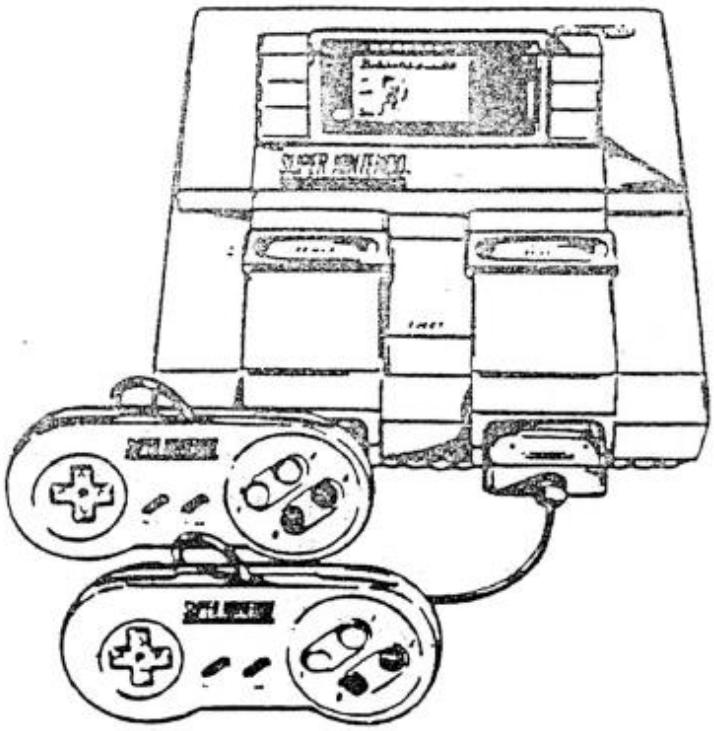
SUPER NINTENDO.
ENTERTAINMENT SYSTEM



INDEX

GENERAL OVERVIEW

Console

The Super NES Console uses:
• a Central Processing Unit (CPU),
• a two-IC Image Processing Unit (PPU1 & PPU2),
• and two sound microprocessors.

1. CPU – Central Processing Unit

- A specially-designed processor for Nintendo
- A 24-bit address bus that allows the CPU to access 16M of ROM/RAM simultaneously
- Internal 16-bit registers
- 8-bit system data bus
- Contains Direct Memory Access (DMA) operations

2. PPU1 / PPU2 – Image Processing Units 1 and 2

- Specially-designed processors (two ICs) for Nintendo
- Up to **128 sprites**, each with its own priority level, may appear on screen; **32 sprites per scanline**
- Sprites may be one of four sizes: **8×8, 16×16, 32×32, and 64×64 pixels**
- **32,768 colors available**
- Eight color palettes available, each containing 16 colors per screen; each sprite contains 16 colors from one palette
- Supports horizontal, vertical, and diagonal scrolling; partial vertical scrolling also possible
- Resolution options: **256×224**, **512×224**, or **512×448** pixels
- Special effects include mosaic, window, fixed-color addition/subtraction, full-screen addition/subtraction, and pseudo-512 horizontal resolution
- 16-bit Data Bus

3. Sound Microprocessors

- Eight channels of Pulse Code Modulation (PCM) sound
- Stereo digital audio

The Super NES is supplied with an AC adapter that converts **120 V AC to 10 V DC (unregulated).**

Super NES Controller Devices

The controller devices allow the player to interface with the game (i.e., they control the actions within the game).
There are twelve input function buttons on the Controller.

The CPU sends control signals to load the Controller's function data into two 8-bit parallel/serial registers, which then shift the data out serially.
This process occurs at software-controlled intervals, approximately every **17 milliseconds**.

Super NES Super Scope

The Super NES Super Scope is a peripheral shooting device made up of two units:
• a transmitter assembly and
• a receiver assembly.

The Super Scope transmitter detects the CRT scanner timing to determine the aim location.
This information is then transmitted, via the receiver unit, to the Console.

Super NES Game Cartridge

Game Cartridges contain semiconductor memory Integrated Circuits (ICs).
Current Game Cartridges contain up to two memory devices in combinations of:

- ROM (Read-Only Memory)
- RAM (Random Access Memory)

The various memories and their functions are listed below:

Program ROM

Contains:

- game program code
- character data

The game program instructs the CPU to send control signals to the sound microprocessor, check controller input, transfer PPU parameters, calculate scores and in-game timing, etc.

Work RAM

Stores:

- character information
- game position data
- also provides extra memory space for use by the Console CPU

In some Game Cartridges, a low-power Work RAM is connected to a battery to retain game information after the Cartridge is removed.

74LS139

Used for chip selection within the Game Cartridge:
selects between Program ROM, Work RAM, and the sound processor (when used).

A special Integrated Circuit (CIC) is also included in Game Cartridges.
The CIC communicates with the corresponding CIC in the Console to prevent unauthorized software from operating on the Console.

OPERATION OF THE SUPER NES CONSOLE

Purpose

The Super NES Console contains the essential system hardware.
Its purpose is to display video graphics stored in firmware (information programmed and stored in ROM) and to allow the player to interact with the game.

Player control is performed through input read by the Console CPU via the controller device.

The Super NES can connect to either a television or a monitor.
The Multi-Out port can provide:

- Stereo A/V
- RGB video data
- S-VHS (S-Video) output

A separate RF output allows RF connection to a TV using an RF Switch Box (mono sound only).

The Console also includes a **28-pin expansion port** for external peripherals.

Theory of Operation

The Console cannot operate alone.
It requires:

- the information programmed in the Game Cartridge ROM
- the user input from the controller

The Console's primary functions include:

- executing instructions stored in ROM
- producing sound
- generating on-screen displays through image processing
- scrolling backgrounds
- moving characters based on controller input

All activity is controlled by the **CPU**.

The Console can be divided into **eight functional blocks**:

1. Power Supply Section
2. Controller Input Section
3. System Timing Section
4. RESET Section
5. RESET IC Section
6. Sound Section
7. Display Section
8. Program Section

Below is an overview of each block.

---

1. Power Supply Section

The AC Adapter provides **10 V DC unregulated** to the Console.

The power section on the main PCB includes:

- noise filter
- reverse-current protection
- power switch
- surge suppressor
- ripple filter
- voltage regulator

Outputs:

- **5 V regulated DC (Vcc)**
- **7 V unregulated DC (Vs)** used by the sound amplifier

---

2. Controller Input Section

Includes diode arrays and capacitors for voltage-spike protection and pull-up resistors for logic levels.

The CPU reads controller input approximately every **17 ms**, though this rate may be altered by software.

---

3. System Timing Section

Contains:

- **21 MHz oscillator**
- transistors
- capacitors
- resistors

This circuit produces a **21 MHz master clock** for:

- CPU
- PPU1 & PPU2
- Connector P1

The CPU divides the master clock by:

- 6 → **3.58 MHz**
- 8 → **2.68 MHz**
- 12 → **1.79 MHz**

CPU internal operation frequency: **3.58 MHz**
Address mapping transfers operate at **2.68 MHz**.
Clock speed may change automatically based on the accessed device (RAM, ROM, LSI).
DMA operations always run at **2.68 MHz**.

---

4. RESET Section

Contains:

- 4 MHz oscillator
- Hex inverter IC 74HCU04
- CIC security IC
- Reset switch
- discrete resistors/capacitors

The Console CIC communicates with the Cartridge CIC.

To ensure proper processor operation, **both PPUs must initialize before the CPU**.

Therefore, CIC applies RESET to PPU2 first, introducing a delay, then PPU1, then CPU.

---

5. RESET IC Section

The RESET IC (voltage detector) ensures that during power interruptions or low-voltage conditions, CPU operation is held to prevent malfunction or improper writes to Cartridge Work RAM.

---

6. Sound Section

Includes:

- Sound Module
- Sound Amplifier

The Sound Module contains:

- **SMP** sound microprocessor
- **DSP** digital sound processor
- **D/A converter**
- two 256K RAM ICs

The SMP handles sound processing and includes I/O and monitor programs.

Program and tone data are loaded into RAM from Cartridge Program ROM under SMP control.

RAM is time-shared between SMP and DSP.

The DSP includes:

- 8 PCM channels
- noise generator
- echo
- sweep
- envelope generators
- various tone-control circuits

The sound amplifier receives input from:

- Expansion connector (P6)

- 62-pin connector (P1)
- Sound module

Amplifier uses LM358 operational amplifiers for left/right channels.

Outputs:

- stereo via Multi-Out (P4)
- mixed mono for RF modulator and expansion port

---

7. Display Section

Includes:

- PPU (PPU1 + PPU2)
- VRAM (two 256K SRAMs)
- RGB amplifier circuit
- RGB encoder
- sync/video output amplifiers

PPU1 + PPU2 operate together as a single display processor, but with distinct roles:

PPU1 handles:

- generation
- rotation
- scaling
- background address control

PPU2 handles:

- color
- special effects (windows, mosaic, color math)
- generating RGB image signals

RGB signals go to the RGB Encoder, which outputs composite video and S-Video.

---

8. Program Section

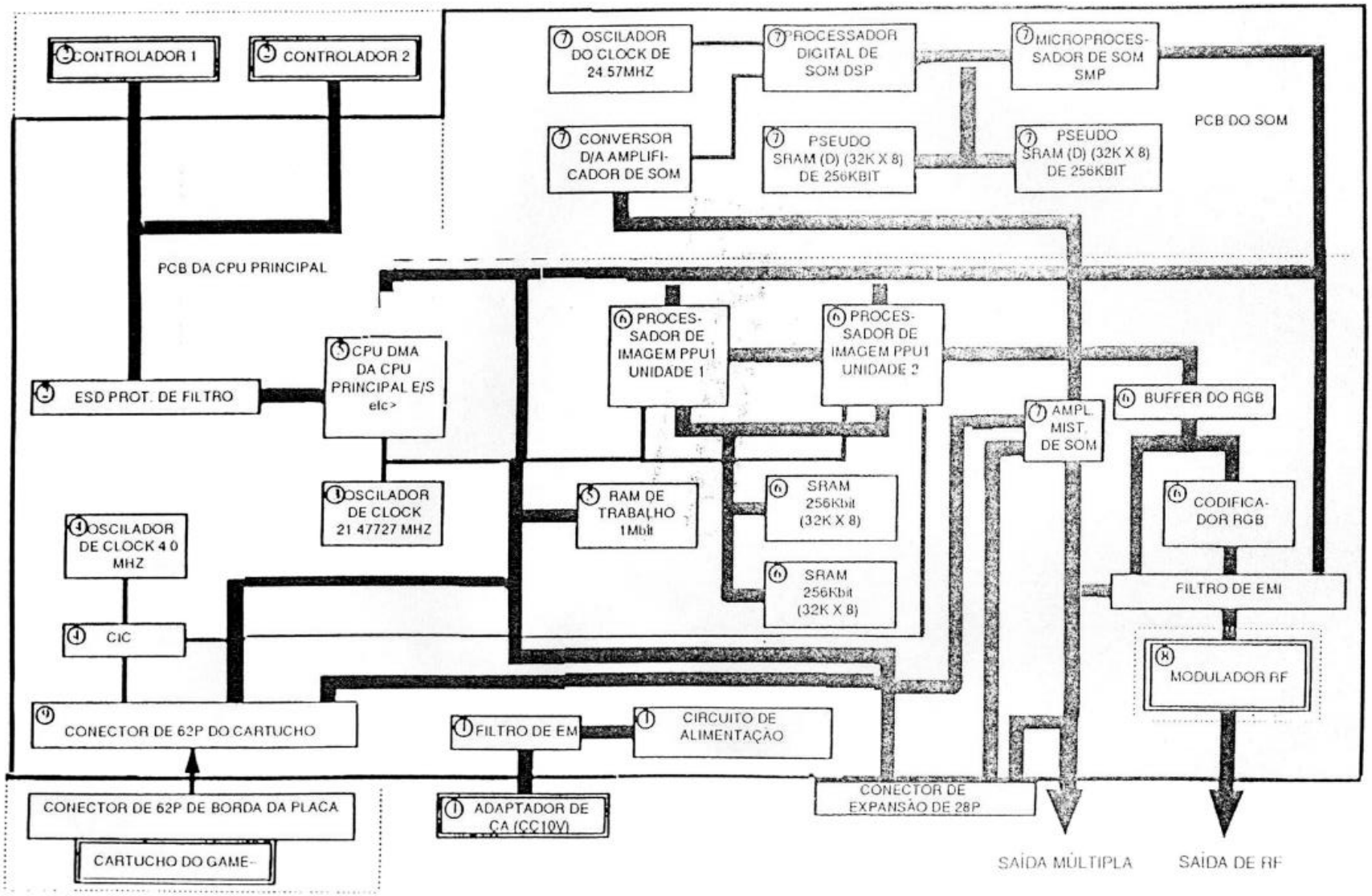
Includes:

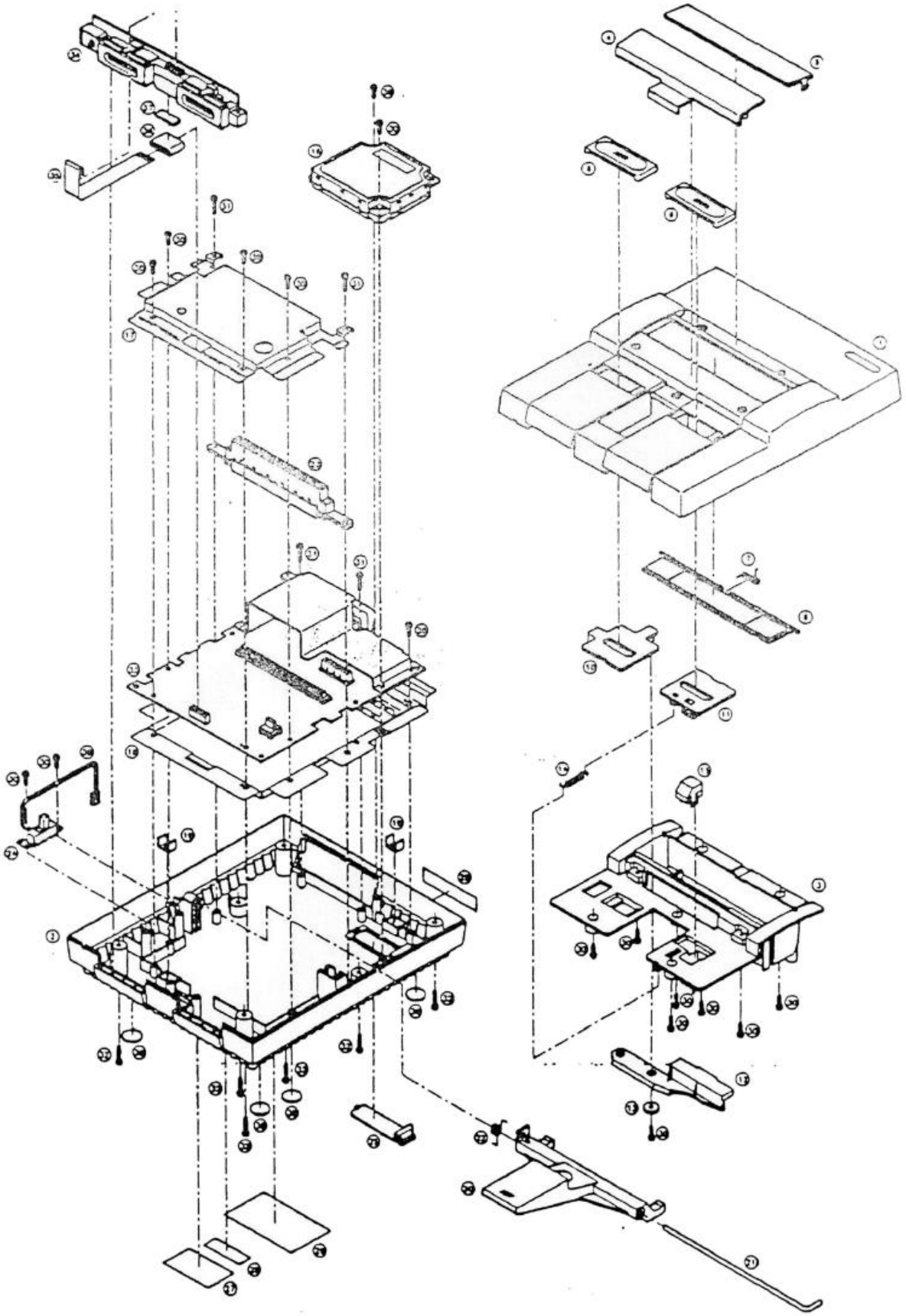
- CPU

- Work RAM

CPU responsibilities:

- Fetch/decode ROM instructions
- Provide timing/control signals to Console components
- Control data transfers across ICs (incl. DMA)
- Perform arithmetic and logic functions
- Respond to peripheral control signals (e.g., RESET)


CONTROLADOR 1
CONTROLADOR 2
OSCILADOR DO CLOCK DE 24 57MHZ
PROCESSADOR DIGITAL DE SOM DSP
MICROPROCES-SADOR DE SOM SMP
PCB DO SOM
CONVERSOR D/A AMPLIFI-CADOR DE SOM
PSEUDO SRAM (D) (32K X 8) DE 256KBIT
PSEUDO SRAM (D) (32K X 8) DE 256KBIT
PCB DA CPU PRINCIPAL
PROCES-SADOR DE IMAGEM PPU1 UNIDADE 1
PROCES-SADOR DE IMAGEM PPU1 UNIDADE 2
CPU DMA DA CPU PRINCIPAL E/S etc>
ESD PROT. DE FILTRO
BUFFER DO RGB
AMPL. MIST. DE SOM
OSCILADOR DE CLOCK 21 47727 MHZ
RAM DE TRABALHO 1Mbit
SRAM 256Kbit (32K X 8)
CODIFICA-DOR RGB
OSCILADOR DE CLOCK 4 0 MHZ
SRAM 256Kbit (32K X 8)
FILTRO DE EMI
CIC
MODULADOR RF
CONECTOR DE 62P DO CARTUCHO
FILTRO DE EM
CIRCUITO DE ALIMENTAÇÃO
CONECTOR DE 62P DE BORDA DA PLACA
ADAPTADOR DE CA (CC10V)
CONECTOR DE EXPANSÃO DE 28P
CARTUCHO DO GAME-
SAÍDA MÚLTIPLA
SAÍDA DE RF



PARTS LIST

ITEM	CODE	DESCRIPTION	QTY
38	340.001.1	Rubber Foot	4
16	640.003.1	Sound Module	1
36	123.001.1	Ferrite Ring	1
17	321.001.1	Upper Shield	1
18	321.004.1	Lower Shield	1
23	240.001.1	Upper 62-pin Connector	1
1	330.001.1	Upper Case	1
2	330.005.1	Lower Case	1
—	240.012.1	12-pin Connector	1
—	240.013.1	28-pin Connector	1
—	240.018.1	21-pin FFC-S Connector	1
—	153.002.1	CPU IC RF5A22	1
—	153.005.1	PPU1 IC RF5C77	1
—	153.006.1	PPU2 IC RF5C78	1
—	152.004.1	WRAM 1M IC	1
—	154.001.1	BA6592F RGB IC (SOP)	1
—	170.001.1	1.5A Fuse (Axial)	1
—	135.002.1	DAN202 Diode Matrix	5
—	135.003.1	DAP202 Diode Matrix	5
—	140.001.1	2SA1037 Transistor	6
—	140.003.1	2SC2412 Transistor	10
—	140.002.1	2SC4699/NP Transistor	2
—	240.003.1	Lower 62-pin Connector	1

ITEM	CODE	DESCRIPTION	QTY
—	155.001.1	F411 CIC IC	1
—	124.001.1	ZJYS5012-2PT Line Filter	1
—	161.002.1	4 MHz Ceramic Oscillator	1
—	160.002.1	21.453666 MHz Crystal (PAL-M)	1
—	150.001.1	74HCU04 IC (SOP)	1
—	154.002.1	LM358 IC (SOP)	1
—	152.003.1	256K SRAM 28-pin (SOP)	2
—	156.001.1	7805 Regulator	1
—	320.001.1	Heatsink	1
—	301.001.1	Insulating Washer	1
—	300.016.1	3×10 Screw	1
—	300.017.1	3×7 Screw	5
—	641.001.1	SNES RF Modulator	1
—	240.011.1	Power Jack	1
—	260.006.1	Reset Switch	1
—	132.003.1	SK-RB110C Diode	1
—	155.002.1	PST529DMT Reset IC	1
—	105.002.1	ERZ-CF2MK220 Varistor	1
—	120.002.1	22 µH Inductor	1
—	110.003.1	0.1 µF / 25V Ceramic Capacitor	6
—	110.001.1	0.1 µF / 50V Ceramic Capacitor	15

ITEM CODE DESCRIPTION QTY

110.002.1 0.047 µF / 25V Ceramic Capacitor 1

110.006.1 10 pF / 50V Ceramic Capacitor 4

110.013.1 47 pF / 50V Ceramic Capacitor 7

110.014.1 56 pF / 50V Ceramic Capacitor 3
110.015.1 82 pF / 50V Ceramic Capacitor 4
110.010.1 220 pF / 50V Ceramic Capacitor 2
110.011.1 300 pF / 50V Ceramic Capacitor 13
110.004.1 1000 pF / 50V Ceramic Capacitor 1
110.009.1 1 µF / 16V Ceramic Capacitor 16
112.003.1 10 µF / 16V Electrolytic Capacitor 3
112.005.1 33 µF / 25V Electrolytic Capacitor 2
112.004.1 2.2 µF / 25V Electrolytic Capacitor 1
100.018.1 39 Ω Resistor (Chip, 1/10 W) 3
100.020.1 56 Ω Resistor (Chip, 1/10 W) 1
100.011.1 100 Ω Resistor (Chip, 1/10 W) 13
100.012.1 120 Ω Resistor (Chip, 1/10 W) 3
100.013.1 130 Ω Resistor (Chip, 1/10 W) 3
100.014.1 180 Ω Resistor (Chip, 1/10 W) 3
100.015.1 200 Ω Resistor (Chip, 1/10 W) 6
100.016.1 270 Ω Resistor (Chip, 1/10 W) 3
100.017.1 300 Ω Resistor (Chip, 1/10 W) 4
100.019.1 470 Ω Resistor (Chip, 1/10 W) 3
100.023.1 560 Ω Resistor (Chip, 1/10 W) 1
100.007.1 1 kΩ Resistor (Chip, 1/10 W) 9
100.003.1 1.2 kΩ Resistor 2
100.004.1 1.8 kΩ Resistor 5
100.009.1 3.3 kΩ Resistor 2
100.006.1 10 kΩ Resistor 25
100.008.1 24 kΩ Resistor 3
100.010.1 33 kΩ Resistor 2
100.005.1 100 kΩ Resistor 1
300.018.1 M3×8 Screw w/Washer 1

Parts List — Continued

ITEM	CODE	DESCRIPTION	QTY
100.021.1		150 kΩ Resistor	1

ITEM	CODE	DESCRIPTION	QTY
100.022.1	330 kΩ Resistor	1	
100.001.1	1 MΩ Resistor	1	
112.002.1	100 µF / 10 V Electrolytic Capacitor	1	
112.001.1	220 µF / 10 V Electrolytic Capacitor	1	
161.001.1	3.58 MHz Crystal	1	
150.002.1	74LS139 IC (SOP)	1	
150.003.1	BA6592F RGB Encoder	1	
150.004.1	BA5222F IC (SOP)	1	
150.005.1	BA6594F IC (SOP)	1	
154.003.1	AN7522N Audio IC	1	
120.004.1	4.7 mH Inductor	1	
121.001.1	47 µH Coil	1	
121.002.1	390 µH Coil	1	
110.012.1	22 pF Ceramic Capacitor	2	
131.001.1	1SS133 Diode	2	
131.002.1	MA165 Diode	1	
132.002.1	RB152 Diode	2	
140.004.1	2SC1740 Transistor	1	
140.005.1	2SA933 Transistor	1	
100.002.1	220 Ω (1/4 W) Resistor	2	
—	300.020.1	M3×6 Screw	1
—	300.021.1	M3×10 Screw w/Washer	1
—	300.022.1	3×8 Tapping Screw	1

Top View Diagram (Description)

The diagram shows the internal layout of the Super NES, with circled numbers linking each part to the parts list. The following components appear:

1. **Upper Case**
2. **Lower Case**
3. **Power Jack**
4. **Reset Switch**
5. **RF Modulator**
6. **Multi-Out Connector**
7. **62-Pin Cartridge Slot (Upper + Lower Connectors)**
8. **Main PCB**
9. **Shielding Plate (Upper)**
10. **Shielding Plate (Lower)**
11. **Sound Module**
12. **CPU (RF5A22)**
13. **PPU1 (RF5C77)**
14. **PPU2 (RF5C78)**
15. **WRAM (1 Mbit SRAM)**
16. **VRAM (Two 256 Kbit SRAMs)**
17. **RGB Encoder IC**
18. **Reset IC**
19. **Line Filter**
20. **7805 Voltage Regulator with Heatsink**

The board drawing includes connector labels:

- **P1** – 62-pin game connector
- **P2** – Power connector
- **P3** – RF output
- **P4** – Multi-Out A/V connector
- **P5** – Reset switch
- **P6** – Audio expansion connector
- **P7** – Controller ports (on a separate sub-board)

Diagram Notes (Translated)

Symbols Used

- Ground is shown with the standard symbol.
- Direct connections between components are shown with solid lines.
- Shield plates are drawn with cross-hatched shading.

- Signal paths (audio, video, control) are labeled accordingly.

Important Functional Blocks (as shown on diagram)

- **Power Block**
  Showing regulator, filter, fuse, and diode protection circuits.
- **CPU Block**
  The RF5A22 processor and its connections to WRAM, ROM (via cartridge), and CIC.
- **PPU Block**
  PPU1 + PPU2 with VRAM and the RGB output section.
- **Sound Block**
  Sound Module, sound RAM, DSP, amplifier, and output paths.
- **Security (CIC) Block**
  Showing handshake lines to the cartridge CIC.

Super NES — Functional Block Diagram (Textual Representation)

1. Power Supply Block

- AC Adapter → Noise Filter → Reverse-Current Protection → Power Switch → Surge Absorber → Ripple Filter
- Output Lines:
  - **+5 V (Vcc)** — regulated
  - **+7 V (Vs)** — unregulated sound supply

---

2. Reset / Security Block

Contains:

- **CIC (Checking Integrated Circuit)**
- **Reset Switch**
- **Reset IC (Voltage monitor)**
- 4 MHz Oscillator
- Hex inverter (74HCU04)

Functions:

- Synchronizes boot-up sequence between Console and Cartridge
- Applies delayed reset in proper order:
  1. PPU2
  2. PPU1

3. CPU
- Holds system in RESET on power dips

---

3. CPU Block

RF5A22 Microprocessor

Connected to:

- **WRAM** (1 Mbit)
- **PPU1 / PPU2**
- **Sound Module** (via APU I/O bus)
- **Cartridge ROM / RAM** (via 62-pin connector)
- DMA controller
- Interrupt lines

Clock input: **3.58 MHz** (divided from 21 MHz master)

---

4. PPU Block

Contains:

- **PPU1 (RF5C77)** – background, rotation/scaling, address generation
- **PPU2 (RF5C78)** – color math, windows, blending, RGB output

Connected to:

- **VRAM** (256 K × 2)
- **RGB Encoder** (BA6592F)
- Sync generator
- Video amplifiers

Outputs:

- RGB
- Composite video
- S-Video (Y/C)

---

5. Sound Block

Contains:

- **SMP (Sound Microprocessor)**
- **DSP (Digital Sound Processor)**
- **D/A Converter**
- **256K × 2 Sound RAM**
- **Audio Low-Pass Filters**
- **LM358 Audio Amplifier**

Outputs:

- Stereo (via Multi-Out)
- Mixed Mono (to RF modulator)

---

6. Controller Input Block

- Serial shift-register based input
- Two controller ports
- Pull-up resistor arrays
- Protection diodes

CPU polls controllers every **~17 ms**, unless modified by software.

---

7. Cartridge Slot Block

62-pin connector includes:

- Address/Data Bus
- ROM / RAM lines
- CIC handshake
- Audio expansion pins
- Decoder signals
- IRQ / DMA lines

---

8. System Timing Block

Contains:

- **21.47727 MHz Crystal**
- Divider circuits in CPU

- Timing distribution to:
    - CPU
    - PPU1
    - PPU2
    - APU
    - Connectors

Diagram Notes (Translated)

**“The diagram illustrates how all console sections interact.
Signal paths for audio, video, CPU buses, reset control, timing, and cartridge interface are clearly shown.”**

Super NES Console — Top Case Replacement

This page describes **disassembly** and **reassembly** procedures for the console’s exterior plastic case.

Removing the Upper Case

1. Remove the **six (6)** screws located on the bottom of the console.
    - These screws are security-type “Gamebit” fasteners.
2. Lift the **upper case** vertically.
    - Be careful not to bend or stress the Reset switch linkage.
3. The Reset switch button is held in place by a guiding rail.
    - Remove gently if necessary.

Removing the Lower Case Components

Once the upper case is removed:

- The **main PCB**, **RF module**, **heat sink**, and **shielding plates** become accessible.
- DO NOT touch the VRAM, WRAM, or CPU/PPU pins unless properly grounded.
- Avoid applying force to the Multi-Out or Cartridge connectors.

Inspection Notes

When inspecting the console:

- Check for **broken standoffs** on the plastic housing.
- Verify that the **heat sink** is firmly attached to the voltage regulator (7805).
- Confirm that the **RF modulator shield** is intact.

All components must be in place before reassembly.

Reassembling the Super NES Console Case

Follow the reverse order of disassembly.
Ensure all components are properly seated before closing the case.

1. Check the Shielding Plates

- The **upper** and **lower** RF shields must be properly aligned with their mounting points.
- Ensure no wires or components are trapped between the shields and the PCB.

2. Check Reset Mechanism

Before placing the upper case:

- Verify that the **Reset switch rod** is aligned with the Reset button channel on the upper shell.
- Press the Reset button several times to confirm smooth movement.

3. Insert the Upper Case

- Lower the upper case vertically.
- Make sure the case fits without force—misalignment may indicate obstruction by shielding plates or the heat sink.

4. Install the Six Security Screws

- Use the appropriate **4.5 mm Gamebit tool**.
- Tighten screws evenly to avoid stress on the case.

Inspection After Assembly

Once the console is fully reassembled:

- Check that the **Power switch** slides correctly.
- Check that the **Reset button** freely moves.
- Confirm that the **cartridge slot** is unobstructed and centered.
- Inspect the **Multi-Out** port alignment for proper cable insertion.
- Verify no internal rattling is heard when gently shaking the console.

Final Test Procedure

After reassembly, perform a simple functional test:

1. Connect AC adapter and video cable.
2. Power on the console without a cartridge.
    - The screen should remain black, no noise.
3. Power off, insert a known-good cartridge.
4. Power on again.
    - Confirm **video**, **audio**, and **controller inputs**.

If any symptom appears (no video, corrupted graphics, static sound, no controller input), proceed to the troubleshooting pages later in the manual.

Console External Features (Description Page)

This page originally contains labeled drawings of the exterior of the Super Nintendo console. Below is the translated and structured content.

Front Panel

- **POWER Switch**
  Slide button used to turn the console ON and OFF.
  When ON, the cartridge eject mechanism is locked.
- **EJECT Button**
  Raises the cartridge from the slot using a mechanical lever system.
- **RESET Button**
  Momentarily resets the CPU, restarting the game program.
- **Controller Ports (1 & 2)**
  Two 7-pin connectors for standard Super NES controllers.
  Each port supports serial communication at CPU-controlled intervals.

Top Panel

- **Cartridge Slot**
  62-pin connector accepting SNES game cartridges.
  Includes CIC communication lines, addressing bus, audio expansion pins.
- **Cartridge Dust Cover**
  Spring-loaded flap protecting the connector from dust.

Rear Panel

- **AC Adapter Input**
  Accepts 10V DC from the external power supply.
- **MULTI-OUT Connector**
  Provides:
  - Composite video
  - S-Video (Y/C)
  - RGB (depending on region)
  - Stereo audio
- **RF OUT**
  Output for RF Switch Box (mono sound).
  Used for televisions without composite/S-Video inputs.
- **Cooling Vents**
  Plastic louvers for airflow and internal heat dissipation.

Bottom Panel

- **Expansion Port**
  28-pin connector for future peripherals (unreleased in most regions).
- **Security Screws**
  Six screws used to secure the top and bottom plastic shells.
- **Serial Number Label**
  Indicates manufacturing batch and region.

CONSOLE TEST PROCEDURE

Required Equipment:

- Stereo television monitor
- Super NES AC Adapter
- Super NES RF Switch
- Super NES Stereo A/V Cables
- Two (2) Super NES Controllers
- Super Mario World Game Cartridge
- Super NES Console Test Device

Test Procedure:

Before inserting the game cartridge into the Console, inspect the **upper 62-pin connector** for contamination or damage.

1. Connect the AC adapter, RF switch, and Stereo A/V cables to the console.
2. Connect controllers to ports **1** and **2**, and insert the **Super Mario World** cartridge.
3. Power on the console and set the TV to **RF mode**.
   Select channel **3 or 4** for a clear display.
4. Using Controller #1, select **two-player mode** and start the game.
5. Play briefly and verify proper operation:
   - Color
   - Image
   - Sound

   Move the screen 3–4 times (scrolling) to check for visual stability.
   Test both controllers for full and correct functionality.

6. Switch between channels **3 and 4** on the RF modulator to confirm correct picture and audio on both channels.
   While connected, **wiggle the RF connector** to ensure solid contact.
7. Switch the TV mode from RF to **A/V** and wiggle the Multi-Out plug similarly.
   In A/V mode, verify **stereo output**.

NOTE:

Before performing any repairs on the PCB, test the unit again using:

- a known-good **sound module**, and
- a known-good **upper 62-pin connector**.

TROUBLESHOOTING

Super NES Console Test Device

1. Auto Test 1

Auto Test 1 checks system RAM and registers, verifying read/write functions from the CPU and confirming correctness of returned values.

Priority of Component Substitution When Faulty

Tested Item	1st Priority	2nd	3rd	4th
WRAM	CPU	WRAM	—	—
DRAM	CPU	WRAM	—	—
VRAM LOW	PPU1	CPU	SRAM (U5)	PPU2
VRAM HIGH	PPU1	CPU	SRAM (U4)	PPU2
DMA MEMORY	CPU	—	—	—
OAM	PPU1	CPU	—	—
CG RAM	PPU2	CPU	—	—
CPU MULTIPLIER	CPU	—	—	—
PPU MULTIPLIER	PPU1	CPU	—	—
CPU DIVIDER	CPU	—	—	—
DMA	CPU	PPU1	PPU2	—
EXT LATCH	PPU1	CPU	—	—
TIMER	CPU	PPU2	PPU1	—
HV COUNTER	PPU1	CPU	—	—
V224/V239	CPU	PPU2	PPU1	—
FIELD FLAG	PPU2	CPU	PPU1	—
VH FLAG	PPU2	CPU	PPU1	—
OBJECT OVERFLOW	PPU1	CPU	—	—
CPU CLOCK	CPU	—	—	—
CPU	CPU	—	—	—
APU (Sound Module)	CPU	—	—	—

NOTE:
Check this table if errors occur during Auto Test 1.
For multiple errors, judge after reviewing all factors.
Remember: the CPU checks each LSI function through the bus lines.

2. Controller Port Signals

Faulty Item	Test Location	Area to Inspect
RAMSEL	Tests RAMSEL signal of P1	P1 pin 32
REFRESH	Tests REFRESH signal of P1	P1 pin 33
OUT0 (output)	Tests OUT0 signal of port output	P2, R91, DA1, DA5, C43, CPU
4016D0, 4016D1	Input port test for register 4016	P2, R87, R88, DA3, DA8, C39–C40, CPU
4017D0, 4017D1	Input port test for register 4017	P2, R85, R86, DA4, DA6, C37–C38, CPU
PP6, PP7	I/O ports PP6, PP7	P2, R83, R84, DA?, C35–C38
62P PA / 62P CA	Tests address line	P1
Through (pass-through)	Tests line continuity from P1 to P6	P1 pin 2 → P6 pin 24 → R87
IRQ	Tests external interrupt to CPU	P1 pin 18, P6 pin 26, R29, CPU

3. 4 MHz Oscillator Frequency Measurement

Acceptable Range: 3.8 to 4.2 MHz

If the 4 MHz CIC clock frequency is outside the acceptable range, inspect:

- 4 MHz ceramic oscillator (X2)
- HCU04 (U9)
- surrounding peripheral circuitry

Important:
If the clock does NOT oscillate, the CIC will not function properly.
However, the reset output (pin 10) may still go HIGH automatically, in which case the **CPU and PPU will operate normally**.

COLOR TONE DEFECTS

Color tone issues are usually caused by faults **after PPU2**.
The defective component(s) can be narrowed down depending on which output displays the error:

- Composite VIDEO
- RGB
- S-Terminal (S-Video)
- RF output

1. Monochrome Screen

- Check master clock frequency **21.453 MHz**

If adjustment is impossible or unstable, replace in order:

1. Compensation capacitor (TC1)
2. 21.453666 MHz oscillator (X1)

Check video output (RAW VIDEO)

If only RF output is faulty → replace RF modulator.
If BOTH outputs are faulty → inspect discrete components around RGB Encoder (U7) for bad solder or damage.

Replace in order:

1. RGB Encoder (U7)
2. PPU2 (U3)

2. Incorrect Screen Color (reddish, bluish, etc.)

Probable causes:

- Defective PPU2
- Missing/disrupted RED / GREEN / BLUE RGB output signals

Inspect area around RGB amplifier transistors **Q3–Q9** for solder or damage.

If faulty → Replace **PPU2 (U3)**.

Another cause is a defective RGB encoder or color signal filters (C14, C186).
Inspect components around **RGB Encoder (U7)**, then replace **U7** if necessary.

IMAGE DEFECTS

The same problems can appear due to traces/lines on the input or output side.
Take this into account if replacing components does NOT solve the issue.

1. Vertical Lines

If character shapes are normal but vertical lines appear:

Replace in order:
PPU2 → PPU1 → CPU → VRAM

If characters themselves are corrupted AND have vertical lines:

Replace in order:
PPU1 → PPU2 → CPU → VRAM

2. Mosaic Screen / Characters Corrupted

Replace in order:
PPU1 → PPU2 → CPU → VRAM

3. Objects/Character Colors Incorrect

Replace in order:
PPU1 → PPU2 → CPU

4. Screen Flicker / Part of the Screen Not Displayed

Replace in order:
PPU1 → PPU2 → CPU → VRAM

5. Window Screen Position Misaligned

Replace in order:
CPU → PPU2 → PPU1

6. Image Defect Occurs Only With a Specific Game Cartridge

Replace in order:
CPU → PPU2 → PPU1

CONSOLE DIAGNOSTIC TABLE

Before performing any repair on the PCB, disassemble the unit and thoroughly inspect the board.

Look for:

- damaged/missing components
- damaged traces
- bad solder joints
- dirt or corrosion

Do NOT attempt to clean 62-pin connectors — they must be replaced.

Malfunction → Probable Cause (in priority order)

Solid Color Screen

- Defective 62-pin connector
- Defective Sound Module
- Bad solder joint
- Defective CPU (U1)
- Defective PPU2 (U3)
- Defective WRAM (U8)
- Defective PPU1 (U2)
- Defective RGB Encoder (U7)
- Defective 21 MHz Oscillator (X1)
- Defective CIC (U8)

Distorted Video Images

- Defective 62-pin connector
- Bad solder joint
- Defective PPU2 (U3)
- Defective CPU (U1)
- Defective PPU1 (U2)
- Defective VRAM (U4 or U5)

Distorted or Missing Audio

- Defective Sound Module
- Faulty P5 connector
- Defective CPU (U1)
- Defective Audio Amplifier (U10)
- Defective Q18 transistor

No RESET

- Defective CIC (U8)
- Defective 4 MHz Oscillator (X2)
- Defective 74HCU04 (U9)

Weak or Missing Color

- Defective TC1
    - Master clock must be adjusted to **21.453666 MHz**
- Defective PPU2 (U3)
- Defective RGB Encoder (U7)
- Defective 21 MHz Oscillator (X1)

No Power

- Defective 1.5 A Fuse (F1)
- Defective D1
- Defective Line Filter (T1)
- Defective 7805 Regulator (U12)
- Defective Surge Absorber (VA1)
- Defective Q18
- Defective Power Switch (P3)

Snow / Interference on Screen

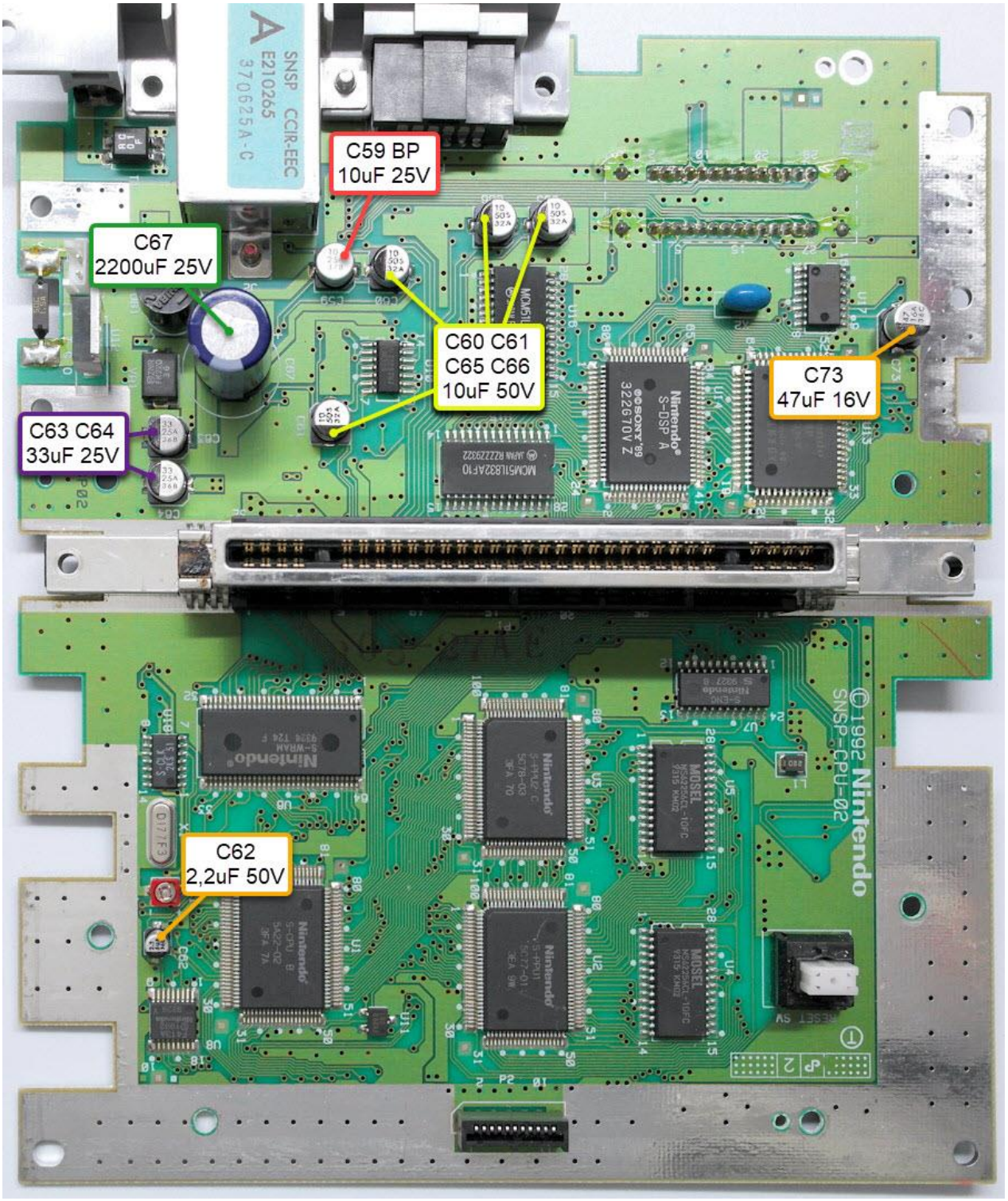
- Defective RF Modulator

Intermittent or No Controller Operation

- Faulty Front-Unit PCB
- Faulty FCC Ribbon Cable
- Faulty P2 Connector
- Defective CPU (U1)

Frozen Screen / Program Operation Unstable

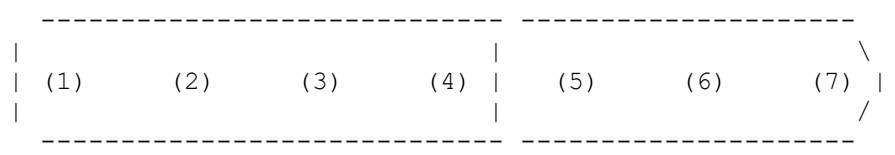
- Defective 62-pin connector
- Bad solder joint
- Defective CPU (U1)
- Defective PPU2 (U3)
- Defective PPU1 (U2)


C59 BP
10uF 25V
C67
2200uF 25V
C60 C61
C65 C66
10uF 50V
C73
47uF 16V
C63 C64
33uF 25V
C62
2,2uF 50V
SNSP CCIR-EEC
E210265
370625A-C
Nintendo
S-DSP A
SONY '89
S-CPU B
S-PPU1
S-PPU2
S-WRAM
MOSEL
©1992 Nintendo
SNSP-CPU-02

This is a document intended to describe the various hardware ports on the SNES. It will not describe how these ports are used by what may be plugged into them.

Controller Ports

The controller ports of the SNES has 7 pins, laid out something like this:



Pin	Description	Wire Color
1	+5v (power)	White
2	Clock	Yellow
3	Latch	Orange
4	Data1	Red
5	Data2	?
6	IOBit	?
7	Ground	Brown

A standard controller has no wires for pins 5 and 6; these pins are used for non-standard controllers and controller splitters. Latch is written through bit 0 of register $4016. Writing 1 to this bit results in Latch going to whatever state means 'latch' to a joypad.

Clock of Port 1 is connected to the 'read' signal of `$4016`, in that reading `$4016` causes Clock to transition. Data1 and Data2 are then read, and Clock transitions back (at this point, the pad is expected to stick its next bits of data on Data1 and Data2). Clock of Port 2 is connected to `$4017`.

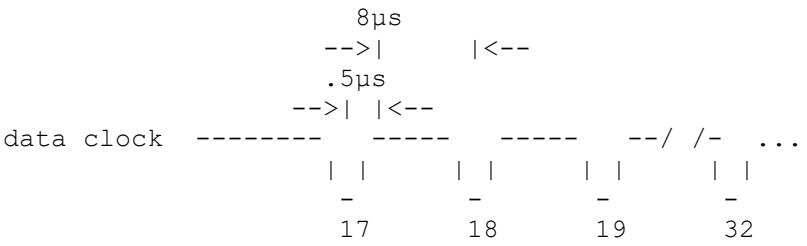
Data1 and Data2 are read through bits 0 and 1 (respectively) of `$4016` and `$4017` (for Ports 1 and 2, respectively). Thus, you must read both bits at once, you can't choose to read only Data1 and leave Data2 for later.

IOBit is connected to the I/O Port (which is accessed through registers `$4201` and `$4213`). Port 1's IOBit is connected to bit 6 of the I/O Port, and Port 2's IOBit is connected to bit 7. Note that, since bit 7 of the I/O Port is connected to the PPU Counter Latch, anything plugged into Port 2 may latch the H and V Counters by setting IOBit to 0.

Data1 and Data2 are pulled to logic-0 within the SNES, so reads will return 0 if nothing is plugged in.

Mouse Protocol

The SNES mouse uses the same timing and protocol as a regular pad for it's buttons. The left button is reported at the 9th cycle, and the right button at the 10th cycle. The SNES recognizes the mouse when the bit at the 16th clock cycle is low instead of high. 2.5ms after the 16 clock pulses, another series of clock pulses occurs. This is in fact cycles 17 to 32 since no new latch pulse had occurred yet. The data is active low, just like the buttons. This time, the clock timing is different:



Here is the meaning of the mouse specific cycles:

Clock Cycle	Button Reported
17	Y direction (0=up, 1=down)
18	Y motion bit 6
19	Y motion bit 5
20	Y motion bit 4
21	Y motion bit 3
22	Y motion bit 2
23	Y motion bit 1
24	Y motion bit 0
25	X direction (0=left, 1=right)
26	X motion bit 6
27	X motion bit 5
28	X motion bit 4

Clock Cycle	Button Reported
29	X motion bit 3
30	X motion bit 2
31	X motion bit 1
32	X motion bit 0

Each time the SNES polls the mouse, the mouse reports how it moved since last poll. When no movement occurred since last poll, all the motion bits stays high (which means binary 0). The direction bits keep their last state.

Mouse Sensitivity

The mouse has 3 configurable sensitivity levels. The currently active sensitivity level is reported by bits 11 and 12:

- Bit 11 low, Bit 12 high: High sensitivity
- Bit 11 high, Bit 12 low: Medium sensitivity
- Bit 11 high, Bit 12 high: Low sensitivity

Selecting the Sensitivity Mode

A special sequence is used to rotate between the 3 modes. First, a normal 12us latch pulse is applied. Next, the first 16 bits are read using normal button timings. Shortly after (about 1ms), 31 short latch pulses (3.4µS) are sent, with the clock going low for 700ns during each latch pulse. For selecting a specific sensitivity, simply execute the special sequence until bits 11 and 12 are as desired.

Cart Connector

The cart connector has 62 pads, laid out something like this:

```
                    +--------+
     21.477MHz Clock |  1  32 | /WRAM
              EXPAND |  2  33 | REFRESH
                 PA6 |  3  34 | PA7
               /PARD |  4  35 | /PAWR
                     |--------|
                 GND |  5  36 | GND
F                A11 |  6  37 | A12
r                A10 |  7  38 | A13
o                 A9 |  8  39 | A14
n                 A8 |  9  40 | A15
t                 A7 | 10  41 | A16
                  A6 | 11  42 | A17
```

```
o                  A5 | 12  43 | A18
f                  A4 | 13  44 | A19
                   A3 | 14  45 | A20
c                  A2 | 15  46 | A21
a                  A1 | 16  47 | A22
r                  A0 | 17  48 | A23
t                /IRQ | 18  49 | /CART
                   D0 | 19  50 | D4
                   D1 | 20  51 | D5
                   D2 | 21  52 | D6
                   D3 | 22  53 | D7
                  /RD | 23  54 | /WR
   CIC out data (p1) | 24  55 | CIC out data (p2)
    CIC in data (p7) | 25  56 | CIC in clock (p6)
              /RESET | 26  57 | CPU_CLOCK
                 Vcc | 27  58 | Vcc
                     |--------|
                 PA0 | 28  59 | PA1
                 PA2 | 29  60 | PA3
                 PA4 | 30  61 | PA5
    Left Audio Input | 31  62 | Right Audio Input
                     +--------+
```

- `A0-A23` are the Address Bus A lines.
- `/WR` and `/RD` are the associated read and write lines.
- `/WRAM` is low when the CPU is accessing WRAM.
- `/CART`, also called `/ROMSEL`, is low when the CPU is accessing ROM (banks `$40-$7D` and `$C0-$FF`, or `$8000-$FFFF` of banks `$00-$3F` and `$80-$BF`).

```
ROMSEL = (addr & 0x408000 == 0) || (addr & 0xFE0000 == 0x7E0000)
```

- `PA0-PA7` are the Address Bus B lines, with `/PARD` and `/PAWR` the associated read and write lines.
- `D0-D7` are the data bus lines.
- `REFRESH` is (presumably) the RAM refresh signal sent out each scanline for 40 master cycles.
- `EXPAND` is connected to pad 24 of the expansion port.
- `/IRQ` is connected to the CPU's `/IRQ` line. This can be read by the cart, or activated by the cart to invoke an IRQ on the CPU.
- `/RESET` is the reset signal, activated by the big reset button on the console. It can also be activated by hardware on the cart, if the system needs to be reset at a hardware level.
- `CPU_CLOCK` is (presumably) the current CPU cycle clock, which is either 6, 8, or 12 master cycles per cycle (3.58MHz, 2.68MHz, or 1.79MHz).

The signals input on Left and Right Audio Inputs are mixed into the APU's output audio.

The CIC pins are connected to the CIC chip, which is used for region lockouts. If the CIC in the console doesn't get the proper handshake over these pads, the reset signal is never released on the PPU2 chip, and so you never get anything on the display.

Many carts connect only to pins 5-27 and 36-58, as the remaining pins are mainly useful only if the cart contains special chips.

Expansion Port

The expansion port has 28 pads, laid out something like this. This pinout view faces the "cable" (i.e. expansion DEVICE socket). If you are looking at the port on the bottom of the console with the front of the console facing down, pin 1 is bottom-right and pin 28 is top-left.

```
                +--------+
           PA0 |  1   2 | PA1
           PA2 |  3   4 | PA3
           PA4 |  5   6 | PA5
           PA6 |  7   8 | PA7
          /PAWR |  9  10 | /PARD
            D0 | 11  12 | D1
            D2 | 13  14 | D3
            D4 | 15  16 | D5
            D6 | 17  18 | D7
         /RESET | 19  20 | Vcc
         SMPCLK | 21  22 | DOTCK
            GND | 23  24 | EXPAND
Mono Audio Out | 25  26 | /IRQ
 Left Audio In | 27  28 | Right Audio In
                +--------+
```

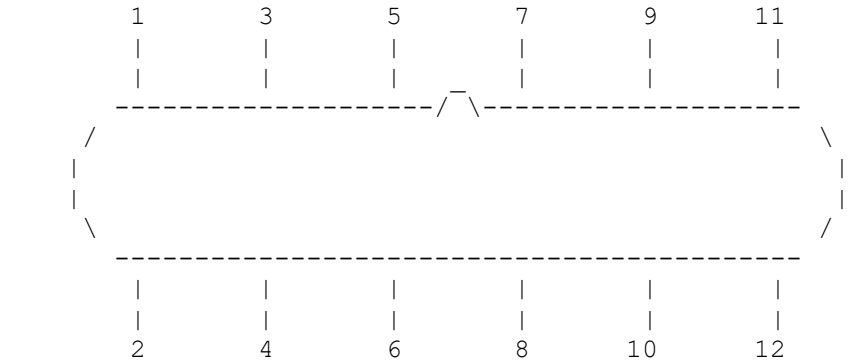
- `PA0-PA7` are the Address Bus B lines, with `/PARD` and `/PAWR` the associated read and write lines.
- `D0-D7` are the data bus lines.
- `/RESET` is the reset signal, activated by the big reset button on the console. It can also be activated by hardware on the attached device, if the system needs to be reset at a hardware level.
- `EXPAND` is connected to pad 2 of the cart connector.
- `/IRQ` is connected to the CPU's /IRQ line. This can be read by the attached device, or activated by the attached device to invoke an IRQ on the CPU.
- The signals input on Left and Right Audio Inputs are mixed into the APU's output audio. The (mono) audio output is brought back into pin 25.
- `SMPCLK` and `DOTCK` are not really known. `SMPCLK` comes from the audio subsystem; reports are that it's about 8.192MHz (i.e. 3 APU master clock cycles per cycle). `DOTCK` comes from PPU2, and seems to be PPU dot clock at about 5.369 MHz (that's 21.477/4).

Multi-Out

The multi-out port has 12 pads, and looks something like this (2 version below):

```
   --------^--------
  /11  9  7  5  3  1\
 |                   |
  \12 10  8  6  4  2/
   -----------------
```

Pin	Description
1	Red analog out
2	Green analog out
3	Composite H/V sync out ()
4	Blue analog out
5	Ground
6	Ground
7	S-Video Y (luma) signal
8	S-Video C (chroma) signal
9	Composite video signal
10	Vcc
11	Mid sound (L+R)
12	Side sound (L-R)



Pin	Description
1	Red analog video out (1v DC offset, 1vpp video into 75 ohms)
2	Green analog video out (1v DC offset, 1vpp video into 75 ohms)
3	Composite H/V sync out (1vpp into 75 ohms)
4	Blue analog video out (1v DC offset, 1vpp video into 75 ohms)
5	Ground

Pin	Description
6	Ground
7	Y (luma) signal for S-Video (1vpp into 75 ohms)
8	C (chroma) signal for S-Video (1vpp into 75 ohms)
9	Composite video signal (1vpp into 75 ohms)
10	Vcc / +5v (Could be a high logic signal or power for an external RF modulator)
11	Left channel audio out
12	Right channel audio out

RF Out

I know nothing about the output of this port, except that it has a Vcc, a GND, a video signal, and a mono audio signal.

Power

This one is simple. It's a barrel connector, a hollow cylindrical plug. It expects to be supplied with DC 10V, 850mA DC power source. The US power adapter produces centre-negative DC. At least on PAL units (SNSP), the DC jack can be fed with centre-positive or centre-negative as there is a AC-DC converter inside.

ROM Pinout

This seems to be consistent with both mask ROMs, some are 32pin, others 36pin.

```
32PIN MaskROM
=============
A17   01         32   Vcc
A18   02         31   /OE
A15   03         30   A19
A12   04         29   A14
 A7   05         28   A13
 A6   06         27   A8
 A5   07         26   A9
 A4   08         25   A11
 A3   09         24   A16
 A2   10         23   A10
 A1   11         22   /CS
 A0   12         21   D7
 D0   13         20   D6
 D1   14         19   D5
```

```
 D2   15         18   D4
Vss   16         17   D3

36PIN MaskROM
=============
A20   01         36   Vcc
A21   02         35   A22
A17   03         34   Vcc
A18   04         33   /OE
A15   05         32   A19
A12   06         31   A14
 A7   07         30   A13
 A6   08         29   A8
 A5   09         28   A9
 A4   10         27   A11
 A3   11         26   A16
 A2   12         25   A10
 A1   13         24   /CS
 A0   14         23   D7
 D0   15         22   D6
 D1   16         21   D5
 D2   17         20   D4
Vss   18         19   D3
```

DSP Pinout

This is how most DSP chips are hooked up. DSP is a uPD77C25 made by NEC.

```
 Vcc   01   28   Vcc
 Vcc   02   27   register select(A14 used when DSP is mapped to cartridge
memory region,
  NC   03   26   /CS             A12 used when DSP is mapped to expansion
memory region)
  NC   04   25   /RD
  NC   05   24   /WR
  D0   06   23   NC
  D1   07   22   NC
  D2   08   21   Vcc
  D3   09   20   Vcc
  D4   10   19   Vcc
  D5   11   18   Vcc
  D6   12   17   GND
  D7   13   16   RESET (inverted /RESET- SNES slot)
 GND   14   15   CLOCK
```

MAD-1 Pinout

The MAD-1 stands for Memory Address Decoder revision 1. It is used for memory mapping in both HiROM and LoROM. And is used for battery power control on a static RAM.

```
                        /HI  01      16  /LOW
                  SRAM /CS  02      15  A15 (LoROM), A13 (HiROM)
                        NC  03      14  BA4 (LoROM), A14 (HiROM)
                   ROM /OE  04      13  BA5
                  SRAM Vcc  05      12  Vcc or BA6 (LoROM), A15 or BA6(HiROM)...
                       Vcc  06      11  /CART (pad 49 on cartridge edge)
resistor to +3V of battery  07      10  GND=LoROM, Vcc=HiROM
                       GND  08      09  /RESET (pad 26 on cartridge edge)

 /HI <--- if two ROM chips, this selects the upper one
 /LOW <--- if two ROM chips, this selects the lower one

 Verified & Simplified by kyuusaku
 Pin  9 is A
 Pin 10 is B
 Pin 11 is C
 Pin 12 is D
 Pin 13 is E
 Pin 14 is F
 Pin 15 is G

 The Logic (Invert Outputs)
 ==========================
 Pin  1 is G E C' A + E C' B A
 Pin  2 is G F E D' C B A + G' F E D C' B' A + SRAMVCC'
 Pin  3 is G F E' D' C B A + G' F' E D C' B' A
 Pin  4 is G C' A + C' B A
 Pin 16 is G E' C' A + E' C' B A
```

CIC Lockout Chip / Security Chip

The infamous CIC. Variants include D411, D411A, D411B, F411A, D413, D413A, D413B and F413A.

```
pad24  01      16  Vcc
pad55  02      15  NC
   NC  03      14  NC
  GND  04      13  NC
   NC  05      12  NC
pad56  06      11  NC
pad25  07      10  NC
  GND  08      09  NC
```

SRAM Pinouts

```
16kbit SRAM
=======================
```

```
 A7   01         24   Vcc
 A6   02         23   A8
 A5   03         22   A9
 A4   04         21   /WE
 A3   05         20   /OE
 A2   06         19   A10
 A1   07         18   /CS
 A0   08         17   D7
 D0   09         16   D6
 D1   10         15   D5
 D2   11         14   D4
Vss   12         13   D3
```

```
256kbit SRAM (HY62256ALLP-10)
=============================
  A14   01         28   Vcc
  A12   02         27   /WE
   A7   03         26   A13
   A6   04         25   A8
   A5   05         24   A9
   A4   06         23   A11
   A3   07         22   /OE
   A2   08         21   A10
   A1   09         20   /CS
   A0   10         19   D7
   D0   11         18   D6
   D1   12         17   D5
   D2   13         16   D4
  GND   14         15   D3
```

RGB Monitors

This section covers commonly used monitors and displays used in conjunction with the Super NES or Super Famicom or generally used in retro gaming.

Casio CMV54NT04P / Sharp LM6Q401

This is the Casio (カシオ) 5.4 inch screen (sometimes written as Casio CMV54NTO4P) used in the Interact GameCube LCD screens, Iomega XBOX screens and a number of portable console mods. You can see it in action with various consoles and computers here. It can run off a standard NES power supply. Below is a pinout for the back 10-pin connection, it is clearly labeled on the PCB which pins are 1 and 6.

```
--^-----------^--
| 1  2  3  4  5 |
| 6  7  8  9 10 |
-----------------
```

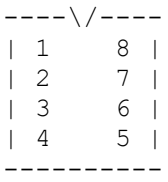
Pin	Description
1	Screen Rotation

Pin	Description
2	Blue
3	Red
4	+12V
5	Ground
6	Screen Flip
7	Composite Sync
8	Green
9	+12V
10	Brightness

In order to use a Japanese SCART with this it requires the use of a LM1881 Chip as seen below.

LM1881 Chip - Video Sync Separator

Most game systems output the composite sync information along with the NTSC video output. Many RGB displays will not accept this extraneous info along with the sync, so we need to remove the video signal. The LM1881 / LM1881N does just this. It's a simple circuit - one small chip, two small capacitors and a small resistor. Download LM1881N_Datasheet.pdf.

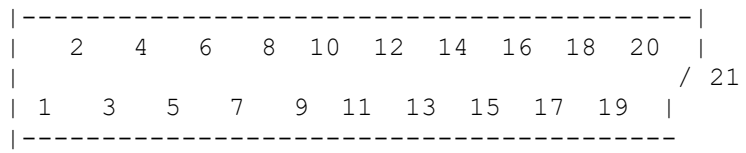


Pin	Description
1	Composite Sync Output
2	Composite Video Input
3	Vertical Sync Output
4	Ground (?)
5	Burst / Back Porch Output
6	R-Set

Pin	Description
7	Odd / Even Output
8	VCC 5-12V

Japanese 21-pin RGB Connector

This is the Japanese RGB connector used on many video game systems. It is physically identical to the European SCART / Péritel / Euroconnector with a different pinout.



1:	Audio Left Channel Input	2:	Audio Left Channel Output
3:	Audio Ground	4:	Audio Ground
5:	Audio Right Channel Input	6:	Audio Right Channel Output
7:	Sync / Video Ground	8:	Video Ground
9:	Sync / CVBS Input	10:	CVBS Output
11:	AV Control Input / +5V	12:	Ym Input
13:	Red Signal Ground	14:	Ground
15:	Red Signal I/O	16:	Ys Input / 1V
17:	Green Signal Ground	18:	Blue Signal Ground
19:	Green Signal I/O	20:	Blue Signal I/O
21:	Plug Shield / Ground		

Special Notes:

- Connect pin 16 to pin 11 via 3.3k Ohm resistor. The resistor isn't usually necessary, and +5V on this pin will most often work.
- Audio Input: 0.40 mVrms, > 47K ohms
- Audio Output: 0.40 mVrms, > 10K ohms
- CVBS (Composite Video) in and out: 1 Vp-p, 75 ohms, sync: negative
- Ym Input: Switches RGB to half-brightness, for video overlay (L: < 0.4V, H: > 1V, 75 ohms)
- Ys Input: RGB in/out: (Ground for output, 1V+ for Input (preferred)1))
- All RGB lines: 0.7 Vp-p, 75 ohms

Nintendo Super System (and SNES) repair/diagnosis logs
Main Nintendo Super System Page
Table of contents
Troubleshooting Tips
Super System motherboard repair log
CPU failures
PPU failures
"What-if" failures

Nintendo Super System / SNES troubleshooting tips
-If working on a Nintendo Super System (arcade) motherboard, be sure to use a V3 BIOS (I've seen several faulty boards wouldn't boot to the menu on V1 or V2, even though the menu circuitry was fine)

-Again, on the Nintendo Super System, be sure to check the +5V at the board and adjust. The stability of the system doesn't seem very sensitive to an exact voltage, but there is no voltage regulator (like the SNES), so the entire SNES circuitry will be running directly from the +5V from the power supply. If the voltage is adjusted too high, it's likely to significantly shorten the life of the chips (possibly the cause of some of the failures I came across).

-A bad CPU seems to be the most common failure, and has many different symptoms (see specific failures below)

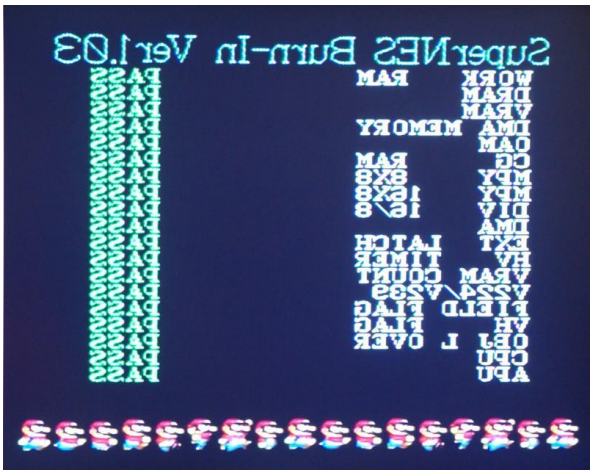
-My experience is that PPU1, PPU2, and VRAM failures don't affect game logic, just the display of the game (though in some cases will cause certain games not to boot). So, for example if the Mario Kart track looks bad and it thinks you're off the track and Lakitu keeps picking you up, it's likely a CPU problem (or possibly WRAM)... but if the game still plays fine, I'd suspect a PPU problem (though you can have CPU problems that don't affect gameplay).

-Statistics: While working with Super System and SNES motherboards, I came across 24 bad CPUs, 4 bad PPU1s, 4 bad PPU2s, 1 bad APU, and no bad WRAMs or VRAMs... if you suspect either a bad CPU or WRAM, I'd say odds are better that the CPU is bad.

-A bad CPU can cause controller problems. I'd check the obvious first (wiring, controllers, etc. on the Super System, and swap the controller, ports, ribbon on the SNES), though if that all checks out, some controller problems I've encountered from CPUs include dead controls and buttons seemingly connected together (press one button and multiples show as pressed).

-A bad APU can cause the system to not boot (not the most likely cause, but it's easy to swap on boards with the sound module).

-I highly recommend making/borrowing/buying a Burn-In test cart (described HERE). Even when other games won't boot, many times it'll boot, test the hardware, and tell you which tests failed. It won't necessarily tell you the bad chip, but you can make educated guesses from the results.



Nintendo Super System motherboard repair log
-1-
Had gray screen issue
Determined the voltage to the video mixing chip was low, due to high resistance of R71
Replaced R71
R101 (for audio circuit) was also out of spec, so replaced that too
---FIXED---

-2-
Some games worked, others had glitchy graphics (SMW looked fine, F-Zero had track issues)
Ran test cart, said bad DMA
Replaced SNES CPU
---FIXED---

-3-
Would boot to menu, but when started, some games would play music with solid color screen, others would do nothing
Ran test cart, said bad HV Timer
Replaced SNES CPU
---FIXED---

-4-
Had previously been worked on, had hacks, etc.
Would boot to menu, but when started, SNES half seemed mostly dead
Ran test cart, would show garbled main screen, when selected any sub-test, it locked up
Replaced SNES CPU Still had garbled graphics, but was able to run test... said bad VRAM, DMA, VRAM COUNT
Looked at rework of PPUs and noticed some pins on PPU1 had no solder. Resoldered PPUs and it booted and everything looked good
There was also no sound.
Looking over the other hacks, someone had removed R101. Replaced with ERJP 1/2W 10 ohm resistor.
---FIXED---

-5-
Played some games fine, but mostly mode 7 games like F-Zero and Mario Kart had problems. F-Zero was really jumpy when turning, and Mario Kart played fine by looking at the map, but the main screen had a scrambled track. Super Mario World also showed a few minor glitches, like the bar of the rotating platform was jumpy.
Ran test cart, said bad MPY 8X8 and DIV 16/8
Replaced SNES CPU
---FIXED---

-6-
Would boot to menu, but when started, SNES half seemed dead
Ran test cart, still dead
Replaced SNES CPU
---FIXED---

-7-
Would boot to menu, but when started, SNES half seemed mostly dead (occasionally booted to garbage graphics)
Ran test cart, would show main screen, but when Burn-In Test selected, would show random Japanese characters in place of graphics
Replaced SNES CPU Passed all tests, but had some graphics problems (some colors wrong, glitchy vertical lines in some graphics, etc)
Replaced SNES PPU2
---FIXED---

-8-
Would boot to menu, but when started, SNES half seemed dead (could hear it reboot occasionally)
Ran test cart, when selected any sub-test, it locked up
Replaced SNES CPU
---FIXED---

-9-
Mostly worked, occasional black screen, some sound issues
Inspected board and noticed poor soldering on modifications
Solder didn't flow on resistor mod
Large glob of solder over both L55 and L56 (shorting both together, not each)
Had loud hum when plugged into the cabinet, though sounded fine on the testbench
Suspected power supply noise was getting through to audio circuitry due to old resistor/jumper wire modification
uninstalled old modification and installed upgraded original resistors
---FIXED---

-10-
Sound not working properly
Inspected board and noticed solder pads configured for Mono audio
Soldered jumpers to original Stereo configuration
Audio worked on testbench, but not when installed in cabinet (cabinet has volume control connected to JAMMA edge, testbench has no control)

Audio is enabled by providing power to one side of potentiometer (JAMMA pin lowercase 'c'). Noticed the board was never powering this pin
Traced pin 'c' to Q15 (NPN Transistor marked "BR", likely BCW60DR), had good voltages in, but never "turned on"
Replaced with same type of transistor from dead SNES motherboard
---FIXED---

-11-
Would boot to menu, but when started, some games would run with graphics issues, others would only show black screen
Ran test cart, said HV Timer failed
Replaced SNES CPU
---FIXED---

-12-
Game plays fine, but when either Y or B are pressed, both were activated
Tested resistance between Y and B inputs, not shorted
Probed lines at IC52 (Parallel to Serial shift register) and CPU and everything looked fine
Replaced SNES CPU
---FIXED---

-13-
Would boot to menu, but when started, SNES half seemed dead
Ran test cart, still dead
Replaced SNES APU
---FIXED---

-14-
Would boot to menu, but when started, SNES half seemed dead
Ran test cart, would boot, but display garbage after selecting test
Replaced SNES CPU Test then showed VRAM HIGH FAIL, DMA FAIL, VRAM COUNT FAIL
Replaced SNES PPU1
---FIXED---

-15-
Would boot, but all the graphics were blocky garbage, though you could tell that the games were playing (scrolling background, track, etc).
Replaced SNES PPU1
---FIXED---
Note: The +5V was adjusted to over 6.5V, which may have contributed to PPU1 failure

-16-
Would boot to menu, but when started, some games would run with weird problems, some would lock up during boot
Ran test cart, passed all tests
Replaced SNES CPU
---FIXED---

-17-
Would boot to menu, but when started, SNES half seemed dead
Ran test cart, still dead (occasionally showed menu and immediately locked up)
Replaced SNES CPU
---FIXED---

-18-
Would boot to menu, but when started, SNES half seemed dead
Ran test cart, still dead
Replaced SNES CPU
---FIXED---

Misc CPU failures

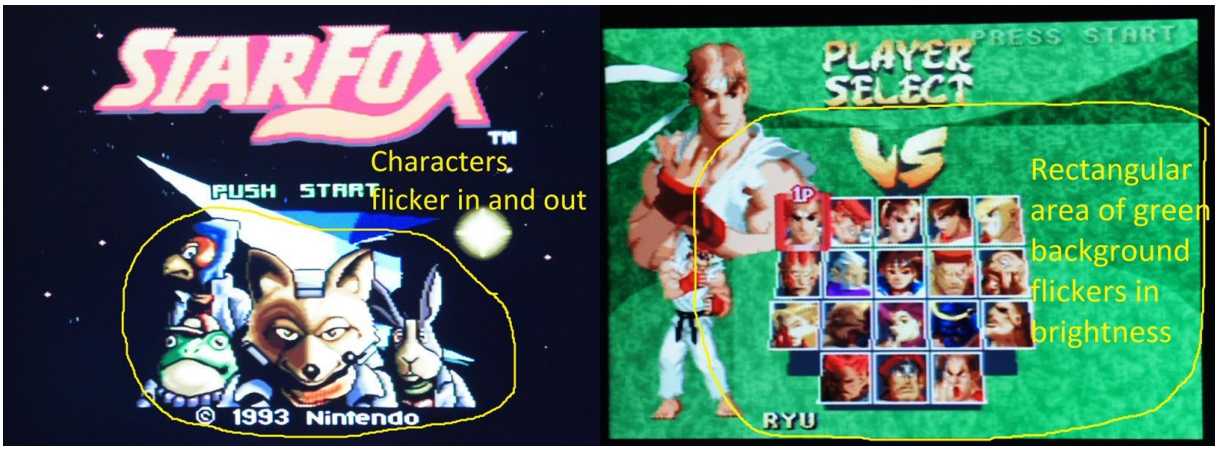
Because of the large number of bad CPUs I encountered, I installed a QFP socket in a Super Famicom system to quickly test suspect CPUs and pulls from unknown motherboards.

1) Won't boot most games, test cart boots, but shows Japanese characters at burn-in test
2) Won't boot most games, test cart hangs at boot
3) Won't boot most games, test cart boots, but locks up after selecting any test
4) Won't boot most games, test cart boots, but locks up after selecting any test (and timer never increases)
5) Dead
6) Won't boot most games, test cart boots, but shows garbage at burn-in test
7) Dead
8) Dead
9) Dead
10) Dead
11) Dead
12) Dead
13) Plays most games fine, all tests pass, won't boot some games (SFA2, Yoshi's Island, etc)
14) Some games boot to garbage or audio only, test cart boots, HV Timer failed
15) Won't boot most games, test cart hangs at boot
16) Won't boot most games, test cart passes all tests
17) Some games boot to garbage or audio only, test cart boots, HV Timer failed
18) Everything works except dead controls
19) Everything works except lots of buttons appear shorted together (press one and lots of buttons are "pressed")
20) Everything works except Y and B appear shorted together (press one and both buttons are "pressed")
21) Some games work correctly, others have weird glitches (like floating shadows in F-Zero), test cart says DMA failed

22) Most games work correctly, some have weird sprite glitches (SFA2 bad flipped sprites, SMK karts on bottom map), some hang during boot (Yoshi's Island), test cart passes all tests

23) Lots of games play fine (like SMW, though the rotating bar is jumpy), but some act really weird (especially the track on Mode 7 games, and the ship on Star Fox is all over the place), test cart says MPY 8x8 and DIV 16/8 failed

24) Lots of games play fine, but some have a few small glitches (flickering graphics)



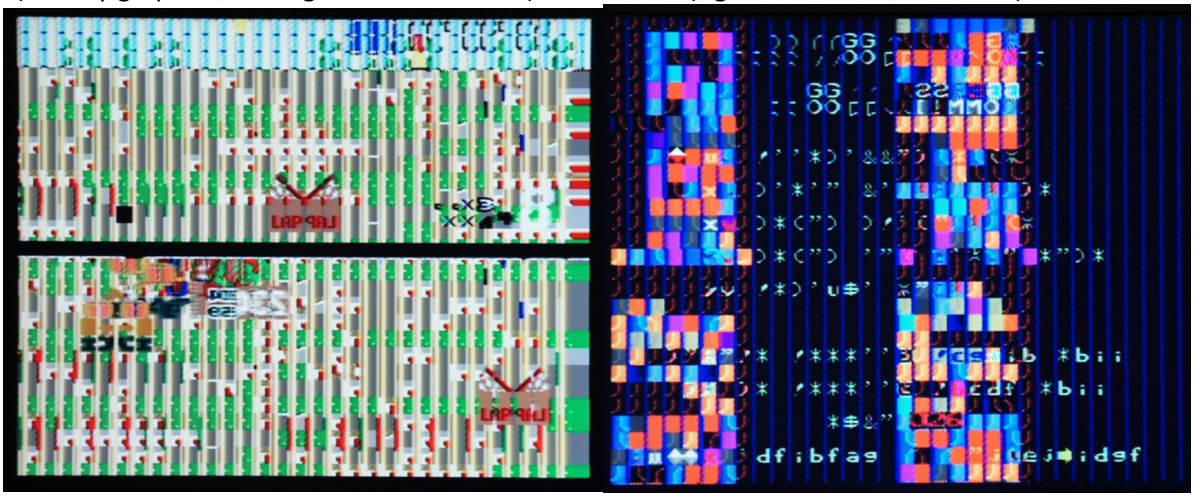
25) Won't boot most games, test cart passes all tests
26) Dead
27) Dead

Misc PPU failures

PPU1

1) Blocky graphics, thick vertical lines, some games won't boot (Yoshi's Island), games run fine, fails test (can't read error)

2) Blocky graphics, some games won't boot (Yoshi's Island), games run fine, fails test (can't read error)



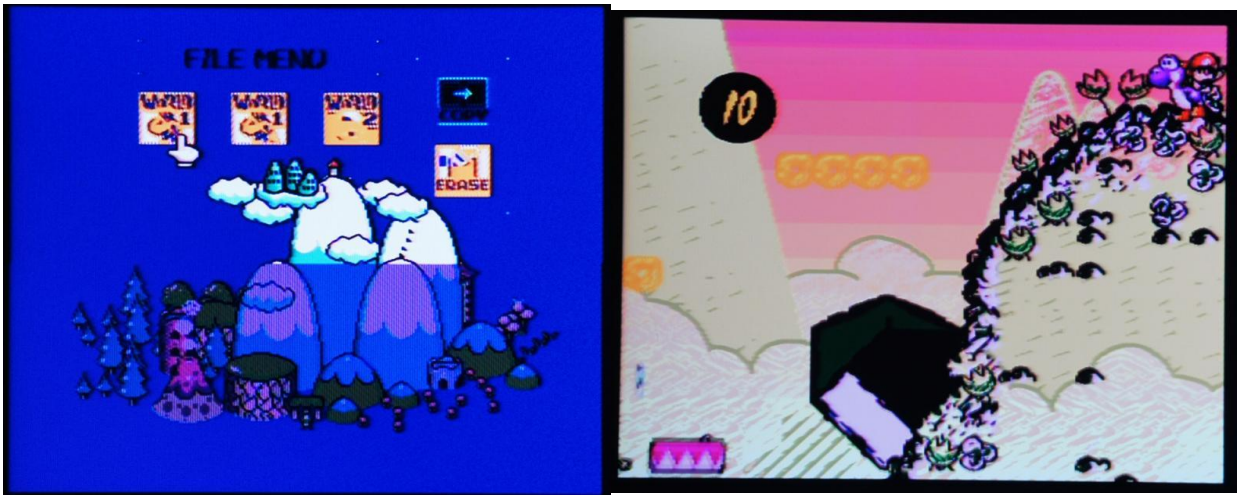
3) Almost everything is fine (Yoshi's Island has a few vertical lines in a few parts), games run fine, fails test (VRAM HIGH, DMA, VRAM COUNT failed)

4) Blocky graphics, games run fine, fails test (can't read error)

PPU2

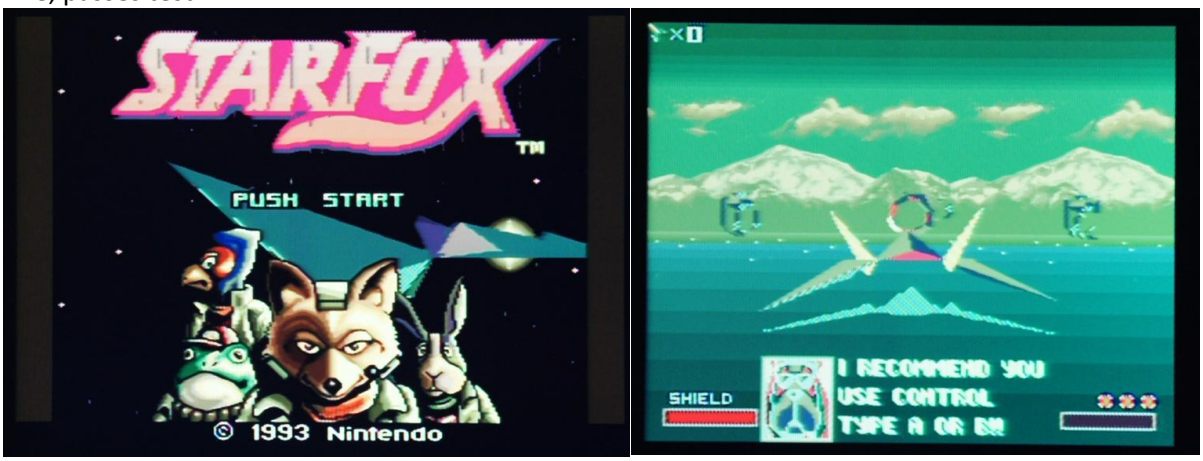
1) Missing a lot of objects, mostly missing red, some games won't boot (Star Fox), games run fine, fails test (CG RAM, EXT LATCH, HV TIMER, V224/V239 failed)

2) Colors wrong, jumpy horizontal scrolling, some games won't boot (Star Fox, SFA2), games run fine, fails test (can't see error)



3) Lots of wrong colors, flickery vertical lines through lots of graphics, missing some graphics, games run fine, passes test

4) Most colors seem tinted yellow, some wrong colors, some vertical lines through graphics, games run fine, passes test



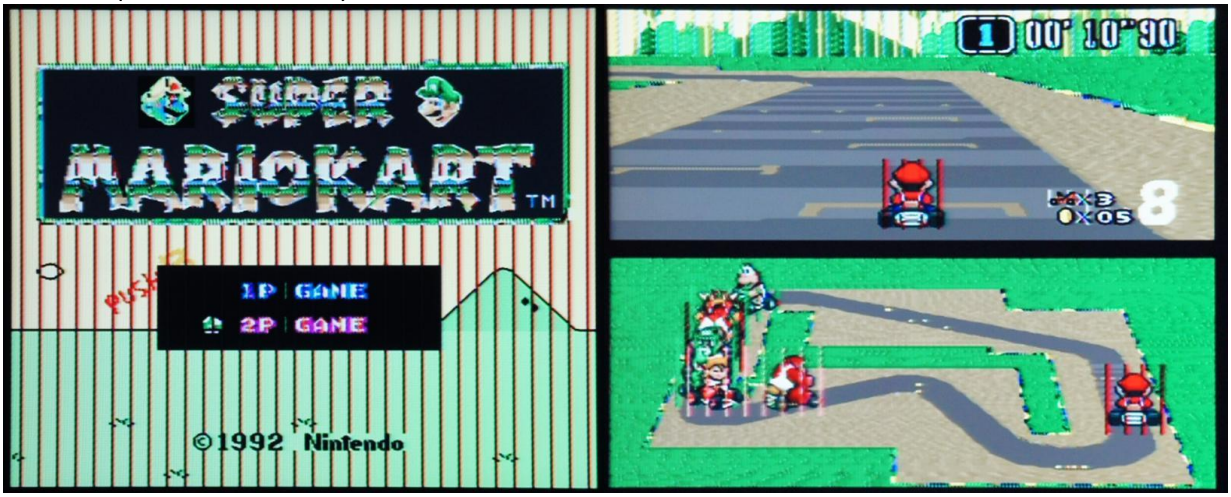
"What-if" failures

WRAM

Dead or bad chip can cause dead system, weird behavior, lockups, etc.
Fails test (WRAM failed) if test is able to run

VRAM

_U4_
Colors wrong, wrong graphics, narrow vertical lines in graphics, games run fine
Fails test (VRAM HIGH failed)

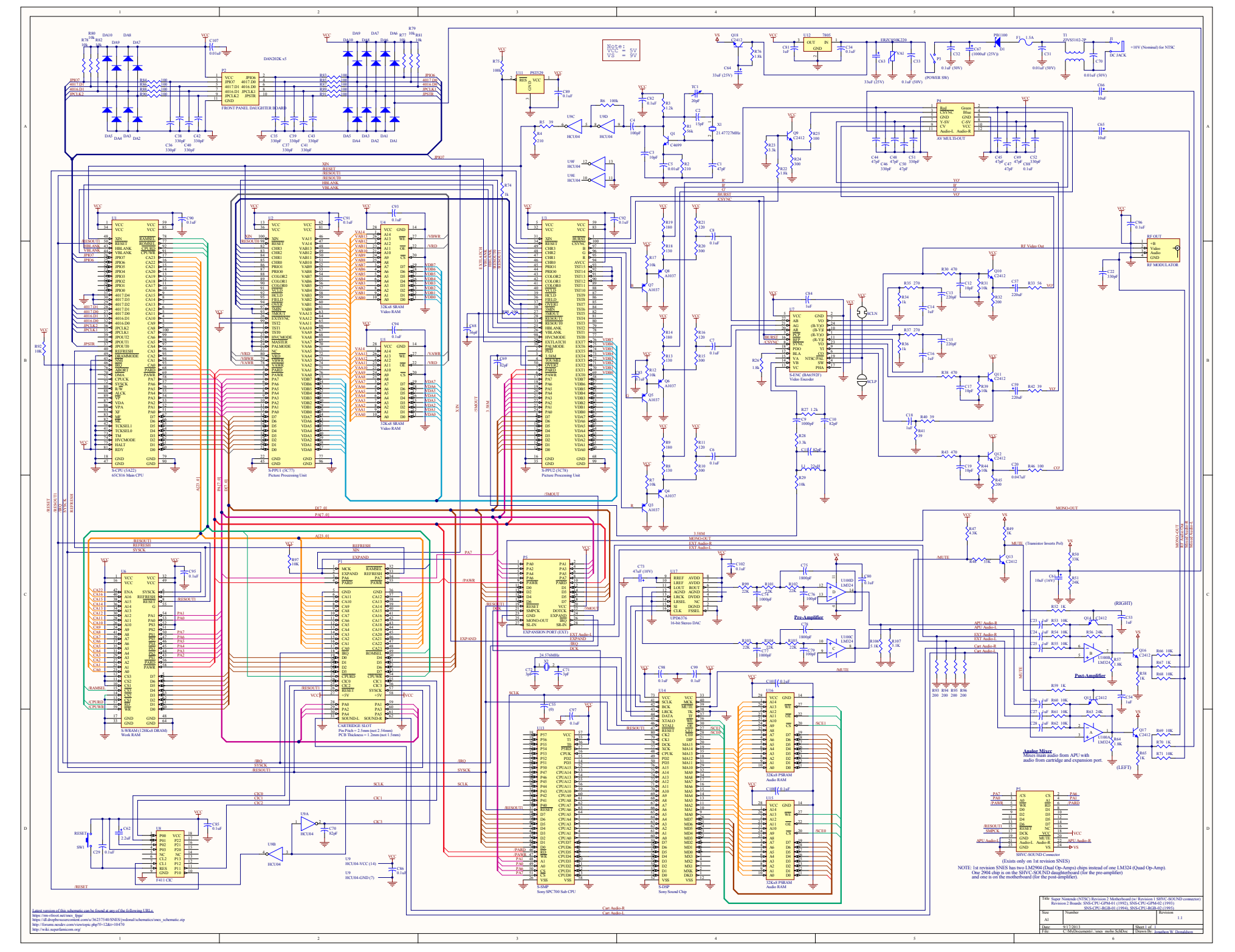


_U5_
Wrong graphics, narrow vertical lines in graphics, weird graphics combinations, lots of duplicated graphics top and bottom, games run fine
Fails test (VRAM LOW failed)

PAL

SNSP-CPU-01 (1992)
SNSP-CPU-02 (1992)
arzi84
Sheet: /
File: PAL-SNES_Schematic.sch
Title: Super Nintendo Entertainment System (PAL)

Size: A1	Date: 2017-11-04	**Rev: 0.8**
KiCad E.D.A. kicad 4.0.6+dfsg1-1		Id: 1/1


Note:
Vcc = 5V
Vs = 9V
S-CPU-1 (5A22)
65C816 Main CPU
S-PPU1 (5C77)
Picture Processing Unit
S-PPU (5C78)
Picture Processing Unit
32Kx8 SRAM
Video RAM
S-ENC (BA6592F)
Video Encoder
RF MODULATOR
AV MULTI-OUT
FRONT PANEL DAUGHTER BOARD
S-WRAM (128Kx8 DRAM)
Work RAM
EXPANSION PORT (EXT)
CARTRIDGE SLOT
Pin Pitch = 2.5mm (not 2.54mm)
PCB Thickness = 1.2mm (not 1.5mm)
S-SMP
Sony SPC700 Sub CPU
S-DSP
Sony Sound Chip
32Kx8 PSRAM
Audio RAM
SHVC-SOUND Connector
(Exists only on 1st revision SNES)
F411 CIC
UPD6376
16-bit Stereo DAC
Pre-Amplifier
Post-Amplifier
Analog Mixer
Mixes main audio from APU with
audio from cartridge and expansion port.
(RIGHT)
(LEFT)
(POWER SW)
(Transistor Inverts Pol)
+10V (Nominal) for NTSC
DC JACK
RF Video Out
MUTE
/MUTE
MONO-OUT
EXT Audio-R
EXT Audio-L
Cart Audio-R
Cart Audio-L
APU Audio-R
APU Audio-L
3.58M
24.576MHz
NOTE: 1st revision SNES has two LM2904 (Quad Op-Amps) chips instead of one LM324 (Quad Op-Amp).
One 2904 chip is on the SHVC-SOUND daughterboard (for the pre-amplifier)
and one is on the motherboard (for the post-amplifier).
Latest version of this schematic can be found at any of the following URLs:
https://nn-elloot.net/snes_fpga/
https://dl.dropboxusercontent.com/u/36225740/SNES/jwdonal/schematics/snes_schematic.zip
http://forums.nesdev.com/viewtopic.php?f=12&t=10470
http://wiki.superfamicom.org
Title: Super Nintendo (NTSC) Revision 2 Motherboard (w/ Revision 1 SHVC-SOUND connector)
Revision 2 Boards: SNS-CPU-GPM-01 (1992), SNS-CPU-GPM-02 (1993)
SNS-CPU-BUB-01 (1994), SNS-CPU-BUB-02 (1995)
Size: A1
Number:
Revision: 1.1
Date: 4/17/2013
Sheet 1 of 1
File: C:\MyDocuments\...snes_mobo.SchDoc
Drawn By: Jonathan W. Donaldson