter setting and the probes across the diode causes the meter to apply a low current through the diode if the diode is forward biased. The voltage drop across a diode is normally 0.2 to 0.3 volt for germanium diodes and 0.6 to 0.7 volt for silicon diodes. Reversing the leads should result in no current flow, so a higher voltage reading should be observed. A low voltage reading when biasing the diode in either direction suggests that the device is leaking or shorted. A high voltage reading in both directions suggests the diode bond has opened. In either case, replace the diode immediately.

In-circuit tests of diodes can also be done using the ohmmeter to check the resistance across a diode in both directions. With one polarity of the meter probes, you should get a reading different from that obtained when the probes are reversed–not just a few ohms different but several hundred ohms different. For example, in the forward-biased direction, you could read 50 to 80 ohms; in the reverse biased direction, 300K ohms. This difference in readings is called DE for "diode effect" and is useful for evaluating transistors. When diode readings in both directions show low resistance, you can be sure the leaky short is present.

TESTING TRANSISTORS

It's no fun to desolder a transistor to test it for failure and find that it tests good. You then have to solder it or a new device back into the circuit board and go to the next suspected transistor and desolder it, and so forth.

Fortunately, there is a way to determine the quality of silicon transistors without removing them from the circuit. In 90 percent of the tests, this procedure will accurately determine whether a device is bad.

Figure 3-24 shows that transistors operate the same way as a configuration of diodes. PNP and NPN transistors have opposite-facing diodes. The transistor functions by biasing certain pins and applying a signal to one of the leads (usually base) while taking an output off the collector or emitter.

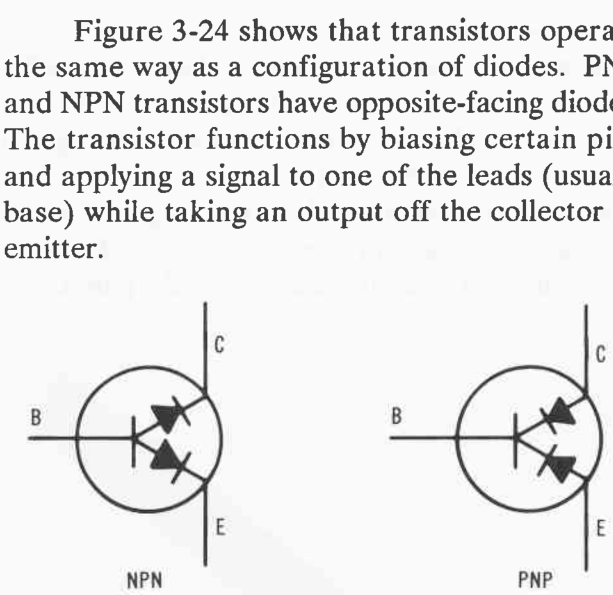


Fig. 3-24. A transistor acts like a pair of diodes.

These tests apply to both PNP and NPN transistors. If an ohmmeter is placed between the collector and emitter as shown in Fig. 3-25, it effectively bridges a two-diode combination in which the diodes are opposing. You should get a high resistance reading with the leads applied both ways. (It's possible to wire the transistor in a circuit which makes the transistor collector-emitter junction act like a single diode. In this case you could get the diode effect. Both results are normal.)

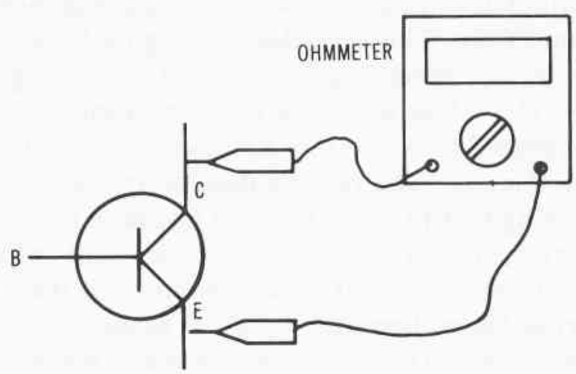


Fig. 3-25. A transistor can be tested using an ohmmeter placed across the collector-to-emitter junction.

Typical collector to emitter resistance readings for germanium transistors are as follows:

Forward biased = 80 ohms
Reverse biased = 8000 ohms (8K)

For silicon transistors you might read:

Forward biased = 22 megohms
Reverse biased = 190 megohms

The high/low ratio is evident and is about the same for both types.

Place the probes across the collector-to-base junction leads. Reverse the probes. You should observe a low reading in one case and a high reading with the test probe leads reversed (the diode effect).

Try the same technique on the base-to-emitter junction leads. Look for the diode effect (Fig. 3-26). If the diode effect is not present in all the previous steps, you can be certain the transistor is bad and needs replacing.

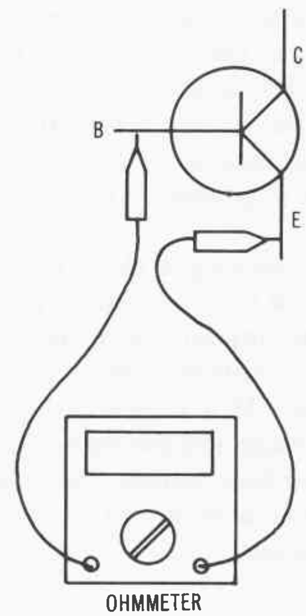


Fig. 3-26. Check the base-to-emitter junction for the diode effect.

Another way to evaluate a transistor is to measure the bias voltage from base to emitter on an energized circuit. Confirm the correct supply voltage first; power supply problems have been known to trick troubleshooters into thinking a certain component has failed.

The base-to-emitter forward bias for silicon transistors should be between 0.6 and 0.7 volt

DC. If the reading is below 0.5 volt, replace the diode–the diode junction is leaking too much current (Table 3-3). If the reading is almost 1 volt, the junction is probably open and again the device should be replaced.

Table 3-3. Base to Emitter Voltage Readings and Action to Take

Base to Emitter Voltage *(forward biased)*	Action
0.5	Replace
0.6-0.7 V	Good, keep
0.9	Replace

Although in some isolated cases some other failure could cause the low reading, the most common cause of low bias voltage is failure in the transistor itself.

If the previous tests are inconclusive, there is something else you can try. Measure the voltage across the collector-to-emitter junction. If the reading is the same as the source supply (+5 volts for Q1 on the color/graphics adapter) and you notice on the schematic that there's plenty of resistance in the collector/base circuit, the junction is probably open. Replace the device.

If your reading is close to 0 volt, take a small length of wire and short the base to the emitter, removing all the transistor bias. The collector to emitter meter reading should instantly rise. If it doesn't, the transistor is shorting internally and should be replaced. If the collector to emitter voltage does rise, it suggests a failure in the bias circuitry, perhaps a leaky coupling capacitor.

SOLDERING AND DESOLDERING TECHNIQUES

Removing Solder

Removing solder from the printed-circuit boards in the IBM PC must be done rapidly. Typically, remelted solder on a joint must be removed in less than 3 seconds. There are many ways to remove solder from printed-circuit boards. One way to remove residual solder is to use a "solder sucker"–a hand held vacuum pump with a spring-driven plunger to pull the hot, melted solder off a connector (Fig. 3-27). The process involves heating the old solder until it melts, placing the spring-propelled vacuum pump in the hot solder, then quickly removing the soldering iron while releasing the vacuum pump's spring, sucking the solder up into a storage chamber in the pump.

Fig. 3-27. The spring-driven plunger in the solder vacuum pump is used to pull hot solder off a connection.

Continuous vacuum solder extraction is another technique for desoldering components from PCBs. This type desoldering tool uses an electrically generated vacuum and tip heating to melt the solder around a joint and rapidly suck the melted solder up into a storage chamber built into the tool. The vacuum desoldering tip has an opening that fits over the lead or pin to produce maximum heat transfer to the joint while providing sufficient space around the lead to permit the molten solder and air to pass through the space. The tip is the primary means to transfer the heat to melt the solder joint. It is also the path through which molten solder will be sucked by vacuum action. A good tip will pull ambient air through the tip to cool the joint and prevent resweating. Resweating occurs when the lead reconnects to the side walls of a plated-through hole in a printed-circuit board.

The tip is also used to manipulate the lead during the desoldering process. Once melting action begins with the tip over the lead, it is used to gently wiggle the lead until the lead swings freely and easily, suggesting full solder melt in the joint hole. Wiggling the lead with the tip provides a mechanical sensation that sufficient solder melting has occurred.

Once the solder has melted, the vacuum is applied while continuing to wiggle the lead. This removes almost all molten solder. It also cools the lead, and the pad, and prevents resweating. Because the similar controlled pressure and heat used to bond a part into a board is used to remove the same part, care must be exercised to prevent delamination. Excessive pressure when applying a soldering or desoldering tool to a solder joint can delaminate circuit pads or runs.

A good rule of thumb is to apply tip temperature heat between 575 and 600° F for about 2 seconds following initial solder fillet melt on the lead side. Then apply vacuum (while wiggling the lead) for 1 or 2 seconds. Once the lead moves freely, full solder melt has occurred. Maintain the lead motion and apply the vacuum suction. With the vacuum still on, gently remove the tip. When the tip is away from the lead, turn the vacuum off.

Lead resweating to the side walls can be prevented by moving the tip during heat and vacuum application. Move the lead in a circular motion when desoldering round leads. Move the lead back and forth along the flat plane when desoldering flat leads.

When desoldering DIP packages, space the pin desoldering sequence to prevent working on adjacent leads and causing localized heat buildup. Do the corners first, then skip over every other pin on the first desoldering pass. On the second pass, desolder the alternate leads.

If sweat joints still occur after desoldering the leads on a component, let the joints cool down. Then resolder the joints that were not completely desoldered and conduct vacuum desoldering once again.

This technique works fine until you try to use it around CMOS chips. Some vacuum pumps produce static electricity, and by now you know what that can do to an MOS or CMOS chip.

A safer way to remove solder is to use solder braid. Solder braid is an inexpensive alternative to vacuum desoldering. In fact, it prevents human error by applying heat too long because the soldering iron never touches the solder. When using solder braid, touch the solder with the end of a strip of braided copper and then heat the braid just a short distance from the solder (Fig. 3-28). The copper braid heats quickly, transferring the heat to the solder, which melts and is drawn into the braid by capillary action. Then, remove the solder-filled braid and cut off the silver solder-soaked portion of the braid and throw it away. The copper-colored portion of the braid is ready for the next application.

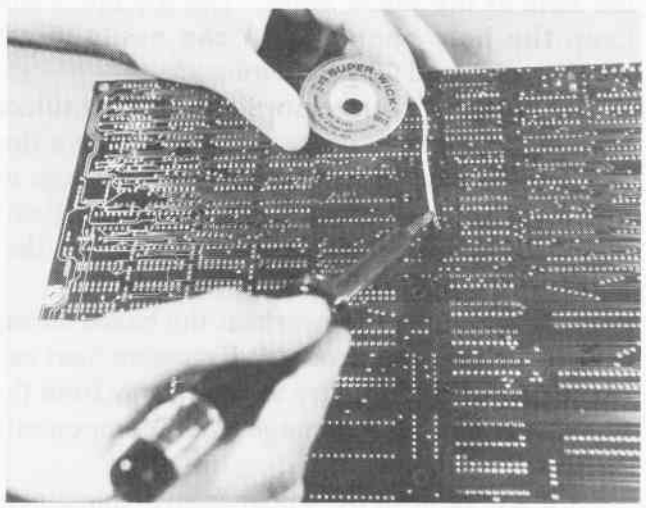


Fig. 3-28. Solder can also be removed from a connection by using a solder wick to draw the solder up into the braid.

Because the soldering iron never touches the solder itself, component traces and the circuit board are protected from thermal damage. This braid can also be used to remove solder bridges or icicles of solder that cause shorts on the board. Desoldering braid works equally well for removing excess solder from plated through holes in printed-circuit boards. The capillary action of the solder wick will pull molten solder easily out from the space between the lead and the hole.

Desoldering braid can be purchased in varying widths ranging from 0.035 to 0.220 inch. The proper size of braid will equal, or be slightly larger than the connection being desoldered.

According to Carolyn Watson in her article "Desoldering today's circuit components" (Electronic Servicing & Technology, October 1984), the smallest size (0.035 inch wide) is good for microcircuits; small pads required braid 0.06 inch wide. Medium pads need 0.075 inch braid, and large pads require 0.11 inch braid. Wicks that are too large may soak up too much heat from the iron restricting heat transfer to the soldered joint. Too narrow a wick will hamper sufficient solder removal and prompt several applications to complete a desoldering operation.

Should you be using a solder removing technique other than copper braid and discover that some solder remains in the circuit board hole, heat the solder and push a toothpick into the hole as the solder cools. The toothpick will keep the hole open so you can easily insert another wire lead for resoldering.

Another way to remove the residual solder blocking a hole is to drill out the hole with a tiny drill bit. Be sure to remove any debris, filings, or pieces of solder before energizing the circuit board. Use a magnifying glass to confirm that nothing unwanted remains on the board.

Be careful not to overheat the board during the solder-removal process. Excessive heat can cause part of the circuitry to come away from the board. It can also damage good components nearby.

If you remove the solder from a component and a lead is still stuck on some residual solder, take a pair of needle-nose pliers and pinch the lead as you gently wiggle it to break it loose from the solder bond. Or you can remelt the residual solder and gently wiggle the lead free.

The pins of some chips are bonded to the circuit board by a process called wave soldering. Wave soldering produces an exceptionally good bond without the added manufacturing expense of a socket. This process helps keep the fabrication costs down, but it makes it more difficult for you to replace the chip.

One effective way to remove wave-soldered chips is to cut the chip leads or pins on the component side and remove the bad chip. Then remove the pieces of pin sticking through the board using a soldering iron and solder braid or a vacuum pump.

Some special tools are available to help you in removing soldered components. Figure 3-29 shows a desoldering tip that fits over all the leads of a chip socket or dual in-line package (DIP) socket.

Figure 3-30 shows a spring-loaded DIP extractor tool. By attaching this device to the chip and then applying the DIP tip shown in Fig. 3-29 to the soldered connections on the opposite side of the board, you can easily remove a complete chip all at one time. Press the load button downward and engage the clips, causing the extractor to place an upward spring pressure on the chip. When the solder on the reverse side melts enough, the chip will pop up and off the board.

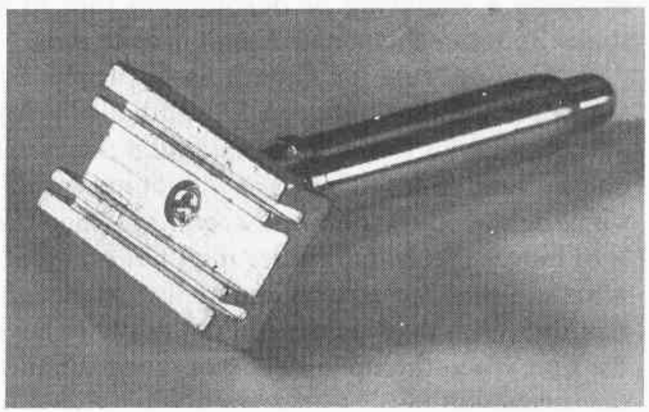


Fig. 3-29. A desoldering tip for removing chips that are soldered to the circuit board.

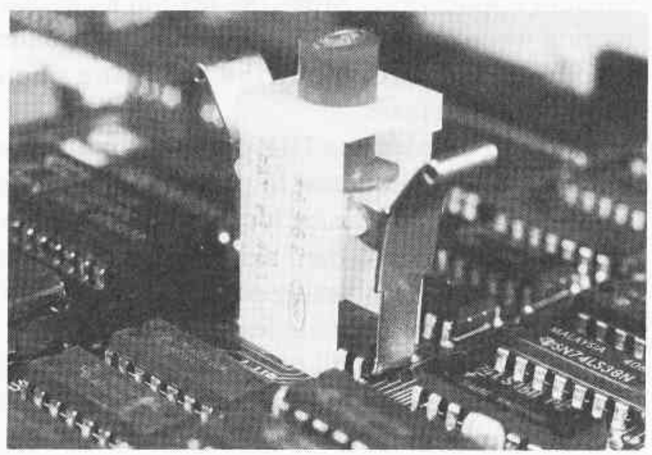


Fig. 3-30. A spring-loaded dual in-line extractor tool.

When you replace a chip that was soldered to the printed circuit board, always solder in a socket and then plug the replacement chip into this socket. This will make future replacements a lot easier. Be careful to maintain the correct pin 1 alignment.

Soldering Tips

No pun intended. Hand soldering is the most misunderstood and most often abused function in electronics repair. Not only do many people use poor soldering techniques, but they also use the wrong soldering irons.

Solder isn't simply an adhesive bandage making two metals stick together. It melts and combines with the metals to form a consistent electrical and mechanical connection. Time and temperature are critical in this process. The typical hand solder job should be accomplished in 1.5 seconds or less if the soldering iron and tip are properly selected and then properly maintained.

The nominal solder melting temperature is 361° F. Metal combination between the solder and the metals being joined occurs at temperatures between 500 and 600° F.

Most soldering jobs join the metals copper and tin, but both of these metals are easily oxidized. Poor or no solder connections are made if the surfaces to be connected are covered by contaminants such as oils, dirt, or even smog, so be sure to use solder with a good cleaning flux. The flux prepares the surfaces for best solder metalization. The flux melts first and flows over the metal surfaces removing oxidation and other contaminants. Then the metal heats so that the solder melts and flows producing a good, shallow bond.

The key to successful soldering is in the soldering iron tip. Most technicians selecting their first soldering iron jump right into a low wattage iron, but this is a mistake. Instead, pick an iron whose tip operating temperature is suited for the circuit board you're to repair. If the tip temperature is too low, the tip sticks to the surface being soldered. If it's too high, it damages the board surface. The ideal working temperature for soldering on your computer's circuit board is between 600 and 700° F.

The soldering iron tip is used to transfer the heat generated in the iron out to the soldering surface. The iron should heat the tip quickly, and the tip should be as large as possible yet slightly smaller than any soldering pad on the board.

Tips and the contacts or pins that you're soldering are made of the same copper material. Copper quickly conducts heat, but it dissolves in contact with tin. Solder is made of tin and lead. To keep the tin from destroying the copper tip, manufacturers plate a thin layer of iron over the tip. The hot iron (now you know where the term "iron" came from) still melts the solder, but now the tip lasts longer. The iron melts above 820° F, so if the heat produced by the iron stays below 700° F, the solder melts but not the iron plating.

The disadvantages of the iron plating are that it doesn't conduct heat as well as copper, and it oxidizes rapidly. To counteract this, you can melt a thin coat of solder over the tip. This is called "tinning." This solder layer helps the soldering iron heat quickly and also prevents oxidation.

The tip of an old soldering iron is usually black or dirty-brown with oxidation, so it doesn't conduct heat very well. These "burned-out" tips can be cleaned with fine emery cloth and then can be retinned and used.

Wiping the hot tip with a wet sponge just before returning the iron to its holder is a mistake. This removes the protective coating, exposing the tip surface to atmospheric oxidation. It's much better to add some fresh solder to the tip instead. Keep your iron well tinned.

Figure 3-31 shows the proper way to solder a socket or connector lead. Place the tip of the iron on one side of the lead and the solder on the other side.

As the solder pad heats, the tin/lead solder melts and flows evenly over the wire and the pad. Keep the solder shallow and even. When you think your soldering job is complete, carefully inspect your work. Sometimes, if you aren't careful, you can put too much solder on the joint such that there's not enough solder on the top or bottom of the connection. It's also possible to get internal voids or hollow places inside the solder joint. Large solder balls or mounds invite "cold solder joints" where contact is only partially made. Figure 3-32 shows some examples of inadequate soldering. These types of solder joints can be a source of intermittent failure.

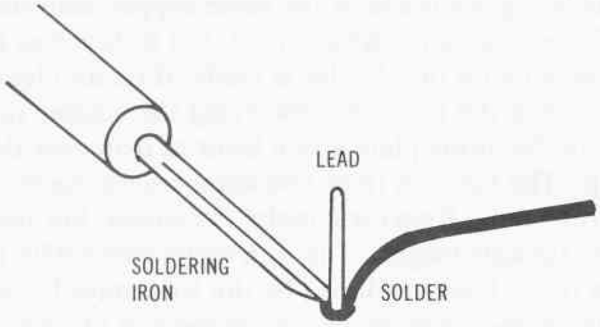


Fig. 3-31. Place the soldering iron on the opposite side of the lead from the solder.

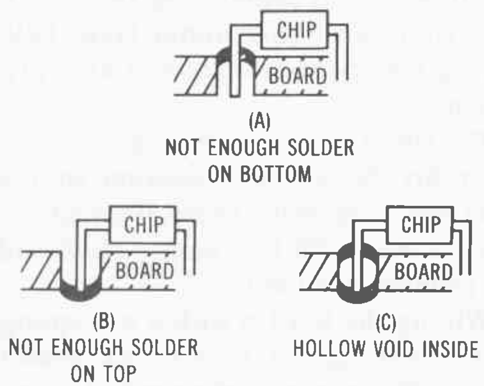


Fig. 3-32. Examples of inadequate soldering on a printed circuit board.

Good soldering takes patience, knowledge, and the right tools–a temperature-controlled soldering iron whose tip temperature is maintained in the 500 to 600° F range for best soldering effect.

Before You Solder It In

A useful thing to do before you solder in a replacement part is to test the device in the circuit. Simply insert the chip or other device into the solder holes and wedge each lead in its hole with a toothpick. Then energize the circuit and test. After proper function is assured, remove the toothpicks and solder the component into the board.

CIRCUIT BOARD REPAIR

Repairing damaged circuit boards is a lucrative business, and several companies have developed around this activity. For some board failures, you can repair your own circuitry and save time and money. As a technician, you may be required to do circuit board repair as part of your job.

Before soldering in new components, check over the board for any broken traces or pads lifting off the board. If a trace is open and is starting to lift away from the board surface, jumper across the broken spot from one component solder pad to another pad. Use solid No. 18 or No. 20 wire tinned at both ends before soldering.

If a pad or trace lifts free, replace it with an adhesive-backed pad or trace overlapping the damaged area. Scrape the coating off the pad or both ends of the trace so the new pad or trace can be soldered firmly to the existing pad or trace. Remove all excess solder and redrill any lead hole that has become covered or plugged with residual solder.

RECOMMENDED TROUBLE-SHOOTING AND REPAIR EQUIPMENT

As a serious hobbyist or technician you'll be tackling some tough problems. You can minimize your investment costs and yet optimize your chance of success by carefully selecting your equipment and tools.

First, make sure you have a set of good screwdrivers–both flathead and Phillips head. Maintain a wide selection of sizes–from the tiny "tweakers" to an 8-inch flat head. You might also find a set of jeweler's screwdrivers helpful.

Maintain several sizes of long-nose or needle-nose pliers, and several sizes of diagonal cutters or "dykes" for cutting wire and pins. A good low wattage soldering iron whose tip temperature is automatically controlled is a must if you intend to replace nonsocketed components. A simple $3^{1}/_{5}$ digit DVM or DMM is useful for test measurements. Another handy tool is the logic probe.

You'll need a 15 to 25 MHz oscilloscope with dual trace and a time-base range of 200 nanoseconds to 0.5 second. Select a scope with a vertical sensitivity of 10 millivolts per division or better.

You can get by quite nicely using the probe, pulser, tracer, and DMM as your primary equipment. An oscilloscope will make certain failures easier and faster to find.

SPARE PARTS

Finding that a trouble exists is only part of the problem. You must locate the specific chip (if the software doesn't) and then make the repair. This, too, can be somewhat challenging. Fortunately, most of the chips on your IBM PC system board are standard 74xx and are readily available. Just about any electronic parts store will have a supply of 7400 series chips. Most of the 8xxx series chips are also easy to obtain. You can make a list of the spare parts you'd like to have on hand from the listing in Appendix A.

Because of the cost involved you will probably want to maintain a minimal stock of repair parts; yet you want to be able to fix a machine quickly when it breaks down.

The optimal backup would include one each of every type of chip on the IBM PC's system board and adapter cards. This represents an investment of $100 to $200 in 150 or more chips. Now, IBM PC custom chips–the ROMs–are available from authorized IBM service centers. The total number of chips in the PC is higher than the number of chips you need as spares because many of the same type chips are used in different places on the system board. In addition, you only need a few of the RAM chips as spares. Your largest expense in chips will be for the ROMs (unless you are using an 8087 coprocessor chip).

Several companies are marketing spare parts packages with schematics, diagnostic tests, and one each of the chips for the IBM PC.

The custom IBM PC chips–the five ROMs –are proprietary to IBM, and I recommend you buy all these custom chips directly from IBM or from your local IBM PC repair center. Most chips for the IBM PC motherboard are generally easy to locate and are also inexpensive.

Should you be using an 8087 coprocessor in your PC, and need to replace the 8087 or the 8088, be aware that these are often sold as a matched pair. To see if your 8088 will work properly with your 8087, check the copyright date marking on the top of the chip. If the copyright date is 78, it *may* work with the 8087. If the date is '78 '81, it is certain to work with the coprocessor chip. Any other date stamp might not work unless it is stamped P8088 or D8088 followed with S4716. These revisions of earlier 8088 devices will also work.

Recent articles in *PC Magazine* suggest that the NEC V20 processor is a direct replacement for the 8088 CPU while providing a 5 to 10 percent processor speed improvement. The NEC V20 is also compatible with the Intel 8087 match coprocessor.

The trade magazines sometimes advertise inexpensive packages of IBM PC repair parts. Some of the PC chips can be replaced only by a board exchange. The exchange price may seem high, but you do get a new board, and the type of failure that requires this action does not occur very often. In fact, it occurs very infrequently.

In Appendix A you'll find a listing of each chip in your computer including its designation, name, and location.

SUMMARY

Customers expect you to be perfect. With the computer field changing rapidly, it's tough to keep on top of new developments. You need books like this technical manual to increase your knowledge, learn new techniques, and avoid the mistakes of poor troubleshooting–such as using a bare cotton swab with low-grade alcohol, "cleaning" a disk drive read head, wiping a soldering iron on a wet sponge just before putting it in its holder. These are common errors of poorly trained (or poorly motivated) technicians. After reading this book, you know the right tools to use and the right procedures to follow to troubleshoot and repair failed systems in minimum time.