

# Technical Notes on Cliff Hanger

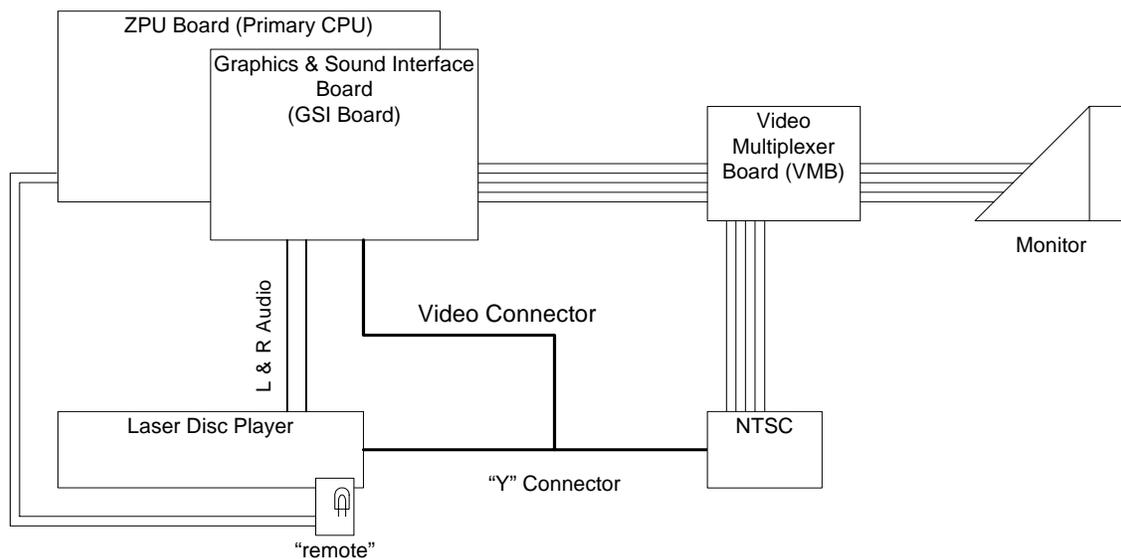
By James Bright, [quarterarcade.com](http://quarterarcade.com)

## Introduction

This is meant to help explain the Theory of Operation of the Cliff Hanger. It is not complete, so at this time I'll refer to this simply as "Technical Notes." This information was gathered from personal experience, information available at [www.d-l-p.com](http://www.d-l-p.com), and conversations with laser game "experts" like Warren Ondras. This will cover some of the material already available in the manual, but hopefully it will fill in some information gaps not necessary documented elsewhere.

## Major Components

This block diagram is based on the manual and the actual game that I worked on, which utilized the Pioneer PR-8210 laser disc player. Your set up may be different. (Power supply and UIB not shown.)



The above diagram is a simplified version of the diagram in the Cliff Hanger/Goal to Go schematic drawings. (Those schematics don't really show the NTSC card however.)

## ZPU Board

CPU board, with a Z80 running at 4MHz. The main memory map for the Z80 is:

Start	End	R	W	Chip	Notes
\$0000	\$1FFF	x		ROM 0, 1N, 2764	
\$2000	\$3FFF	x		ROM 1, 2N, 2764	
\$4000	\$5FFF	x		ROM 2, 3N, 2764	
\$6000	\$7FFF	x		ROM 3, 4N, 2764	
\$8000	\$9FFF	x		ROM 4, 5N, 2764;	Not used in ROM set "Cliff_alt.zip"
\$E000	\$E7FF	x	x	RAM 5126	Battery-backed up NVRAM.
\$E800	\$EFFF	x	x	RAM 2128	Scratch pad RAM. May be 9128, 2128, or 6116. All compatible. (NOTE: what speed rating is required?)

## Fluke 9010A ROM Signature Data

Start	End	Cliff.zip	Cliff_alt.zip
\$0000	\$1FFF	\$8D07 (cliff_u1.bin)	\$F826 (cliff_alt_0.bin)
\$2000	\$3FFF	\$6DFB (cliff_u2.bin)	\$56F8 (cliff_alt_1.bin)
\$4000	\$5FFF	\$3FEC (cliff_u3.bin)	\$B86B (cliff_alt_2.bin)
\$6000	\$7FFF	\$13A9 (cliff_u4.bin)	\$F738 (cliff_alt_3.bin)
\$8000	\$9FFF	\$D128 (cliff_u5.bin)	n/a

## Fluke 9100A ROM Signature Data

*Todo: these must be empirically collected. This is a placeholder.*

For more information about Fluke in general, see <http://tech.quarterarcade.com>. In a nutshell, this is a technical tool used for board level debugging.

## GSI Board

The Graphics and Sound Interface Board is essential for the game to fully function. Some important points to note are:

- The GSI is fed the video from the laser disc player. It scans the frames and decodes a frame number. If the player isn't connected to the GSI board *or* if the GSI board has a fault in that decoding logic, you will get errors such as messages like the player can't seek to a given frame number.
- The GSI board sends graphics to the VMB board where it is multiplexed<sup>1</sup> with the laser disc images.

---

<sup>1</sup> Meaning that this essentially acts as a switch between the inputs from the laser disc player and the graphics card.

- The GSI board has a pot that controls the image sent to the VMB. If you don't get any images from your player, but you get the graphics from the GSI board, then check this control. The VMB may not problem multiplex the signal because it may think that the GSI is "blinking out" the laser disc player. (I haven't taken out my scope to confirm this, but I'm pretty sure that's how it works. The relative voltage difference between the laser player and GSI must be a in a specific range to work properly.)
- Audio amplification is done on this board.

## Special Notes about the TMS9128NL Chip

This chip is often missing, or possibly non-functional. The problem with this is that it's an extremely hard chip to come by. Mark Capps (a.k.a. "[Big Dog](#)") has had these in the past. Even still, they are expensive if you can find one. Another alternative is to find an old Baby Pac-Man Vidiot board and "borrow" it from there. While I hate stealing parts from one classic board, a heck of a lot more Baby Pac-Man games were made. It may be more realistic to find a TMS9128 from a pulled board than a NOS source<sup>2</sup>.

The TMS9128 (See Appendix) chip is a special video processor chip that was used in early computers. I haven't been able to find a data sheet on these chips, but basically the TMS9118/TMS9128/TMS9129 chips are in the same family<sup>3</sup> (I believe the data sheet covers all three).

The TMS9128 has two supporting RAM (TMS4416 NMOS RAM) chips at 4A and 5A. These RAMs should still be available, and you can test them by going into the self-test mode (assuming you can get that far). See the Cliff Hanger manual for more details. NOTE: if you can test them in self-test mode then it ought to be possible to test them using automated test via a Fluke. I haven't had need to get into this yet, but looking at the disassembled code should help<sup>4</sup>.

The TMS9128 should have a large heat sink attached. While not mandatory, *is it a good idea to use one*. Excessive heat greatly reduces component life, and a 50 cent part that is probably available at your local Radio Shack is a great way to make the most out of that \$20 (plus!) chip.

## Adjustments

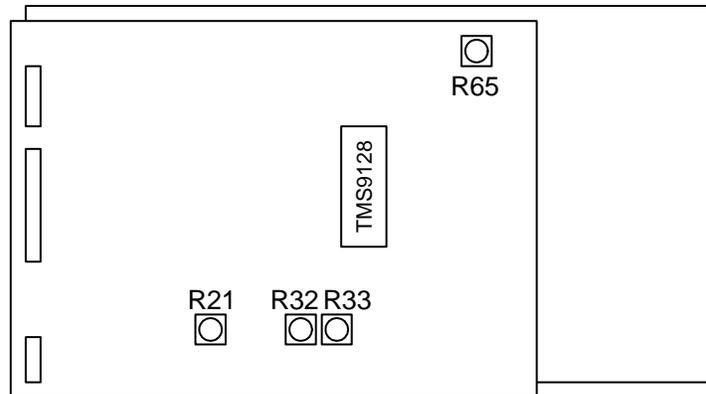
There are a few adjustments that you can make on the GSI board. This is partially from the manual and partially from my experience.

---

<sup>2</sup> At the time of this writing I have not tested using a chip from a Baby Pac although I plan on doing so in the near future. (I'll update this note when I've tried it.)

<sup>3</sup> See <http://junior.apk.net/~drushel/pub/coleco/twwmca/wk961118.html>

<sup>4</sup> See <http://www.jeffsromhack.com/code/cliffhanger.htm> , Jeff K's Cliff Hanger disassembly. If I get his permission, I will include a reference to the code in this document.



GSI (Adjustments)

The above picture shows the location of the pots.

Pot	Description
R65	“Picture Control”. This is used to set when the B-Y (?) level will make the VMB switch between the GSI generated graphics and the laser disc player. If it’s not properly adjusted, you may not get images from the laser disc player. I’ve noticed this more when I tried to use NTSC card substitutions.
R33	Balance between left & right and beep/boop sound.
R32	“Ch1” Volume for left channel/speaker.
R21	“Ch2” Volume for right channel/speaker.

### **Laser Player/Remote**

This is pretty wild stuff. For the PR laser player, the player is actually controlled by a mini remote control that ZPU uses to issue commands. In reality it’s not that much different than computers that have IR ports to communicate wirelessly. At any rate, if this board is missing, you should be able to build one yourself from the schematics in the manual. It’s just two resistors and two IR LEDs. This is how the PR-8210 player gets sent commands<sup>5</sup>. If, when you turn on your game the player starts to play, then the remote controller is most likely working. I have to build a remote control for a Cliff Hanger that I am restoring. When I get there, I’ll document the process. It’s definitely a “beginner” project as all of the necessary parts are available at your local hobby electronics store.

### **Video Multiplexer Board (VMB)**

Reproductions were produced by Mark Speath for a while (I believe he stopped making them). This board does just what it sounds like: it merges the images from the GSI (computer style graphics) with the images from the laser disc player (via the NTSC card). If you aren’t sure if your VMB is working you can try these tests:

<sup>5</sup> See <http://www.dragons-lair-project.com/tech/docs/8210info.asp> for differences between PR-8210 and PR-8210A. There may be different versions of Cliff Hanger that did not rely on the IR port.

- Unplug the input from the NTSC card to the VMB [get label]. You're essentially removing the laser disc player from the picture. Do you get any pictures during start up (blue cross hatch pattern) or anything while the game runs through attract mode? You should be seeing the words "Cliff Hanger" fly in during the opening, and then during the end the game will show the Programmer's credits and high score tables, all computer generated. If you see these images after you've unplugged the laser disc player, you have a problem elsewhere. If you do not get any images still, you will want to confirm that your ZPU and GSI board works, as well as your power supply.
- Plug the output of the NTSC card directly into the monitor. Do you get any pictures? If you do get pictures, then your NTSC card is working, and your player is working. You may have something wrong elsewhere like at the VMB or the adjustment pot on the GSI. If the image doesn't sync up, then confirm that you have it hooked up correctly. (90% of the time, sync problems are because you're not connecting the sync line properly). In my case, I was able to plug directly into the monitor from the NTSC card.

NOTE: The VMB expects positive sync from the NTSC and probably from the SGI as well. (I did not confirm that with my scope, but that's probably the case). If you substitute your NTSC with another, make sure are using positive sync into the VMB.

### ***NTSC Card***

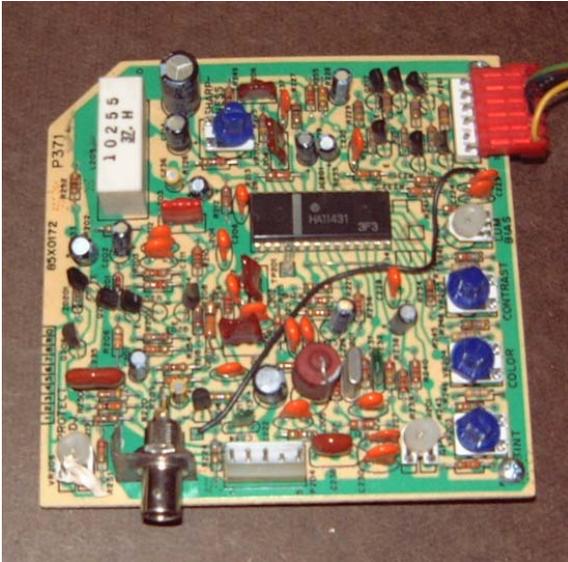
The NTSC card that game with my Cliff Hanger board was fed AC power directly from the same line as the monitor. I believe that this is correct, but I can't confirm for sure. The schematics currently available are sketchy at best, and some wires were cut in my cabinet. I was able to get my game running with the original NTSC (in black and white), but I ended up doing a substitution.



My original NTSC card. I installed the molex connector. I'm not sure where the other red wire is supposed to go.

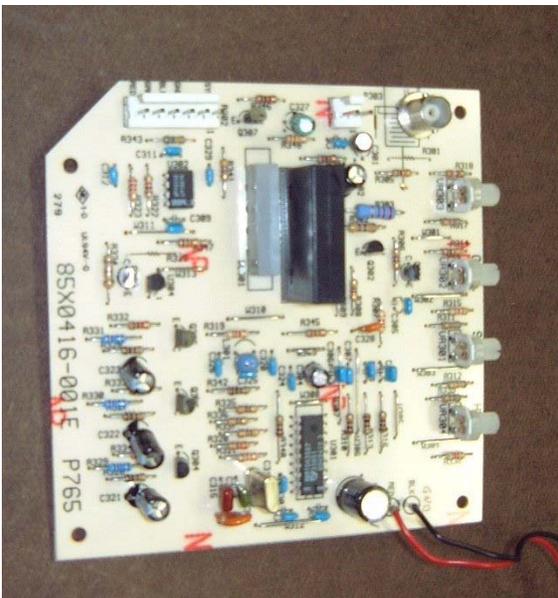
The purpose of the NTSC card is to take the output from the laser disc player and convert it into a signal that a monitor can use. This is a common function, and pretty much any NTSC card could work if they're hooked up properly (more on this in a moment). The NTSC that came with Cliff Hanger outputs R, G, B, ground, and positive composite sync. You could plug this directly into your monitor if you wanted to test that part of your set up. (You should be able to force the player into play mode and confirm that part works.)

### Using a Wells Gardner NTSC as a Substitute (original DL hardware)



NOTE: There is more than one version of the Wells Gardner card. This speaks to the "original" WG card that was installed in some Dragon's Lair/Space Ace games. Wells Gardner, until recently, was making brand new NTSC cards that do not require this modification. To do this, you would make the changes that Ruben documented here: [http://www.users.bigpond.com/tmek/wg\\_ntsc.txt](http://www.users.bigpond.com/tmek/wg_ntsc.txt). I did not try this approach, but I know that some people have done this. I'll include this in the appendix. It's publicly available on the internet, but just in case it ever gets lost.

### Using a Wells Gardner NTSC as a Substitute (newer card)



The brand new NTSC cards made by WG are very nice and suitable for replacing the original Cliff Hanger NTSC cards. They run off of +12V and you can tap into the original power supply. The only downside is that they output negative composite sync. In order to use this card, you have to build a simple inverter circuit for the sync line before it's fed into the VMB.

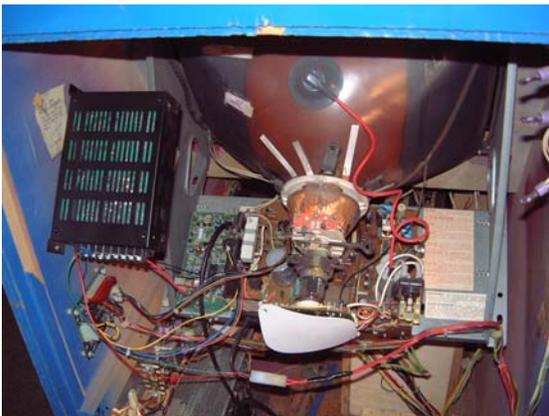
## Using an Electrohome NTSC as a Substitute



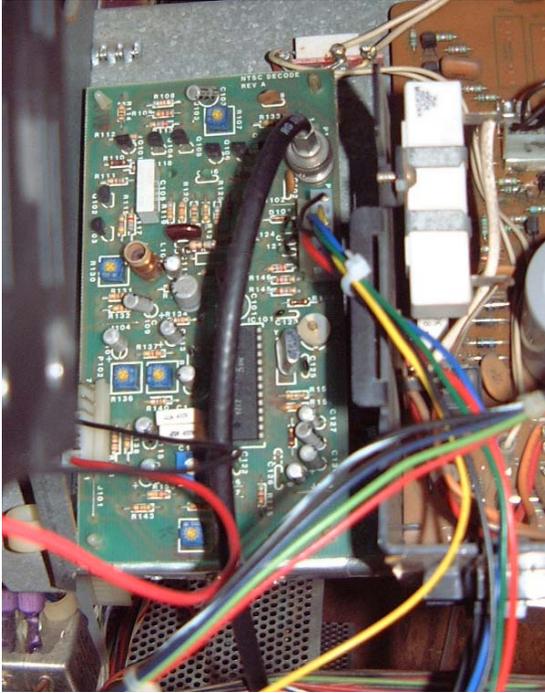
(Picture from [Dragon's Lair Project](#) as I had already installed my Electrohome card)

This is probably the best alternative if you don't have a working original card. Although it takes +16V as input, you can run that off of the +12V line and you should be fine. I noticed a little distortion, but it was acceptable to me. Since this card outputs positive composite sync, it was nearly plug and play. I had to modify the connector to the VMB because the pinout is different than the original, but that was about it.

NOTE: to fix the distortion, I simply installed a switching power supply and cranked it up as high as it could go. I just tapped into the A/C line where the monitor plugs in (it was convenient), and I only used it for the NTSC. What was happening was that whenever the game had sounds, the video was distorted, most likely because the amplifiers were drawing a little more current.



The back of my Cliff Hanger with the switcher installed and the NTSC mounted on the monitor chassis.



A close-up on the NTSC. It was a bit painful getting it to fit, but I was able to mount it quite nicely. (This isn't an original monitor, but a borrowed G07).

***Universal Interface Board (UIB; Not shown in block diagram)***

Basically a diode matrix board that sits between the control panel and the ZPU.

# Appendix A: Using an NTSC card in a JAMMA cabinet

*This procedure can also be used to retrofit a Cliff Hanger*

Source: Ruben Panossian, [http://www.users.bigpond.com/tmek/wg\\_ntsc.txt](http://www.users.bigpond.com/tmek/wg_ntsc.txt)

## Pin-outs for the Wells Gardner NTSC Decoder

This decoder is made on a small square PCB and has the model number DY-1-0 at the end of the PCB with the corner cut off. There are two basic revisions, 85X0172 (P371) and 85X0311 (P600).

<p>Pin-outs: -----</p> <p>P204: 1 - Vcc * 2 - Vcc * 3 - GND 4 - Key 5 - Blanking Input</p> <p>P203: 1 - Positive composite Sync out ** 2 - Key *** 3 - GND 4 - Video Blue out 5 - Video Green out 6 - Video Red out</p>	<p>Notes:</p> <p>* The Vcc is dependant on the revision of the PCB. The earlier revision PCBs (P371) have a Vcc of +15V DC. The newer revision PCBs (P600) have a Vcc of +12V DC, although, +12V still works with the earlier (P371) revision PCBs. The difference between the two versions is with R256. It is used to lower the voltage from +15V to +12V, on the earlier revision PCBs. The newer PCBs, using regulated power supplies as Vcc source, just have a wire link in the place of the resistor. The newer revision also did not have the over voltage protection circuit (components not installed) as the Vcc was intended to be supplied from a regulated power supply.</p> <p>** This pin will have negative Sync output on it after the modification</p> <p>*** This pin was used to supply +12V and blanking in some (modified) versions. They were altered to suit some games, so take note when using an old revision decoder.</p>
---	--

## How can I use this NTSC decoder in my JAMMA cabinet?

To enable this decoder to be used with a JAMMA cabinet, without modifying the harness or monitor chassis, the NTSC decoder needs to be modified.

After completing the modifications use +12V DC as the Vcc supply for the NTSC decoder.

The following parts will be required for the modification:

- 1 x 2N3904 transistor
- 1 x 3.9K resistor
- 1 x 4.7K resistor
- 1 x 3.3K resistor
- 50mm of wire
- 50mm of 3.2mm heat shrink tubing

## Procedure:

Note: All references are to the solder side of the PCB unless otherwise noted. Read the entire step before proceeding with it.

Pinouts of the 2N3904 transistor: while looking at the transistor with the ID lettering right side up (legs facing downwards) the left leg is the Emitter, the middle leg is the Base, and the right leg is the Collector.

Take care when locating components on the PCB, as the part IDs are close to adjacent parts and it would be easy to incorrectly identify a part.

Step 1. Cut the middle leg of the 2N3904 transistor leaving approximately 5mm (1/4"). Do the same with one end of the 4.7K resistor.

Step 2. Solder the short end of the 4.7K resistor to the Base (middle leg) of the 2N3904 transistor. It is easy to do if you first tin both the transistor and resistor ends with solder, then put both ends together and heat.

- Step 3. Cover the middle leg of the transistor and resistor with heat shrink tubing to about 5mm (1/4") covering the other lead of the resistor. Heat the heat shrink tube.
- Step 4. Locate R256. There will either be a resistor or a wire link installed. If there is a resistor installed, remove it and install a wire link.
- Step 5. Locate Q204 (2N3906). Near Q204 is a black wire (on the parts side) which is soldered onto the PCB, between R224 and R223. On the solder side of the PCB you will notice that there is a track going from the solder pad, where the black wire is soldered, to R224. Cut the track between the black wire's solder pad and R224.
- Step 6. Position the 2N3904 transistor under the BNC connector (on the solder side of the PCB) so that the flat side of the case is facing the PCB and the leads are pointing towards P204. Solder the Emitter (lead closest to the edge of the PCB) of the 2N3904 transistor to the end of R224 closest to the edge of the PCB (GND).
- Step 7. Solder the end of the resistor, which is connected to the base of the 2N3904 transistor, to the end of R224 closest to Q204.
- Step 8. Solder a wire from the end of R224 closest to Q204 (or to the collector of Q204, as they are connected by a short track) to the blank solder pad on the track located between pin 5 of P204 and R222.
- Step 9. Solder the Collector of the 2N3904 transistor (remaining leg) to the solder pad where the black wire (on the parts side) is soldered to the PCB.
- Step 10. Solder one end of the 3.9K resistor to the collector of the 2N3904 transistor (same location as in Step 9) and the other end of the 3.9K resistor to the closest end of R256 (wire link). Before soldering the resistor, cover enough of the resistor with heat shrink tubing so that it will not short with any of the adjacent connections.
- Step 11. Remove the capacitor from the black wire that goes to the end of the PCB near P203. Install the black wire directly to the PCB. Note: Only the newer NTSC decoders will have the capacitor installed.
- Step 12. Solder one end of a 3.3K resistor to pin 1 of P203 and the other end to ground. Ground is connected to Pin 3 of P203.

After completing the above steps, recheck the modifications to ensure that they have been done properly. Make sure that the track mentioned in step 5 is cut properly.

The NTSC decoder is now ready to be used in a JAMMA cabinet.

If the picture's colour is not stable or completely absent then the APC control will have to be adjusted. The luminance bias adjustment may also need adjusting, if the APC was not incorrectly adjusted in the first place.

#### PCB adjustments:

The Tint, Color, Contrast, and Sharpness adjustments are straight forward to make when using a color adjustment pattern. All laser disc games have the pattern in their self test options.

Do not adjust the APC or Lum(inance) Bias adjustments. If you suspect that one or both require adjustment then follow the following procedure.

#### APC Adjustment:

1. Apply a color bar signal to the Video Input terminal.
2. Turn the Color control fully clockwise and position the Tint control at its mechanical center.
3. Connect a jumper between TP201 and ground.
4. Turn the APC adjustment control until the color bars on the screen are synchronized.
5. Remove the jumper from TP201 and ground.

#### Luminance Bias Adjustment:

1. Connect an oscilloscope to one of the three video outputs.
2. Turn the Lum. Bias adjustment fully counter-clockwise.
3. Slowly turn the Lum Bias adjustment in the clockwise direction until the black level is just at the sync tips. The sync appears fully compressed.

## Appendix B: Port Map

Address	Direction	Operation/Notes
\$39	read	Unknown
\$44	write	TMS9128NL
\$45	read	TMS9128NL
\$46	write	TMS9128NL ports (sound, overlay blanking)
\$50	read	LD frame low byte (??? Check: whole byte?)
\$51	read	LD frame middle byte
\$52	read	LD frame high byte
\$53	read	Unknown
\$54	write	Same as \$44
\$55	read	Unknown, TMS read?
\$57	write	Latches current frame # from decoder (?)
\$60	write	Switch bank select
\$62	read	Switches
\$64	write	LDP control signal
\$66	write	LDP control signal
\$6A	write	LDP control signal
\$6E	write	IR LED on
\$6F	write	IR LED off

## Appendix C: TMS9128 musings

Reference: <http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF-8&selm=81vof9%246ah%241%40news3.infoave.net>

Here's an excerpt from the "TMS9118/TMS9128/TMS9129 Data Manual" for the VDP pinouts.

The 9118/9128/9129 VDP's are the successors of the 9918A/9928A/9929A VDP's. These new chips have improved memory addressing circuitry which allows the interface of either 8 TMS4116 (2k) or 2 TMS4416 (8K) dynamic RAMs.

		VDP PROCESSOR TYPE			
		9118	9128/29	9918A	9928A/29A
RAS	=   1 U 40	= XTAL1			
CAS	=   2 39	= XTAL2	=====	=====	=====
AD7	=   3 38	= .....	CPUCLK	R-Y	CPUCLK R-Y
AD6	=   4 37	= .....	NC	CPUCLK	GROMCLK GROMCLK
AD5	=   5 36	= .....	COMVID	Y	COMVID Y
AD4	=   6 35	= .....	EXTVDP	B-Y	EXTVDP B-Y
AD3	=   7 34	= RESET/SYNC			
AD2	=   8 33	= vCC			
AD1	=   9 32	= RD0			
AD0	=   10 31	= RD1			
R/W	=   11 30	= RD2			
vSS	=   12 29	= RD3			
MODE	=   13 28	= RD4			
CSW	=   14 27	= RD5			
CSR	=   15 26	= RD6			
INT	=   16 25	= RD7			
CD7	=   17 24	= CD0			
CD6	=   18 23	= CD1			
CD5	=   19 22	= CD2			
CD4	=   20 21	= CD3			

SIGNATURE	PIN	I/O	DESCRIPTION	SIGNATURE	PIN	I/O	DESCRIPTION	
AD0	MSB	10	O	VRAM address bus	CD0	MSB	24	I/O CPU data bus
AD1		9	O		CD1		23	I/O
AD2		8	O		CD2		22	I/O
AD3		7	O		CD3		21	I/O
AD4		6	O		CD4		20	I/O
AD5		5	O		CD5		19	I/O
AD6		4	O		CD6		18	I/O
AD7	LSB	3	O		CD7	LSB	17	I/O

SIGNATURE	PIN	I/O	DESCRIPTION	SIGNATURE	PIN	I/O	DESCRIPTION	
RD0	MSB	32	O	VRAM read/write	CD0	MSB	24	I/O CPU data bus
RD1		31	O	data bus	CD1		23	I/O
RD2		30	O		CD2		22	I/O
RD3		29	O		CD3		21	I/O
RD4		28	O		CD4		20	I/O
RD5		27	O		CD5		19	I/O
RD6		26	O		CD6		18	I/O
RD7	LSB	25	O		CD7	LSB	17	I/O

SIGNATURE	PIN	I/O	DESCRIPTION
CAS	2	O	VRAM column address strobe
CSR	15	I	CPU-VDP read strobe
CSW	14	I	CPU-VDP write strobe
INT	16	O	CPU interrupt output
MODE	13	I	CPU interface mode select; an address line or select
RAS	1	O	VRAM row address strobe
RESET/SYNC	34	I	Trilevel input; below 0.8v - initializes the VDP above 9v sync - sync input for EXTVDP
R/W	11	O	VRAM write strobe
vCC	33	I	+5v supply
vSS	12	I	Ground reference
XTAL1	40	I	10.738635 MHz crystal connection
XTAL2	39	I	10.738635 MHz crystal connection

#### 9918/28/29 Differing Pins

SIGNATURE	PIN	I/O	DESCRIPTION
EXTVDP	35	I	Multiple TMS9118 VDP operation
B-Y	35	O	Blue color difference output
COMVID	36	O	Composite Video Output
Y	36	O	Y (Black/White luminance and composite sync) output
NC	37		Reserved - Do not use
CPUCLK	37	O	CPUCLK equals XTAL/3 (color burst freq)
CPUCLK	38	O	CPUCLK equals XTAL/3 (color burst freq) Can be used as an external clock
R-Y	38	O	Red color difference output

## Appendix D: Relevant Screen Captures



First boot screen. Self test mode where the graphics will be in various patterns. (If you're even getting this far, it's a good sign. You have a lot of working components.)



After the self test section, the player will be sent commands to start playing. If you got to the first step (above), you'll probably get to this part.



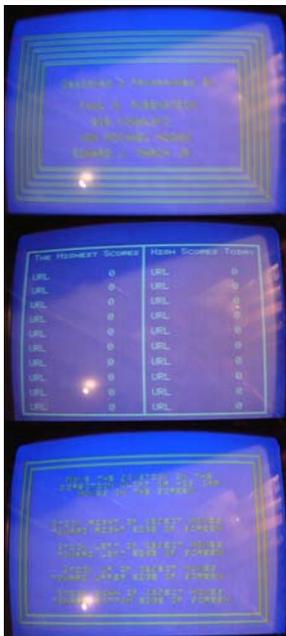
If your game is working properly, you'll see computer generated graphics on the screen ("Cliff Hanger"). This is also the first frame in the attract sequence.



Next part of attract mode. Still computer graphics.



If everything is working, you'll probably see laser player images finally. The attract sequence is just a couple of minutes or so. You may or may not have sound (I believe this is a DIP setting).



After the video sequence, you'll see a few short series of PCB generate graphics. This is not comprehensive, but gives you an idea.

- Programmers
- High Score Table
- Instructions

Then the game goes back to the Cliff Hanger attract screen. If you don't have your screen adjusted properly, you may see text (rolling) on the screen that says something like "Searching Frame 0". The game is supposed to be showing the PC graphics and not showing that laser player generated image.



This does NOT mean that your player is bad! (Although it could mean that). Be sure that you "Y" cable is connected correctly. Also, if you can verify your player is working (see the NTSC section for plugging the image directly into the monitor), then you may have an issue on the GSI board.



Starting the game. Notice that there are two areas where PC graphics are overlaid. (Score and number of Players.)

Turning on and off the graphics overlay is another DIP switch setting. If you're not seeing those graphics, check the DIPs



Also playing the game



When an “Action” or “Stick” movement is required, a hint appears at the bottom of the screen. It’s a PC graphic overlay. I believe this is also DIP configurable.



You’ll get the “correct” move (configurable).



Then your score, and then you’ll pick up near where you left off. Usually near a start point where you died.



You may get the buy-in/continue screen.