V Technical Troubleshooting

Troubleshooting

Introduction

The most common problems occur in harness components such as the coin acceptor, player controls, interconnecting wiring, etc. The TV monitor and PCB computer cause their share of problems too, but not as much as the harness and its component parts. TV monitor troubleshooting will not be covered here because it is covered in that section of this manual.

As you already know, the PCB computer is a complex device with a number of different circuits. Some circuits remain basically the same among games, but overall there are a great many differences between them. PCB troubleshooting procedures, therefore, can be lengthy and will differ greatly among games. However, some basic Z-80 CPU information is involved in this section.

General Suggestions

The first step in any troubleshooting procedure is correctly identifying the malfunction's symptoms. This includes not only the circuits or features malfunctioning, but also those still operational. A carefully trained eye will pick up other clues as well. For instance, a game in which the computer functions fail completely just after money was collected may have a quarter shorting the PCB traces. Often, an experienced troubleshooter will be able to spot the cause of the problem even before opening the cabinet.

After all the clues are carefully considered, the possible malfunctioning areas can be narrowed down to one or two good suspects. Those areas can be examined by a process of elimination until the cause of the malfunction is discovered.

Harness Component Troubleshooting

Typical problems falling in this category are coin and credit problems, power problems and failure of individual features.

NO GAME CREDIT

For example, your prospective player inserts his quarter and is not awarded a game. The first item to check is if the quarter is returned. If the quarter is returned, the malfunction most certainly lies in the coin acceptor itself. First, use a set of test coins (both old and new) to ascertain that the player's coin is not undersize or underweight. If your test coins are also returned, coin acceptor servicing is indicated. Generally, the cause of this particular problem is a maladjusted magnet gate. Normally, this will mean slightly closing the magnet gate a little by turning the adjusting screw out a bit (see section on coin acceptor for more details).

If the quarter is not returned and there is no game credit, the cause of the malfunction may be in one of several areas. First try operating the coin return button; if the coin is returned, the problem is most likely in the magnet gate. Enlarge the gap according to the coin acceptor service procedures. If this does not cure the problem, remove the coin acceptor, clean it and perform the major adjustment procedure.

If the trapped coin is not returned when the wiper lever is actuated, you may have an acceptor jammed by a slug, gummed up with beer, a jammed coin chute, or mechanical failure of the acceptor mechanism. In this case, first check for the slug that will generally be trapped against the magnet. If so, simply remove the slug and test the acceptor. If the chute is blocked, remove the acceptor and remove the jammed coins. If there is actual failure of the acceptor, remove the unit and repair as indicated in the coin acceptor service procedures.

If the coin is making its way through the acceptor (that is, falling into the coin box), yet there is still no game credit, you either have a mechanical failure of the coin switch or electrical failure of the coin and credit circuits. The first place to begin is by checking the coin switch. Most of these switches are the make/break variety of micro switch, which is checked by testing for continuity between the NO, NC, and C terminals. When not actuated, the NC and C terminals should be continuous and the NO terminal open. When operated, the NO and C terminals should close and the NC should be open. If the coin switch checks out, examine the connections to the terminals to make sure there is good contact. If necessary, use the continuity tester and check from the terminal lug on the switch to the associated PCB trace. This will tell you if there is a continuous line all the way to the credit circuit.

If the coin switch wires do not check out, the problem is in the computer — most likely in the coin and credit circuitry.

If you do get game credit when a coin is deposited, but the game will not start when the start switch is pressed, you may have a problem in the start switch, the interconnecting wiring or in the computer. First check the switch. If the switch is OK, proceed to check the wiring. Again, make sure you go from the terminal lug on the switch to the PCB trace. This way, you will check the terminal contact as well as PCB edge connector contact. If the wiring is continuous, proceed to check the PCB credit circuit. If not, check each section of the wiring, until the discontinuity is located. If the wiring is OK, the problem must lie in the computer.

Transformer and Line Voltage Problems

Your machine must have the correct line voltage to operate properly. If the line voltage drops too low, a circuit in the computer will disable game credit. The point at which the computer will fail to work will vary some from game to game, but no game will work on line voltage that drops below 105 VAC.

Low line voltage may have many causes. Line voltage normally fluctuates a certain amount during the day as the total usage varies. Peak usage times occur mainly at dawn or dusk, so if your machine's malfunction seems to be related to the time of day, this may be a factor. A large load connected to the same line as the game (such as a large air conditioner or other device with an exceptionally large motor) may drop the line voltage significantly when starting up. This drop can result in an intermittent credit problem. In addition, poor connections in the location wiring, plug, or line cord may also cause a significant drop in power. Cold solder joints in the game's harness, especially in areas like the transformer connections, interlock switch, or fuse block, may also produce the same results, although probably on a more permanent basis.

Sometimes location owners (especially in bars) replace light switches with dimmer rheostats, and the game is sometimes on the same line. Obviously, the voltage available to the game is going to drop dramatically when the dimmer is turned.

In any case, the way to check for correct line voltage is with your VOM. Set the VOM to 250 VAC and stick the probes in the wall receptacle. If it's OK here, check the transformer primary connections. If you do not get 117 VAC, examine the solder joints on the transformer, fuse block, and interlock switch. If you do get 117 VAC, the problem must be either in the transformer, harness connections, or in the PCB power supply.

If you suspect the transformer, check its secondaries with the VOM set to 50 VAC and correlate the readings with the legend on the side of the transformer. The transformer must also be correctly grounded, so check the ground potential as well, especially if there is a hum bar rolling up or down the TV screen.

HARNESS PROBLEMS

Other harness problems include blowing fuses and malfunctioning controls. The repeating blown-fuse problem can sometimes be quite exasperating to solve, for short circuits have the tendency to occur in areas almost impossible to find. First, try inserting a new fuse, as old fuses age and blow without cause. If the new one also blows, you definitely have a short.

The best way to approach this problem is by turning the power off and disconnecting devices that may be causing the problem, such as the TV, transformer, and PCB. Disconnect the devices by pulling off their connectors, but do not allow them to touch. If necessary, insulate them with small pieces of electrical tape. Then, connect your VOM across the terminals of the fuse block (all electrical power shut off), and set it to one of the resistance scales. This will save blowing a fuse each time you want to check the circuit.

If the VOM reveals that disconnecting the devices removed the short, reconnect the devices one by one until the short returns. The last device connected is the one that is at fault. If the VOM reads a short even after the devices are disconnected, the fault must lie in the harness itself, and only patient exploration will reveal its location. First, carefully examine all the wiring, looking for terminals that may be touching, metal objects such as coins shorting connections or burned insulation. If necessary, use the VOM to check each suspected wire.

MALFUNCTIONING CONTROLS

One of the most common problems here is a bad potentiometer. Typically, a bad pot will cause the image to jump as it reaches a certain point. The only cure for this one is to install a new pot.

If a feature that is operated by a switch (for example, joysticks, foot pedals, control panel buttons) does not operate at all, check the switch with a VOM or continuity tester to verify its operation. If the switch does not check out, replace it. If the switch is OK, you should suspect the input to the switch from the PCB. In this case, get out the harness and logic schematics and check to see what kind of input it is. In many cases, the input will be +5 VDC. If so, use the VOM to check its presence. Normally, the switch is used to pull a +5 VDC line LOW to GND or to pull a LOW line HIGH. If the PCB output is missing, check the wire length from the PCB. If you find the signal at the PCB trace, the wire length or connection is at fault. If not, begin exploring the PCB using the logic schematics.

A Glossary of Microprocessor Terms

MICROPROCESSOR — one or several micro-circuits that perform the function of a computer's CPU. Sections of the circuit have arithmetic and comparative functions that perform computations and executive instructions.

CPU — central-processing unit. A computing system's "brain", whose arithmetic, control and logic elements direct functions and perform computations. The microprocessor section of a microcomputer is on one chip or several chips.

PROM — programmable read-only memory. User permanently sets binary on-off bits in each cell by selectively fusing or not fusing electrical links. Non-erasable. Used for low-volume applications.

EPROM — erasable, programmable, read-only memory. Can be erased by ultraviolet light bath, then reprogrammed. Frequently used during design and development to get programs debugged, then replaced by ROM for mass production.

ROM — read-only memory. The program, or binary on-off bit pattern, is set into ROM during manufacture, usually as part of the last metal layer put onto the chip. Nonerasable. Typical ROM's contain up to 16,000 bits of data to serve as the microprocessor's basic instructions.

RAM — random-access memory. Stores binary bits as electrical charges in transistor memory cells. Can be read or modified through the CPU. Stores input instructions and results. Erased when power is turned off.

LSI — large scale integration. Formation of hundreds or thousands of so-called gate circuits on semiconductor chips. Very large scale integration (VLS) involves microcircuits with the greatest component density.

MOS — metal-oxide semiconductor. A layered construction technique for integrated circuits that achieves high component densities. Variations in MOS chip structures create circuits with speed and low-power requirements, or other advantages (static will damage a MOS chip).

Introduction to the Z-80 CPU

The term "microcomputer" has been used to describe virtually every type of small computing device designed within the last few years. This term has been applied to everything from simple "microprogrammed" controllers constructed out of TTL MSI up to low end minicomputers with a portion of the CPU constructed out of TTL LSI "bit slices." However, the major impact of the LSI technology within the last few years has been with MOS LSI. With this technology, it is possible to fabricate complete and very powerful computer systems with only a few MOS LSI components.

The Zilog Z-80 family of components can be configured with any type of standard semiconductor memory to generate computer systems with an extremely wide range of capabilities. For example, as few as two LSI circuits and three standard TTL MSI packages can be combined to form a simple controller. With additional memory and I/O devices a computer can be constructed with capabilities that only a minicomputer could previously deliver.

New products using the MOS LSI microcomputer are being developed at an extraordinary rate. The Zilog Z-80 component set has been designed to fit into this market through the following factors:

1. The Z-80 is fully software compatible with the popular 8080A CPU.
2. Existing designs can be easily converted to include the Z-80.
3. The Z-80 component set is at present superior in both software and hardware capabilities to any other microcomputer system on the market today.
4. For increased throughput the Z80A operating at a 4 MHZ clock rate offers the user significant speed advantages.

Microcomputer systems are extremely simple to construct using Z-80 components. Any such system consists of three parts:

1. CPU (Central Processing Unit)
2. Memory
3. Interface Circuits to peripheral devices

The CPU is the heart of the system. Its function is to obtain instructions from the memory and perform the desired operations. The memory is used to contain instructions and in most cases data that is to be processed. For example, a typical instruction sequence may be to read data from a specific peripheral device, store it in a location in memory, check the parity and write it out to another peripheral device. Note that the Zilog component set includes the CPU and various general purpose I/O device controllers, while a wide range of memory devices may be used from any source. Thus, all required components can be connected together in a very simple manner with virtually no other external logic.

General Purpose Registers

There are two matched sets of general purpose registers, each set containing six 8-bit registers that may be used individually as 8-bit registers or as 16-bit register pairs by the programmer. One set is called BC, DE and HL while the complementary set is called BC', DE' and HL'. At any one time the programmer can select either set of registers to work with through a single exchange command for the entire set. In systems where fast interrupt response is required, one set of general purpose registers and an accumulator/flag register may be reserved for handling this very fast routine. Only a simple exchange command need be executed to go between the routines. This greatly reduces interrupt service time by eliminating the requirement for saving and retrieving register contents in the external stack during interrupt or subroutine processing. These general purpose registers are used for a wide range of applications by the programmer. They also simplify programming, especially in ROM based systems where little external read/write memory is available.

Arithmetic & Logic Unit (ALU)

The 8-bit arithmetic and logical instructions of the CPU are executed in the ALU. Internally the ALU communicates with the registers and the external data bus on the internal data bus. The type of functions performed by the ALU include:

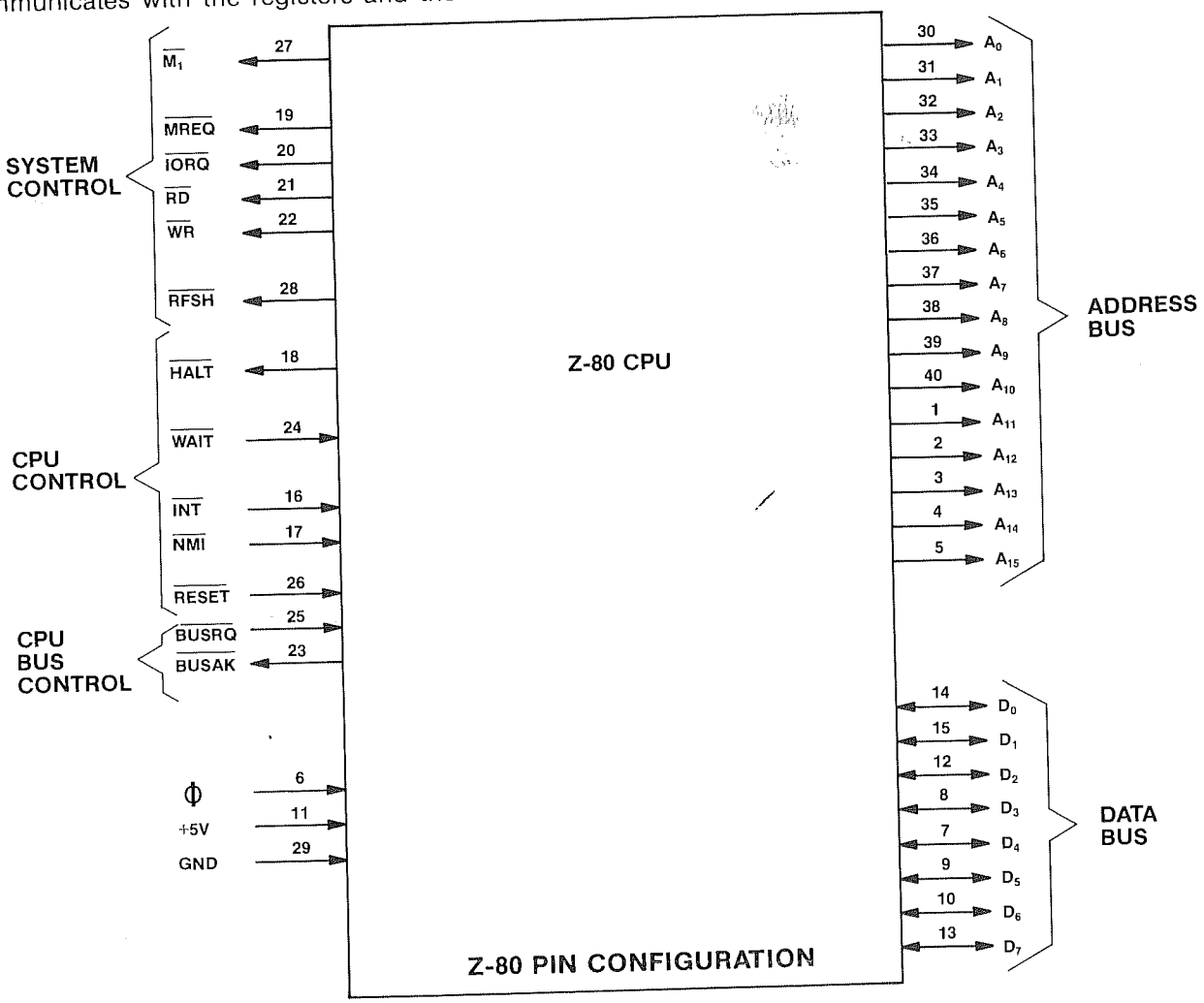
Add	Left or right shifts or rotates (arithmetic and logical)
Subtract	Increment
Logical AND	Decrement
Logical OR	Set bit
Logical Exlusive OR	Reset bit
Compare	Test bit

Instruction Register and CPU Control

As each instruction is fetched from memory, it is placed in the instruction register and decoded. The control sections performs this function and then generates and supplies all of the control signals necessary to read or write data from or to the registers, control the ALU and provide all required external control signals.

Z-80 CPU Pin Description

The Z-80 CPU is packaged in an industry standard 40 pin Dual In-Line Package. The I/O pins are shown in the below figure and the function of each is described.



A_0-A_{15}
(Address Bus)
Tri-state output, active high. A_0-A_{15} constitute a 16-bit address bus. The address bus provides the address for memory (up to 64K bytes) data exchanges and for I/O device data exchanges. I/O addressing uses the 8 lower address bits to allow the user to directly select up to 256 input or 256 output ports. A_0 is the least significant address bit. During refresh time, the lower 7 bits contain a valid refresh address.

D_0-D_7
(Data Bus)
Tri-state input/output, active high. D_0-D_7 constitute an 8-bit bidirectional data bus. The data bus is used for data exchanges with memory and I/O devices.

M_1
(Machine Cycle one)
Output, active low. $\overline{M_1}$ indicates that the current machine cycle is the OP code fetch cycle of an instruction execution. Note that during execution of 2-byte op-codes, $\overline{M1}$ is generated as each op code byte is fetched. These two byte op-codes always begin with CBH, DDH, EDH or FDH. $\overline{M1}$ also occurs with $\overline{IORQ}$ to indicate an interrupt acknowledge cycle.

MREQ
(Memory Request)
Tri-state output, active low. The memory request signal indicates that the address bus holds a valid address for a memory read or memory write operation.

IORQ
(Input/Output Request)
Tri-state output, active low. The $\overline{IORQ}$ signal indicates that the lower half of the address bus holds a valid I/O address for a I/O read or write operation. An $\overline{IORQ}$ signal is also generated with an $\overline{M1}$ signal when an interrupt is being acknowledged to indicate that an interrupt response vector can be placed on the data bus. Interrupt Acknowledge operations occur during M_1 time while I/O operations never occur during M_1 time.

RD
(Memory Read)
Tri-state output, active low. $\overline{RD}$ indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.

WR
(Memory Write)
Tri-state output, active low. $\overline{WR}$ indicates that the CPU data bus holds valid data to be stored in the addressed memory or I/O device.

RFSH
(Refresh)
Output, active low. $\overline{RFSH}$ indicates that the lower 7 bits of the address bus contain a refresh address for dynamic memories and the current MREQ signal should be used to do a refresh read to all dynamic memories.

HALT
(Halt state)
Output, active low. $\overline{HALT}$ indicates that the CPU has executed a HALT software instruction and is awaiting either a non maskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOP's to maintain memory refresh activity.

WAIT
(Wait)
Input, active low. $\overline{WAIT}$ indicates to the Z-80 CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter wait states for as long as this signal is active. This signal allows memory or I/O devices of any speed to be synchronized to the CPU.

INT
(Interrupt Request)
Input, active low. The Interrupt Request signal is generated by I/O devices. A request will be honored at the end of the current instruction if the internal software controlled interrupt enable flip-flop (IFF) is enabled and if the $\overline{BUSRQ}$ signal is not active. When the CPU accepts the interrupt, an acknowledge signal ($\overline{IORQ}$ during M_1 time) is sent out at the beginning of the next instruction cycle. The CPU can respond to an interrupt in three different modes that are described in detail in section 5.4 (CPU Control Instructions).

NMI
(Non-Maskable Interrupt)
Input, negative edge triggered. The non maskable interrupt request line has a higher priority than $\overline{INT}$ and is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop. $\overline{NMI}$ automatically forces the Z-80 CPU to restart to location 0066H. The program counter is automatically saved in the external stack so that the user can return to the program that was interrupted. Note that continuous $\overline{WAIT}$ cycles can prevent the current instruction from ending, and that a $\overline{BUSRQ}$ will override a $\overline{NMI}$.

RESET
Input, active low. RESET forces the program counter to zero and initializes the CPU. The CPU initialization includes:

1) Disable the interrupt enable flip-flop

2) Set Register I = 00_H

3) Set Register R = 00_H

4) Set Interrupt Mode 0

During reset time, the address bus and data bus go to a high impedance state and all control ouput signals go to the inactive state.

BUSRQ
(Bus Request)

Input, active low. The bus request signal is used to request the CPU address bus, data bus and tri-state output control signals to go to a high impedance state so that other devices can control these buses. When BUSRQ is activated, the CPU will set these buses to a high impedance state as soon as the current CPU machine cycle is terminated.

BUSAK
(Bus Acknowledge)

Output, active low. Bus acknowledge is used to indicate to the requesting device that the CPU address bus, data bus and tri-state control bus signals have been set to their high impedance state and the external device can now control these signals.

CLK
(Clock)

Single phase TTL level clock which requires only a 330 ohm pull-up resistor to +5 volts to meet all clock requirements.