**Introduction to repairing arcade game PCBs.** Version 2.05

*Note: This introduction is generic in nature however in the examples provided the game being described is a Atari Centipede PCB.*

**Lesson1: CPU signal Basics**

**Introduction:**

Repairing arcade game PCBS seems like a magical and arcane skill, however it's simply basic troubleshooting. Understanding how a system works, breaking it down into small distinct subsystems and testing the various subsystems in a logical way. If you have a logic probe (or oscilloscope) and a Fluke 9010a with the correct CPU pod, there's a very straightforward approach to troubleshooting that can fix probably 90% of all problems.

**The arcade game PCB is nothing but a very simple computer**:

The arcade PCB is a very simple "single purpose computer". For the arcade game to work and be useful it needs the following components:

1. A working CPU
2. Working ROMs (game code) (and supporting circuitry)
3. Working RAM (and supporting circuitry)
4. Working video display circuitry
5. Working input circuitry

Let's look at item number #1 (A working CPU). Obviously the game will not work without a working CPU. If the CPU is bad you probably will actually see something on the display, but it will probably be garbage, this is because the video display circuitry is actually separated from the CPU for the most part. The video unit simply reads a certain block of memory periodically and based on what's in that memory will display something to the screen. If the CPU is not active the memory will still be there but filled with un-initialized garbage.

 Brian Brzezicki

So how do we check the CPU? There are a few steps that must be done before even looking at swapping the CPU:

1) You need good DC power (specifically +5V on most arcade CPUs)
2) You need to get a reset signal when the CPU powers up
3) You need a valid clock signal (unless the CPU has the clock internal)

**Checking Power:**

Without good power there is no chance your CPU and game will work, NONE, let me repeat that NONE. Power is the foundation of any electronics system. So how do you check your power? Easy

**Checking +5V power**

- Get a multi-meter and set to the DC power reading
- Find a 74LSxxx series chip on the board (there should be LOTS of them)
- Consider the edge of the 74LSxxx chip with the "notch" in it to be the top.
- Find the GROUND pin on the 74LSxxx (usually the bottom left pin of the chip)
- Find a +5V point your 74LSxxx chip (usually the top right pin of the chip)

Measure the power between these two points. If it's not between +5V - +5.15V adjust the power supply until it is in that range. If your power supply does not adjust it might be a bad power supply or your board might have some type of short. (Unfortunately shorts are not easy to troubleshoot). If the power is OK move on to the next step.

**Reset signal:**

When a computer starts up the CPU and memory are in an indeterminate (mostly RANDOM) state, and the system will not actually work. Computer systems will have either a reset circuit or a watchdog circuit which is responsible for sending a quick reset signal to the CPU. When the CPU gets this signal it will jump to a hard coded memory address and start running code. This bootstrap code initializes memory, and usually tests the system to some degree and then starts the normal program.

You want to test the CPU to verify that it gets a reset signal. To test this all you need is a logic probe (preferably with audio output). If your probe has different settings ensure that the logic type is set to **TTL** and **pulse** (as opposed to **memory** and **CMOS**).

**Using a logic probe to test the reset signal.**

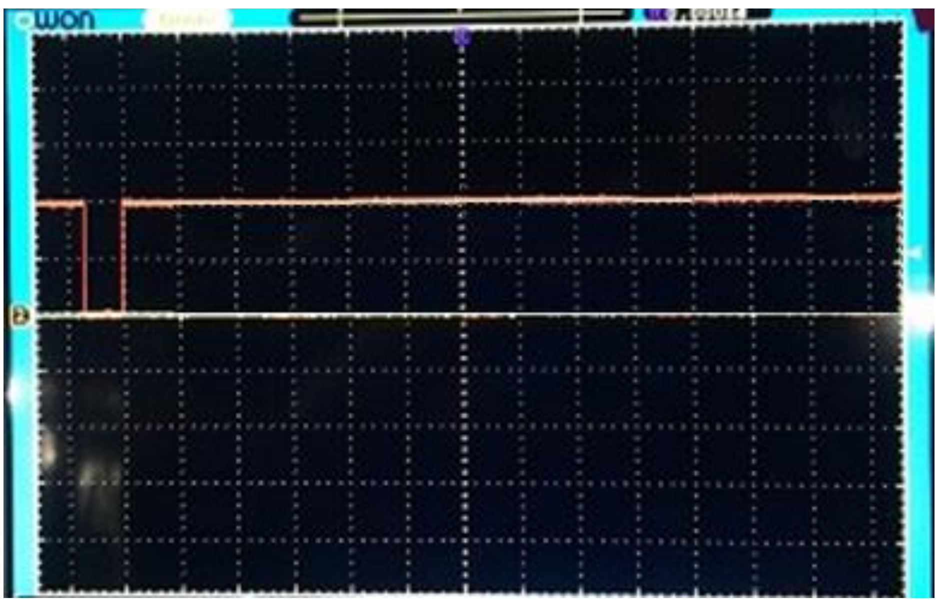
- Take the logic probe, attach the red and black wires to +5V (red) and Ground (black) and find the pin on the CPU that is the reset. Touch the logic probe tip to the reset pin and turn the system on and see what happens.

On a Atari Centipede with a 6502 processor the RESET pin is pin 40 (top right). When the pin is “low” the CPU is getting the reset signal (it’s called NOT reset, that is RESET is ENABLED on LOW). When the system turns on the signal should be high, followed by a quick transition (that you’ll hear) to low then return and stays high.

If you have an oscilloscope you can use that instead of a logic probe. However you'll need to consult your manual to learn to set a trigger to detect the state change.

**Using an oscilloscope to check the reset signal.** (Note oscilloscopes are complex devices and each model will have different controls to manage the settings. This is meant to be a generic example).

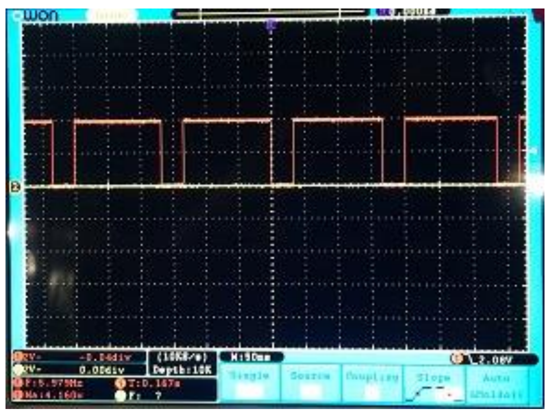
- Connect your oscilloscope probe to the oscilloscope channel 1, and the ground lead of the probe to the ground point on the Centipede board.
- Configure your oscilloscope to trigger on falling edge and put it in single trigger mode
- Configure the trigger point to be somewhere halfway between +5 and Ground (+2.5V is great)
- Configure the horizontal grids to be set to 1 microsecond (1us)
- Connect the oscilloscope probe to pin 40 of the 6502 on the centipede board.
- Without lifting the oscilloscope probe from the test point, turn the game on.
- You should see a high signal, a short dip to ground, then back high like the picture below



**Troubleshooting RESET tips:**

- If there is no RESET signal that's a problem, you need to  troubleshoot the reset circuit.
- If the reset signal is always high and never goes low, hit the reset button if one exists and see if it goes low, if it does not, check the circuit from the reset switch to the reset pin on the CPU. Note it is very hard to catch the reset on start up, so if you don't see the low transition however it does go low when the reset is hit, then you may skip this step.
- If the reset signal is always low, something in the reset circuit is not working, check the components in the reset circuit from switch to the RESET pin.

Most likely on a broken game with a valid RESET and WATCHDOG circuit, you will find the reset signal is pulsing steadily as shown in the image below. This is usually caused by the watchdog circuit detecting the system is not running correctly and constantly trying to reset the system.  If you find that the board is properly receiving the initial RESET signal then the system returns to a stable non-reset state most likely your game is actually working unless your watchdog circuit happens to be broken. If you are getting the signals that you should expect from a working game, but you don't have video, then you might want to skip all the sections and go troubleshooting video.



What is a WATCHDOG?

Most dedicated systems contain a WATCHDOG circuit. On these systems the main code periodically writes data to a certain memory address to signify that everything is working properly. The WATCHDOG circuit watches this memory address, if it does not see the data that specifies that all is OK, it will send a RESET to the CPU to restart the systems.

**The clock signal**

Most older games have CPUs that are externally clocked;  there is a crystal that oscillates at a certain frequency, and some counter chips that “step down the frequency” ultimately going to the clock signal on the CPU.  This clock circuitry will provide a steady transition of low to high that you should be able to hear. Without this steady clock signal the CPU will NOT work.

**Using a logic probe to check the clock signal.**

To test this signal, hook up your logic probes GROUND and +5V wires to the appropriate test points in the board, and touch the probe tip to the clock pin on the CPU (this is pin 37 on a 6502). Normal operation is that the pin will be beeping steadily transitioning from high to low states.

**Troubleshooting the clock signal with a logic probe:**

- If the clock pin is missing, stuck HIGH, stuck LOW, or beeping in a non-uniform pattern, the clock circuitry is NOT working and the game will NOT work. You need to check the circuitry from the crystal to the clock pin.
- Often on games the crystal will be missing completely, or will be broken. If it is replace it. Sometimes the crystal looks good but is not. The way to test this circuit is to look at the first output pin from the first TTL chip in the schematics after the crystal. NOTE because this is of high frequency your logic probe will probably get a weird signal that you might think is bad, you may not even get a signal and that might be OK, it’s because it’s going too fast for the logic probe. Be worried though if the output is stuck high or low.

**Testing the clock signal with a Fluke 9010A**

The Fluke 9010A with appropriate POD will not work without a valid clock signal, there for you can use it to test the clock signal.

Before we test the clock signal, let's first learn how to ensure the Fluke itself works, if you have not already done that then seen the instructions at the end of the chapter for testing the Fluke 9010a and the associated POD.

- TURN OFF the game board (always power on the fluke FIRST)
- TURN OFF the FLUKE
- Insert the POD into the CPU (in the correct direction of the chip :)
- Turn On the fluke
- Set the fluke settings in setup to “ignore active line force”
- Turn on the game board

- Hit “bus test” on the fluke, if it says “BUS TEST OK” your clock is fine, if it says “POD timeout” or "BAD POWER SUPPLY" then most likely you don’t have good voltage to the CPU POD or your clock is bad.
- Then turn OFF the game board first (never have the game board on and the Fluke off)
- Turn the Fluke off

If you find that your clock signal is not what is expected you need to troubleshoot the clock circuitry. If you do get a steady transition then proceed to the next lesson.

**Using an oscilloscope to test the clock signal** (Note oscilloscopes are complex devices and each model will have different controls to manage the settings. This is meant to be a generic example)

- Connect your oscilloscope probe to the oscilloscope channel 1, and the ground lead of the probe to the ground point on the Centipede board.
- Configure your oscilloscope to trigger on falling edge
- Configure the trigger point to be somewhere halfway between +5 and Ground (+2.5V is great)
- Configure the horizontal grids to be set to 1 microsecond (1us)
- Turn the game board on
- Connect the oscilloscope probe to pin 37 of the 6502 on the centipede board.
- You should see a signal that looks something like the picture below. If your oscilloscope has thea frequency measurement option you should see the frequency of channel 1 is about 1.5Mhz (the clock speed on centipede). If you don't have a frequency option you can align one edge of the up or down signal to a grid line, then see how many grid lines (or fractions of it) it takes to make a complete cycle. If you figure out how long each cycle is compared to the grid lines you can do some math to figure out the frequency. (For example if your grid lines are set to 1us (1 millionth of a second) and there are 3 complete cycles in 2 grid lines (2 millionths of a second) then the frequency is 3 cycles /2 millionths of a second = 1.5Mhz


*Special thanks to KLOV users TROXEL and BARITONOMARCHETTO for reading through this guide, doing the exercises, catching many many typos, and generally making this a better document.*