

Estructura del Juego Super Mario Bros (NES)

Este documento describe la estructura técnica y arquitectura del juego Super Mario Bros basada en el análisis del assembler y código fuente.

Índice

- 1. [Arquitectura General](#)
- 2. [Estructura de Memoria](#)
- 3. [Sistema de Paletas](#)
- 4. [Sistema de Colisiones](#)
- 5. [Sistema de Niveles](#)
- 6. [Sistema de Sprites](#)
- 7. [Rutinas Principales](#)
- 8. [Datos y Tablas](#)

Arquitectura General

Plataforma

- **Consola:** Nintendo Entertainment System (NES)
- **Procesador:** 6502 (1.79 MHz)
- **Memoria:** 2KB RAM + 32KB PRG-ROM + 8KB CHR-ROM
- **Formato:** iNES (.nes)

Estructura de Archivo ROM

SuperMarioBros.nes

- Header iNES (16 bytes)
- PRG-ROM (32KB)
 - Banco 0 (\$8000-\$BFFF)
 - Banco 1 (\$C000-\$FFFF)
- CHR-ROM (8KB)
 - Tiles gráficos (512 tiles × 16 bytes)

Estructura de Memoria

Mapeo de Memoria NES

\$0000-\$07FF: RAM del sistema (2KB)

\$0800-\$0FFF: Mirror de RAM

\$2000-\$2007: Registros PPU

\$4000-\$4017: Registros APU/Input

\$6000-\$7FFF: SRAM (no usado en SMB)

```
$8000-$BFFF: PRG-ROM Banco 0  
$C000-$FFFF: PRG-ROM Banco 1
```

Variables Importantes en RAM

- **\$0744: BackgroundColorCtrl** - Control de color de fondo
- **\$074E: AreaType** - Tipo de área (agua, normal, subterráneo, castillo)
- **\$0756: PlayerSize** - Tamaño de Mario (pequeño/grande)
- **\$0757: PlayerStatus** - Estado de Mario (normal, fuego, estrella)

Sistema de Paletas

Estructura de Paletas NES

El NES usa un sistema de paletas de 4 colores por sprite/fondo:

- **4 paletas de sprites** (16 colores total)
- **4 paletas de fondo** (16 colores total)
- **Color universal** (1 color compartido)

Paletas Principales en Super Mario Bros

Paletas de Sprites

```
tab_b0_85cb: ; Sprites principales  
    .bulk $0f,$16,$27,$18 ; Mario (piel, gorra roja, sombra, overall  
azul)  
    .bulk $0f,$30,$27,$19 ; Luigi (piel, gorra verde, sombra, overall  
verde)  
    .bulk $0f,$37,$27,$16 ; Mario fuego (piel, amarillo, sombra, rojo)
```

Paletas de Fondo

```
GroundPaletteData: ; Niveles normales  
    .bulk $0f,$29,$1a,$0f ; Suelo/cielo normal  
    .bulk $0f,$36,$17,$0f ; Bloques  
    .bulk $0f,$30,$21,$0f ; Pipes  
    .bulk $0f,$27,$17,$0f ; Decoraciones  
  
WaterPaletteData: ; Niveles de agua  
    .bulk $0f,$15,$12,$25 ; Agua azul  
    .bulk $0f,$3a,$1a,$0f ; Bloques submarinos  
    .bulk $0f,$30,$12,$0f ; Pipes submarinos  
    .bulk $0f,$27,$12,$0f ; Decoraciones submarinas  
  
UndergroundPaletteData: ; Niveles subterráneos  
    .bulk $0f,$0f,$30,$21 ; Fondo oscuro
```

```
.bulk $0f,$36,$17,$0f ; Bloques
.bulk $0f,$30,$21,$0f ; Pipes
.bulk $0f,$27,$17,$0f ; Decoraciones
```

Colores de Fondo Dinámicos

```
BackgroundColors:
    .bulk $22,$22,$0f,$0f ; Colores base del cielo
```

Sistema de Selección de Paletas

```
VRAM_AddrTable_Low: ; Tabla de direcciones de paletas
    .dd1 <VRAM_Buffer1
    .dd1 <WaterPaletteData
    .dd1 <GroundPaletteData
    .dd1 <UndergroundPaletteData
    .dd1 <CastlePaletteData
    .dd1 <DaySnowPaletteData
    .dd1 <NightSnowPaletteData
```

Sistema de Colisiones

Rutina Principal de Colisión

```
PlayerEnemyCollision: ; $D88A
    LDA StarInvincibleTimer ; Verificar timer de estrella
    BEQ HandlePECollisions ; Si timer = 0, daño normal
    JMP ShellOrBlockDefeat ; Si no, matar enemigo
```

Tipos de Colisión

1. **Mario vs Enemigo:** Daño o muerte del enemigo
2. **Mario vs Power-up:** Cambio de estado
3. **Mario vs Bloque:** Interacción con bloques
4. **Mario vs Plataforma:** Física de salto

Estados de Mario

- **Pequeño:** 1 hit = muerte
- **Grande:** 1 hit = pequeño
- **Fuego:** Puede disparar bolas de fuego
- **Estrella:** Invencible temporalmente

Sistema de Niveles

Estructura de Nivel

```
Nivel
├── Metadatos
│   ├── Tipo de área (agua, normal, subterráneo, castillo)
│   ├── Paleta de fondo
│   ├── Música
│   └── Objetivos
├── Layout de Bloques
│   ├── Suelo
│   ├── Bloques destructibles
│   ├── Bloques de poder
│   └── Pipes
├── Enemigos
│   ├── Goombas
│   ├── Koopas
│   ├── Piranha Plants
│   └── Bowser (nivel final)
└── Power-ups
    ├── Hongos
    ├── Flores de fuego
    └── Estrellas
```

Tipos de Área

- **Área 0:** Niveles de agua (1-2, 2-2, etc.)
- **Área 1:** Niveles normales (1-1, 1-3, etc.)
- **Área 2:** Niveles subterráneos (1-2, 4-2, etc.)
- **Área 3:** Castillos (1-4, 2-4, etc.)

Sistema de Sprites

Estructura de Sprite

Cada sprite ocupa 4 bytes en OAM (Object Attribute Memory):

```
Byte 0: Posición Y
Byte 1: Tile ID
Byte 2: Atributos (paleta, flip, prioridad)
Byte 3: Posición X
```

Sprites Principales

- **Mario:** Sprites 0-3 (cuerpo, cabeza, brazos, piernas)
- **Enemigos:** Goombas, Koopas, Piranha Plants

- **Power-ups:** Hongos, flores, estrellas
- **Efectos:** Partículas, explosiones

Animación

- **Frames de animación:** Cambio de tile ID por frame
- **Estados:** Caminando, saltando, agachado
- **Dirección:** Flip horizontal según dirección

Rutinas Principales

Rutinas de Inicialización

```
Reset:                ; $8000 – Punto de entrada
    SEI                ; Deshabilitar interrupciones
    CLD                ; Modo decimal off
    LDX #$FF           ; Inicializar stack
    TXS
    JMP InitializeMemory

InitializeMemory:     ; $8009
    ; Limpiar RAM
    ; Configurar PPU
    ; Cargar paletas iniciales
```

Rutinas de Juego Principal

```
GameLoop:             ; $80A0
    JSR ReadControllers
    JSR UpdatePlayer
    JSR UpdateEnemies
    JSR UpdateSprites
    JSR UpdateScreen
    JMP GameLoop
```

Rutinas de Colisión

```
PlayerEnemyCollision: ; $D88A
PlayerBlockCollision: ; $D1E0
PlayerPowerUpCollision: ; $D2A0
```

Rutinas de Renderizado

```
UpdateScreen:      ; $8E00  
    JSR UpdateScroll  
    JSR UpdatePalettes  
    JSR UpdateSprites  
    RTS
```

Datos y Tablas

Tablas de Paletas

- **tab_b0_85cb**: Paletas de sprites principales
- **GroundPaletteData**: Paletas de niveles normales
- **WaterPaletteData**: Paletas de niveles de agua
- **UndergroundPaletteData**: Paletas de niveles subterráneos
- **CastlePaletteData**: Paletas de castillos
- **DaySnowPaletteData**: Paletas de nieve (día)
- **NightSnowPaletteData**: Paletas de nieve (noche)

Tablas de Control

- **VRAM_AddrTable_Low**: Direcciones bajas de paletas
- **VRAM_AddrTable_High**: Direcciones altas de paletas
- **BGColorCtrl_Addr**: Control de colores de fondo
- **BackgroundColors**: Colores base del cielo

Tablas de Datos de Área

- **AreaPalette**: Paletas por tipo de área
- **Palette3Data**: Paletas específicas por área
 - Área 0 (agua): **0f 07 12 0f**
 - Área 1 (normal): **0f 07 17 0f**
 - Área 2 (subterráneo): **0f 07 17 1c**
 - Área 3 (castillo): **0f 07 17 00**

Flujo de Ejecución

Secuencia de Inicio

1. **Reset** → Inicializar hardware
2. **InitializeMemory** → Limpiar RAM y configurar PPU
3. **LoadTitleScreen** → Mostrar pantalla de título
4. **GameLoop** → Bucle principal del juego

Bucle Principal

1. **ReadControllers** → Leer input del jugador

2. **UpdatePlayer** → Actualizar estado de Mario
3. **UpdateEnemies** → Actualizar enemigos
4. **CheckCollisions** → Verificar colisiones
5. **UpdateSprites** → Actualizar sprites en pantalla
6. **UpdateScreen** → Renderizar frame

Rutinas de Colisión

1. **PlayerEnemyCollision** → Mario vs enemigos
2. **PlayerBlockCollision** → Mario vs bloques
3. **PlayerPowerUpCollision** → Mario vs power-ups

Modificaciones Comunes

Cambios de Paleta

- **Cielo nocturno:** Modificar `BackgroundColors`
- **Skins de Mario:** Cambiar `tab_b0_85cb`
- **Efectos visuales:** Modificar paletas de área

Cambios de Gameplay

- **Invencibilidad:** NOP en `PlayerEnemyCollision`
- **Velocidad:** Modificar timers de animación
- **Física:** Cambiar valores de gravedad/salto

Cambios de Contenido

- **Tiles:** Modificar CHR-ROM
- **Niveles:** Cambiar layouts de bloques
- **Enemigos:** Modificar sprites y comportamiento

Herramientas de Desarrollo

Scripts Disponibles

- `patch_sky_palette_final.py`: Modificar colores del cielo
- `patch_mario_full.py`: Crear skins de Mario
- `patch_star_invincible.py`: Hacer Mario invencible
- `mutate_chr_range_args.py`: Modificar tiles gráficos

Archivos de Referencia

- `mario.txt`: Desensamblado completo del juego
- `SuperMarioBros_disasm.asm`: Assembler fuente
- `SCRIPTS_DOCUMENTATION.md`: Documentación de herramientas

