This is the text I had prepared for the “Vintage Computer Repair 101” talk I gave at Tandy Assembly 2021. I did not use this verbatim but rather as a guide. I posted a PDF of the slides at someone’s request and then it occurred to me to post this text as well. My notes as to when to change slides in still present, i.e. (*** CoCo pic) to give you an idea which slide it is referring to. I don’t have time at present to edit this into a proper paper but hope this is still of use to some. – Jeff

Introduction

My I’m Jeff, from the Ozark mountains in Missouri. That is how I start out all my videos, but as Paul Harvey used to say, here is the rest of the story. While I have lived in Missouri most of my adult life, I grew up just a stone’s throw up Highway 68 in Champaign County, near Urbana. When I was young our Radio Shack was in the front of the Hatten and Enright Pharmacy in Urbana and like many folks the Radio Shack was my first exposure to computers. The only Tandy computer I purchased from new was a PC-6 Pocket Computer from the Radio Shack in the Upper Valley Mall here in Springfield. Coming to TASM this year is literally and figuratively coming home for me.

Motivation

I’m here today to talk about repairing the vintage computers we all love. We can all related with the story of person who finds their childhood computer in their parents’ loft. After powering it up they find it no longer works and turn to the various online forums for help. Well intentioned folks try to help with suggestions like, “It could be the RAM.” Or “I had a problem like that, and it was U7.” This often leads to someone growing frustrated after randomly swapping part after part trying to get their treasured old computer working again.

It pains me to see folks struggling like this when trying to diagnose and repair their old machines. My hope is that by passing on what I have learned over the years about using systematic approach to troubleshooting that it will help others in repairing their own equipment.

The plan

We’ll start off by talking about what a ‘systematic approach’ means. Then we’ll look about how to apply this approach to fixing our computers covering a general outline of the steps to take with some tips and examples as well. We’ll then touch a bit on the tools and test equipment needed to do this type of work and hopefully we’ll have some time for some questions.

Systematic Approach to Vintage Computer Repair

What do I mean by systematic approach? Simply put it just means to do things in a logical order. For example, what do you do when you can’t find your car keys? Do you randomly search your house and yard? Probably not, you likely retrace your steps. Let see: You got out of the car, unlocked the house door, put the groceries in the fridge, took your coat off, and then answered the phone. Ah, there your keys are, on the counter where you set them down to jot down a note while talking on the phone.

When diagnosing our computers, we also want to use a systematic approach, we want to do things in a logical order. Every repair is a bit different, there are no GPS guided directions here, sometimes you can turn the wrong way, that's OK. Just back up a few steps and continue on. When the going gets tough taking a break can work wonders. I can't count the number of times a solution has occurred to me after a good night's sleep.

So, what is our first step then? I suggest starting with your four senses.

Smell – Does it smell like the 1980s? Does it smell musty or burnt? If it smells musty it may have been in a damp location and have corrosion issues. The lingering smell of burnt electronics tunes you in as to what to look for.

Sight – a quick visual inspection can save a lot of time later on. Does there look like a charred place on the PCB? Is there evidence of leaking capacitors or battery corrosion? Also look for signs the board has been worked on before. An attempted repair job or modification that did not go so well might be a source of lots of new problems.

Sound – When you get to the stage of powering up the computer does it make funny sounds? Does power supply squeal, or click? Does the disk drive make grinding sounds? If it is supposed to make a beep or boop at start up is that happening?

Taste – When you taste….um, on second thought don't taste your old computer! Let's just stick to the first three senses.

After our sensory journey of discovery, we take stock of what we learned and decide how to proceed. If we found the circuit board is very dirty, we might want to give it a clean first. If we find burnt components or leaking capacitors that will help guide out next steps. Providing that it looks like it should still work we delve into more of the electrical arts.

Stage 1 checks: The three first things I typically check are Power, Clock, and Reset: **PCR**.

POWER - The computer will have one or more voltages supplied to it and may in turn generate other voltages internally. Ideally, we would want to test the power supply separately before using it to power up the computer. This can be difficult to do on same machines though. We'll touch on this later if there is time.

For example, this Model 100 slabtop is powered by 4xAA batteries or a 6V wall wart. Internally it has a DC-DC converter that takes in the 6V and outputs +/- 5V and trickle charges the memory backup battery. A desktop computer typically has a sperate external or internal power supply that is fed by the AC line voltage and outputs a lower DC voltage to the main board (*** CoCo pic). Some motherboards have will accept a lower AC voltage and produce the required voltages right on the board (*** VIC-20 pc) and others do a combination of both (*** C64 pic).

Whatever our power supply configuration is we want to make sure that the proper voltages are getting to the mainboard and that any voltages regulated on the mainboard are also within specification.

On our Model 100 they nicely provided some test points for +5V, -5V and battery voltage. If your computer has a separate external or internal supply, try to find the pinout of the power supply connector so you will know what voltages are on what pins. For onboard voltage regulators I like to check at the regulator for the input and output voltages. It is also good practice to test for the output voltage at the device/chip it is powering. This helps you catch an issue like a break in a trace, etc. (RAM +/- 5V, 12V – SID 9/12V)

Common power faults – Power supplies have a tough job. They generate a lot of heat which is hard on components. Capacitors are especially vulnerable to heat. Many of the components are also relatively large and heavy which when combined with heat can cause cracked solder joints. Look for burnt diodes and bad linear regulators.

CLOCK – The clock is the drum beat that keeps the whole machine working in unison. Without it nothing works. While I say 'clock' there can in fact can be multiple clocks generated form one source. A computer might have a section of circuitry dedicated to generating the needed clock signals, while some processors like the 8085 in our Model 100 here have the clock generation functionality built in. (* M100 8085 clock pic.)

A typical clock generation circuit might consist of a crystal which is connected to a few inverters, or a crystal connected to a dedicated clock generation chip, etc. (*** invertor generator pic) Whatever the arrangement is you will typically want to test for the clock signal at the output from where it is generated. Measuring the crystal directly will not work. We want to make sure the clock signals are present and are the correct frequency.

Common clock faults – While crystals can fail it has been my experience that they are fairly reliable. The ICs involved in the clock generation are typically suspect. A 'simple' clock circuit like this which uses a few inverter gates can be very particular of not only which logic family is used but also which particular brand of IC. This can catch you out if you think the inverter chip is bad and find the clock still does not work after replacing the chip. A failing crystal might mostly work with one clock generation chip but not another.

RESET – When you turn on a computer the ICs will start up in an unpredictable state. The RAM will contain random values and the CPU might think it needs to run code from an invalid address. It is total chaos. (*** reset circuit in CoCo1 pic)

To get everything started off on the right foot a reset circuit is used. This circuit provides a reset pulse at power on to the reset input of chips such as the CPU. When these chips see this signal, they know to reset all of their internal registers to a default setting. This reset pulse needs to be present and of sufficient length to allow the system to stabilize before the signal is released.

Without a proper reset signal the computer may not seem to do anything at all, it may look dead. It might also start up in various random non-working states but with some signs of life. This reset circuit might be as simple as an RC circuit as in the CoCo 1, or a it more advanced using a 555 timer or it may be incorporated with other functionality such low power/inactivity shutdown such as on our Model 100. (*** reset circuit in M100 pic)

Common reset faults – Reset problem can come from leaking capacitors, failing ICs, and in the case of our lovely Model 100s the evil flux they used at manufacturing.

Stage 2 checks: Before we dive into the Stage 2 checks take note that all of the tests we did in Stage 1 normally do not require any soldering, swapping of parts or special test cartridges, etc. If we have gotten to this point without finding a fault, we at least know our computer has the basics it needs to function. However, since our machine is still not working, we need to dig a little deeper. This brings us to **BADS**.

BADS: Bus, Address Decoding, chip Selects – In order to get an idea of what our computer might be doing or not doing we need to snoop a bit on all of the major internal signals. We want to use the oscilloscope to look at all the address lines and data lines for signs of activity and 'normal' looking signals. A logic probe can also be used if that is what you have but won't catch all the things the scope will.

Our address and data lines will often be routed through various chips which control the signal flow to and from parts of the system. For example, if the CPU is trying to read data from the bus we want to make sure that correct device and only the correct device is writing the bus. Sometimes you can get a goofy looking signal like this (***BUS CONVENTION PIC) which is the result of 'bus contention'. One of the gates on this chip is internally shorted to another so we get a sort of 'cross talk' which confuses everyone involved. What was really odd about this particular Model 100 is that is mostly worked but had screen corruption due to this problem.

Checking for the proper address and data bus signals can be a bit tricky with a processor like the 8085 used in this M100 which multiplexes the data bus with the lower 8 bits of the address bus. That is these two separate signals use the same pins on the processor chip use them at different times. Some external chips are used to latch the address and data signals so make them available to the rest of the computer. When this transition is functionality takes place, you can get some odd-looking signals like this. (*** 8085 multiplexing pic) This is not an issue for the computer as it is not trying to read or write to the bus at that exact time.

There is a caveat here that if you have a bad ROM or similar condition that causes the processor to run nonsense code it might actually run a processor halt instruction or something similar. You'll see bus activity stop. If you run into this sort of issue press the reset button to restart things and see if the bus activity comes back. Since we know our reset works, we can use it with confidence.

There are also chips that do things like address decoding and chip selection. What is this? We have all of these chips hanging on our address and data busses. To keep them from trying to talk or listen to the bus when they should not, we need to select or enable each chip when it is its turn to use the bus. When the chip is not selected it does not care what data is in the bus and it won't try to output data on the bus. If the chip is selected at the wrong time or not selected at all you get chaos.

This address decoding/selection can be done with individual logic chips or highly integrated logic chips that roll all the individual chips into one package. A breakdown in part of this decoding and selection logic is not uncommon and it can be a confusing problem to diagnose.

Common faults – Common sources of bus contention issues are address latches, RAM chips, address decoding logic. Since there are usually several chips tied to the same bus it can be a bit of a guess as to which one is faulty. We can often narrow it down by looking at what chips share the two signals that are in contention. This is a common fault on some computers, like the PLA on the C64.

If there is a problem with a chip on the bus you can sometimes remove those chips which are not needed for the computer to boot up. For example, you probably don't need a keyboard latch chip or the sound chip if they are suspect, especially if they are sockets, you can pull them and try it again. You don't need a replacement chip to plug in to do this sort of test.

Stage 3 checks: Testing aids – We have made it to stage 3 where everything looks like the computer should run but it does not. This can be a very tough spot to be in. Do we have bad RAM which is preventing the machine from booting? Maybe it is a bad ROM and the computer is trying to run nonsense code. It could also be running just fine but the video output is not working correctly.

Test carts / harnesses - This is where a diagnostic test utility can really help. Depending on the system you are working on there may be diagnostic cartridges, diagnostic ROMS, and even a test harnesses that route signals around the machine to try and automatically test the I/O.

Note that we did not start with a test cartridge or harness. We first made sure that we had the proper voltages, clocks, reset and bus activity. If we did not have these parts of the system working the test cartridge would do us very little good. These sorts of aides depend on the machine being somewhat functional, functional enough to run a simple diagnostic program.

For example – the test harness for the M100 we have here replaces the system ROM with an elaborate test program. It will run a series test on the RAM. Since it does not know if the RAM is good or not it can't use stack, which is RAM, so no subroutines, just a long series of repetitive code. If it finds at least one good RAM module it will continue to run a more extensive series of tests. If it cannot find a good RAM module the program stops.

This is very useful for finding a bad RAM module which can stop the machine from booting. The very first M100 I worked on had this problem and it took ages to diagnose as I did not have a way to test the RAM at that time. It is also very useful tool for testing all the I/O ports, keyboard and LCD.

Diagnostic ROMs or cartridges for other computers work in a similar manner. Typically, they need to replace the main system ROM so they can do a series of low-level checks that depend on as few resources as possible. Many have the capability to flash the screen, make a sound, etc. to indicate bad RAM that might prevent a machine from booting.

Note again that we needed a computer that was 'mostly' working before getting to the point of using a testing utility like this. While the diagnostic tests done are often very good, they can't tell you which is bad with any accuracy. What they can tell you is that they ran a test on a certain chip and got back results that were not expected.

For example, our diagnostic ROM might tell us a RAM chip failed a test. Is the Ram chip actually bad? Perhaps it is a fault in the chip selection logic for that chip? Maybe it is a broken trace? While the diagnostic ROM can't tell us an exact cause of the problem is has pointed us in the right direction and limited our search.

I had the diagnostic harness on this M100 and it reported a bad RAM module and highlighted address line 10. This could indicate that the module was indeed bad, or it could indicate that this signal was not making it to the module or was disturbed in some way. Looking at A10 it was apparent there was an issue with the edge of the signal. Removing the module and running the diagnostic test again showed the problem with A10 went away so it was indeed a bad module. If the problem with the excessive slope to the edge of the signal remained, then we would need to look elsewhere. Some chips might be more sensitive to out of spec signals than others so even though our RAM chip might not be working right with this signal, it does not mean it was the cause.

As an aide this slope to the edge of the signal looked a lot like capacitive loading of the signal. I later soaked this module in alcohol for a few hours and then cleaned with electronics parts cleaner. When it was dry it tested out fine. I suspect it was contaminated with the evil flux used on these M100. Did I mention the flux is evil?

You might be saying, "Hey Birt! Why bother with all of this? Just pull all the chips and test in another board!". Anytime you de-solder a chip from a board you risk damaging the board or chip. It is also easy to spend a lot of time de-soldering chips which were fine all along. Because we have went about our investigation in a systematic manner, we are more confident that a part needs to be removed before pulling it out, we are reducing the risk to the PCB and chip.

Tools

We have talked about a general methodology for diagnosing our computers and I have mentioned a number of different diagnostic tools that can be used. So, you might be thinking, "Hey Birt! What sort of tools do I need to do all of this and what's it gonna cost?" That's a darn good question and I'm glad you asked. I'll try to make some general recommendations in two broad categories of 'Basic' and Advanced'.

Basic tool set

Hand tools - The basic tool set starts off with the things you use most often like hand tools: screwdrivers, pliers, side cutters, wire strippers, etc. You can start off with the more commonly used things like Philips and flat blade screwdrivers, basic wire strippers and add in the more specialized tools over time.

You don't have to break the bank to get started but my advice is to shy away from bargain tools. Suppliers like Jameco, Adafruit, and Sparkfun will often have some stock of hand tools which you can have some confidence in. Places like Amazon do offer some good tools if you know that to look for. I was recently in need of replacing my small side cutters and suggestions from Twitter ranged in price from $10 to $65. I found some made in the USA Xuron micro-cutters on Amazon for $15.

Multimeter - As far as electrical test gear, the one must have item is a multimeter. A good quality meter is your friend, the $5 special from Harbor Freight is more of a 'frenemy'. If you are concentrating on low voltage things even a cheapo meter is safe enough. If you need to measure mains voltages, then a quality meter is a must. The really cheap ones have very shoddy overvoltage, overcurrent and blast protection and can wind up being a bomb in your hand. You can still get a good meter for < $50 though. There is a thread on the EEVBlog forum that reviews various meters.
https://www.eevblog.com/forum/testgear/multimeter-spreadsheet/

Component tester - Another handy, low cost, piece of test gear is the multifunction component tester. These come complete or in kits. This one I bought years ago for about $10 as a kit and 3D printed a case. Now you can get them for $20 complete with a case and color screen. These do a good job at a go/no-go type of test on a wide variety of components. You can test capacitors, transistors, etc. They are not super accurate, but I have compared the ESR values reported by this one to the fancy equipment we have at work and it was within a few %. Good enough to know if the capacitor is healthy or not.

Logic Probe - To get an idea if your bits are twiddling a good low-cost tool is the logic probe. This is the logic probe I bought from Radio Shack in high school. It still works and it is still useful. It can tell you if a signal is hi, if the signal is low or if it is quickly pulsing. You can tell if a signal is present and if it is changing but for other problems like the bus contention, we talked about previously, they are not much help.

Soldering - If you are going to be removing and soldering in parts you will need a soldering iron. You can spend $5 on an iron or $5,000. My recommendation for getting started is a decent quality, adjustable, temperature-controlled iron that you can get a selection of different tips for. Using a poor soldering iron is the easiest way to ruin a circuit board. These days you can get an OK iron from Adafruit for $20 and a decent soldering station for $50.

Desoldering - For removing solder there are a few good options to start out with. This is my Radio Shack solder sucker from high school which has been kept going with parts from the newer, and crappier, plastic variations of this same tool. There are more 'professional' versions of this tool which have a silicone tip available in the $20 range that do a good job. De-soldering braid is also a low-cost de-soldering aid that is still handy when you have one of the big $$ rework stations. De-soldering braid, or wick, is a flat copper wire mesh with flux inside. You place it over the joint, head the braid and the solder is drawn into it.

Magnifier - Being able to see what you are working on is essential. After we pass 40 our near vision goes downhill and we need some help. My go to tool for this is a lighted magnifier. The one on my bench I inherited from my Dad, but you can buy them very reasonable these days. Something like a headband magnifier can also be used just as well.

Advanced tool set – The advanced tool set tends to consist of those things which are fancier and typically more expensive.

Oscilloscope - The most useful ‘advanced’ tool I think is the oscilloscope. An oscilloscope lets you ‘see’ an electrical signal. Simply put think of it like a multimeter that takes tens of thousands of readings a second and draws a graph on the screen. The vertical axis is voltage, and the horizontal axis is time. This is the tool that lets you find out not only if a signal is there, but it is ‘looks’ right. For pinpointing issues like bus contention, it is the tool to use. (*** bus contention pic)

Logic analyzer - What if you need to measure more than a few bus signals at a time to try and work out if that data being read out of a ROM makes sense? This is the realm of the logic analyzer. A logic analyzer can measure many more signals at once than an oscilloscope but like the logic probe is limited to knowing if a signal is high or low.

To use it you hook up one wire per signal, and ‘record’ for a few seconds and analyze the data. This requires more in-depth knowledge of the data you should be seeing. These range in price for $50 to as much as you want to spend. My little one here is only 8 channels, but it was only about $100 when it came out years ago. (***Example of Model 100 RAM fault.)

Rework station – If you really get into this sort of thing you will no doubt be tempted by a desoldering tool and/or a full rework station. A desoldering tool is a soldering iron with a hollow tip and vacuum pump. It can melt the solder and suck it out of the hole very proficiently. You can find these in a variety of prices ranges these days, even sub $100, many of which are clones of name brands, like Hakko, or clones of clones. It pays to ask what other’s experiences are with a particular tool you are interested in.

A rework station is able to power and control multiple irons, or handpieces at a time. You can have a soldering iron, desoldering tool, and a soldering tweezer all available at once for example. They are more of a professional type of equipment and thus more expensive. I had used a Pace MBT system at a previous job and really wanted one but could not justify the expense. So, I watched eBay for ages until I found a system in good shape at a price I could afford. The tips can be expensive, so I again turn to eBay to pick up NOS tips.

Test harness - I have included the various test cartridges, ROMs and harnesses as ‘advanced’ tools as well. While these are usually not usually very expensive items, they typically are not much use without already having the basic tool set. They are also a dedicated purpose tool. This test harness only works with the M100. T102, M-10, etc. The C64 diagnostic harness only works for the C64 and C128, etc.

These testing aides can be great time savers and can even work as a pretty good automatic test of various system I/O ports. They can’t replace our basic tool set nor are they a substitute for a systematic troubleshooting approach, we can’t expect to plug a diagnostic tool in and have it tell us what part to replace. These are the types of tools you invest in if you work on a lot of a certain type of computer or you are very enthusiastic about fixing a certain one.