# Documentación Técnica: SP1 Mk2 MIDI Controller Firmware

## 1. Introducción General

El SP1 Mk2 es un controlador MIDI modular avanzado basado en ESP32-S3 que funciona como núcleo lógico del sistema SP1. Este firmware gestiona:

* Pantallas OLED de 128x128 píxeles
* Sistema de navegación por menús mediante encoder rotativo
* Almacenamiento y recuperación de presets en tarjeta SD
* Secuenciador de pasos completamente editable
* Sincronización externa vía MIDI Clock (DIN y USB)
* Comunicación MIDI dual (USB y UART)

El firmware combina funcionalidades de:

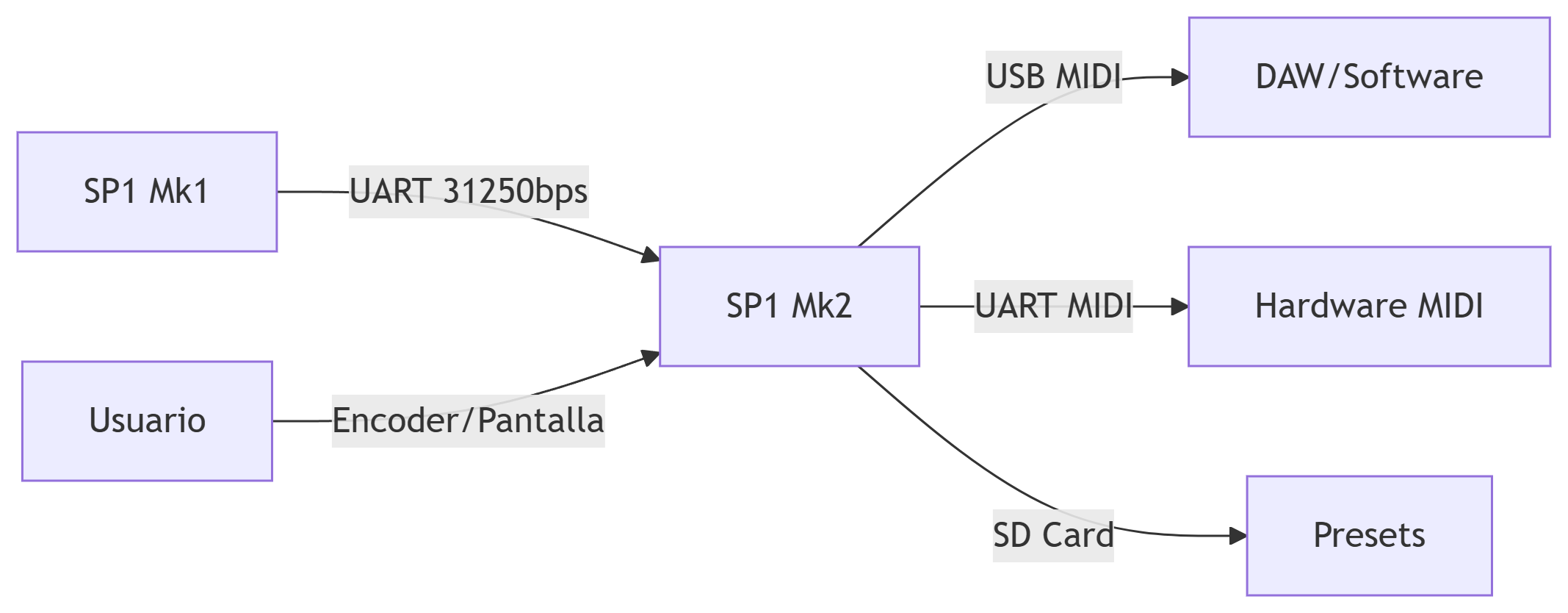
* Superficie de control MIDI (108 controles físicos)
* Teclado MIDI
* Secuenciador paso a paso polifónico

## 2. Arquitectura del Sistema

### 2.1 Componentes Principales

| **Módulo** | **Plataforma** | **Función Principal** |
| --- | --- | --- |
| SP1 Mk2 | ESP32-S3 | Cerebro central (gestión de menús, pantalla, secuenciador, presets) |
| SP1 Mk1 | Arduino Micro | Esclavo multiplexador (gestión de 108 controles físicos) |

### 2.2 Flujo de Comunicación



### 2.3 Especificaciones Técnicas

* **Procesador**: ESP32-S3 (240MHz dual-core)
* **Memoria Flash**: 8MB
* **Comunicación**: USB MIDI + UART MIDI (31250 baudios)
* **Almacenamiento**: MicroSD (FAT32)
* **Pantalla**: OLED 128x128 (SSD1351)

## 3. Sistema de Menú y Navegación

### 3.1 Estructura de Pantalla

La interfaz se divide en 3 zonas interactivas:

1. **Zona Superior** (2 líneas, textSize=2)
   * Menús principales y submenús
2. **Zona Central** (2 líneas, textSize=2)
   * Parámetros de grupo y controles específicos
3. **Zona Inferior** (1 línea, textSize=3)
   * Valores y acciones seleccionables

### 3.2 Navegación por Encoder

| **Acción** | **Función** |
| --- | --- |
| Giro | Cambiar valores/selecciones |
| Clic Corto | Confirmar selección |
| Clic Largo | Cambiar entre zonas activas |
| Giro + Pulsación | Acceso rápido a funciones avanzadas |

### 3.3 Estados de Pantalla

| **Pantalla** | **Descripción** |
| --- | --- |
| Inicio/Test | Animación inicial y autodiagnóstico |
| Standby | Muestra eventos MIDI en tiempo real |
| CONTROL | Último control físico modificado |
| PRESETS | Gestión de LOAD/SAVE/DELETE |
| CONFIG | Ajustes globales del sistema |
| SEQUENCER | Edición completa de pasos y patrones |

## 4. Secuenciador Paso a Paso

### 4.1 Estructura de Datos

cpp

struct Step {

bool active = false;

uint8\_t note = 60; *// 0-127*

uint8\_t velocity = 100; *// 0-127*

uint16\_t duration = 1; *// Duración en ticks*

uint8\_t inicio = 0; *// Tick de inicio*

};

Step secuencia[NUM\_SEQUENCERS][MAX\_STEPS\_CONFIGURABLE];

### 4.2 Características Principales

* **4 pistas** independientes
* Hasta **64 pasos configurables** por pista
* Parámetros editables por paso:
  + Activación (On/Off)
  + Nota musical (0-127)
  + Velocidad (0-127)
  + Duración (ticks)
  + Punto de inicio
* Modos de reproducción:
  + Mono/Poli
  + Legato ajustable (0-100%)
  + Sustain personalizable (0-200%)

### 4.3 Funciones Clave

cpp

void avanzarPaso(); *// Avanza paso en pista activa*

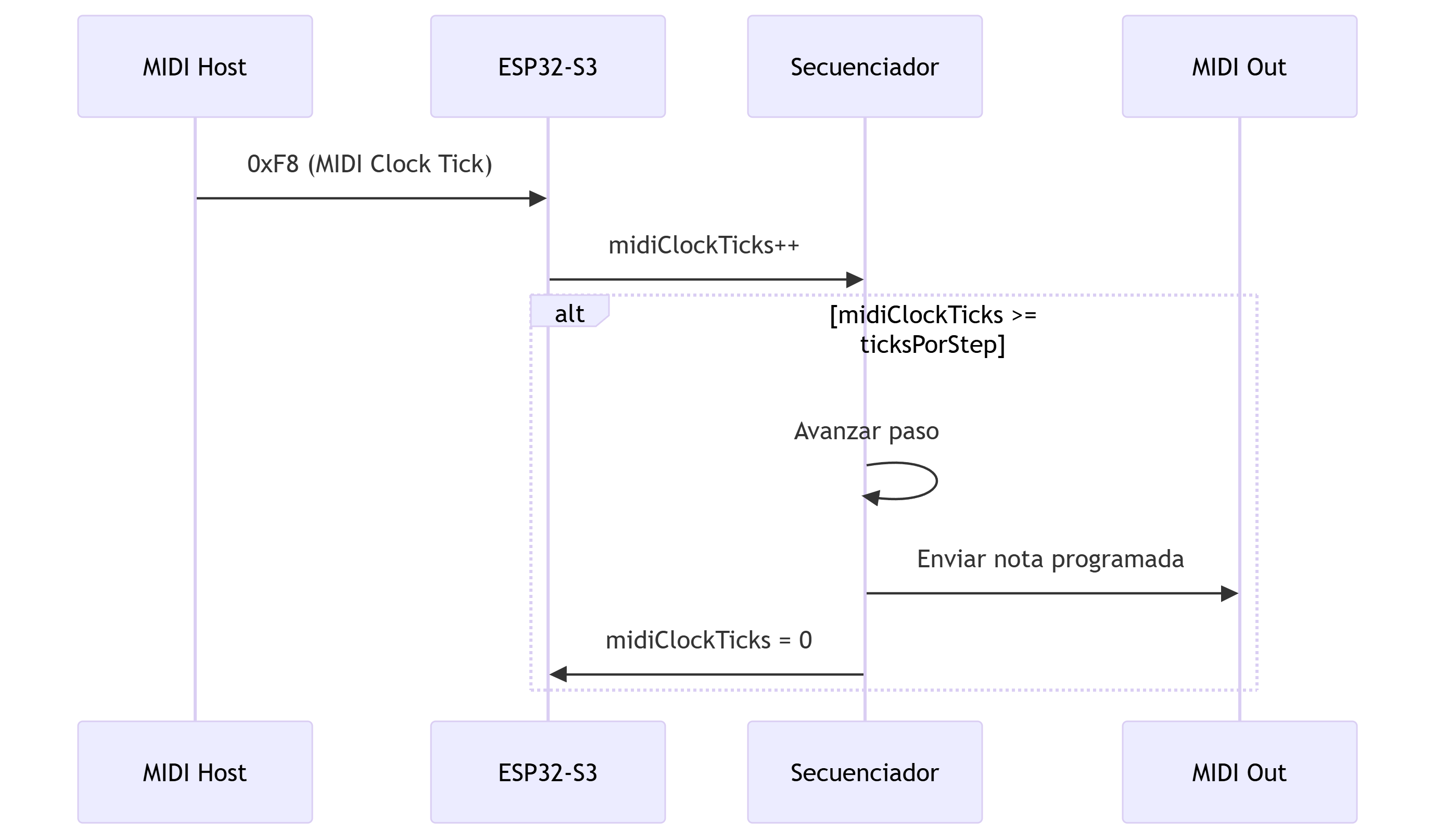
void avanzarSecuencia(); *// Avanza todos los pasos activos*

void editarPaso(uint8\_t index, const Step& nuevoPaso);

void establecerTotalSteps(uint8\_t total);

## 5. Sincronización MIDI Clock

### 5.1 Implementación Actual



### 5.2 Especificaciones Técnicas

* **Ticks por paso**: 6 (1/16 de nota)
* **Detección**: Mensajes 0xF8 (Clock) por USB y UART
* **Configuración**: Activación/desactivación desde menú

### 5.3 Futuras Implementaciones

cpp

*// En desarrollo para v0.7.0*

void handleSystemRealTimeMessage(uint8\_t message) {

switch(message) {

case 0xFA: *// START*

startSequencer();

break;

case 0xFB: *// CONTINUE*

continueSequencer();

break;

case 0xFC: *// STOP*

stopSequencer();

break;

}

}

## 6. Sistema de Presets y Almacenamiento

### 6.1 Estructura de Directorios

text

/Presets Surface/

SERUM\_SP1\_SURFACE\_PRESET01.csv

SERUM\_SP1\_SURFACE\_PRESET02.csv

/Presets Seq/

SERUM\_SP1\_SEQUENCER\_PRESET01.csv

SERUM\_SP1\_SEQUENCER\_PRESET02.csv

### 6.2 Formato de Archivos CSV

**Ejemplo para preset de superficie:**

csv

origin,group,name,channel,cc,value,mux

Pot,KNOB,FILTER CUTOFF,0,74,64,0

Switch,EFFECT,REVERB ON/OFF,0,12,127,7

**Ejemplo para preset de secuenciador:**

csv

type,content

config,120,50,100,true,true

track,0,true,60,100,1,true,16,32,0

step,0,0,true,62,100,0,8

### 6.3 Operaciones desde Menú

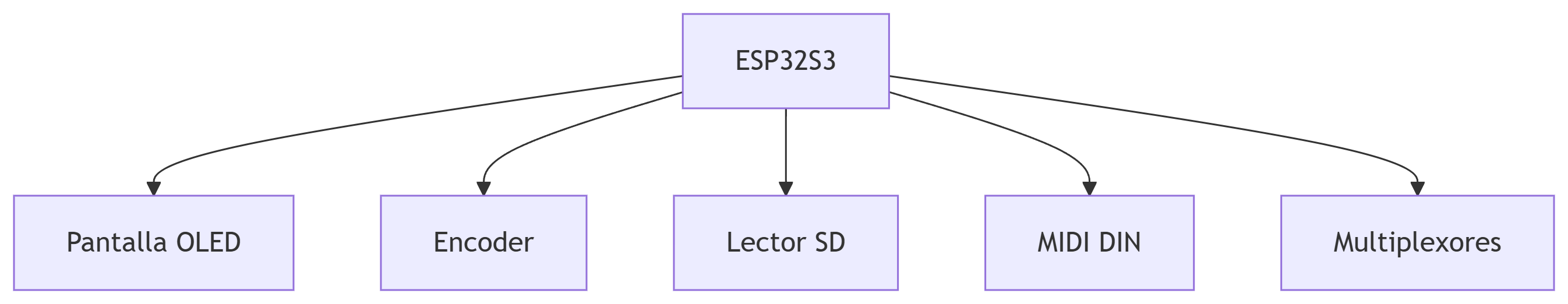
1. Seleccionar acción (LOAD/SAVE/DELETE)
2. Elegir categoría (Surface/Sequencer)
3. Seleccionar preset de lista
4. Confirmar con OK/CANCEL

## 7. Variables Globales Clave

| **Variable** | **Tipo** | **Descripción** |
| --- | --- | --- |
| estadoTempo | uint16\_t | BPM actual (30-300) |
| bpmSyncEnabled | bool | Sincronización MIDI Clock activa |
| midiClockTicks | volatile uint8\_t | Contador de ticks MIDI |
| muteSequencerNotes | bool | Silencia salida del secuenciador |
| secuenciaTecladoLinkeada | bool | Vincula teclado a secuenciador |
| porcentajeLegato | uint8\_t | Nivel de legato (0-100%) |
| porcentajeSustain | uint8\_t | Nivel de sustain (0-200%) |
| secuenciadorGlobalActivo | bool | Estado global del secuenciador |

## 8. Hardware y Asignación de Pines

### 8.1 Diagrama de Conexiones

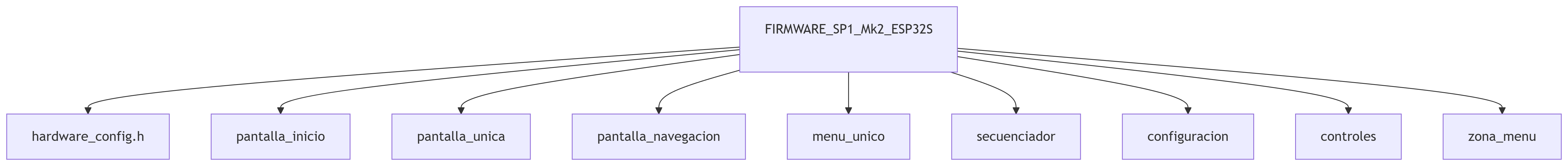


### 8.2 Asignación de Pines

| **Función** | **GPIO** | **Descripción** |
| --- | --- | --- |
| MIDI\_UART\_TX | 14 | Salida MIDI DIN |
| MIDI\_UART\_RX | 13 | Entrada MIDI DIN |
| ENCODER\_A\_PIN | 34 | Encoder fase A |
| ENCODER\_B\_PIN | 35 | Encoder fase B |
| ENCODER\_BUTTON\_PIN | 36 | Botón encoder |
| OLED\_CS | 5 | Chip select OLED |
| OLED\_DC | 16 | OLED data/command |
| OLED\_RST | 17 | Reset OLED |
| SD\_CS | 10 | Chip select tarjeta SD |
| SPI\_SCK | 12 | Reloj SPI |
| SPI\_MISO | 14 | MISO SPI |
| SPI\_MOSI | 3 | MOSI SPI |

## 9. Estructura del Código

### 9.1 Diagrama de Módulos

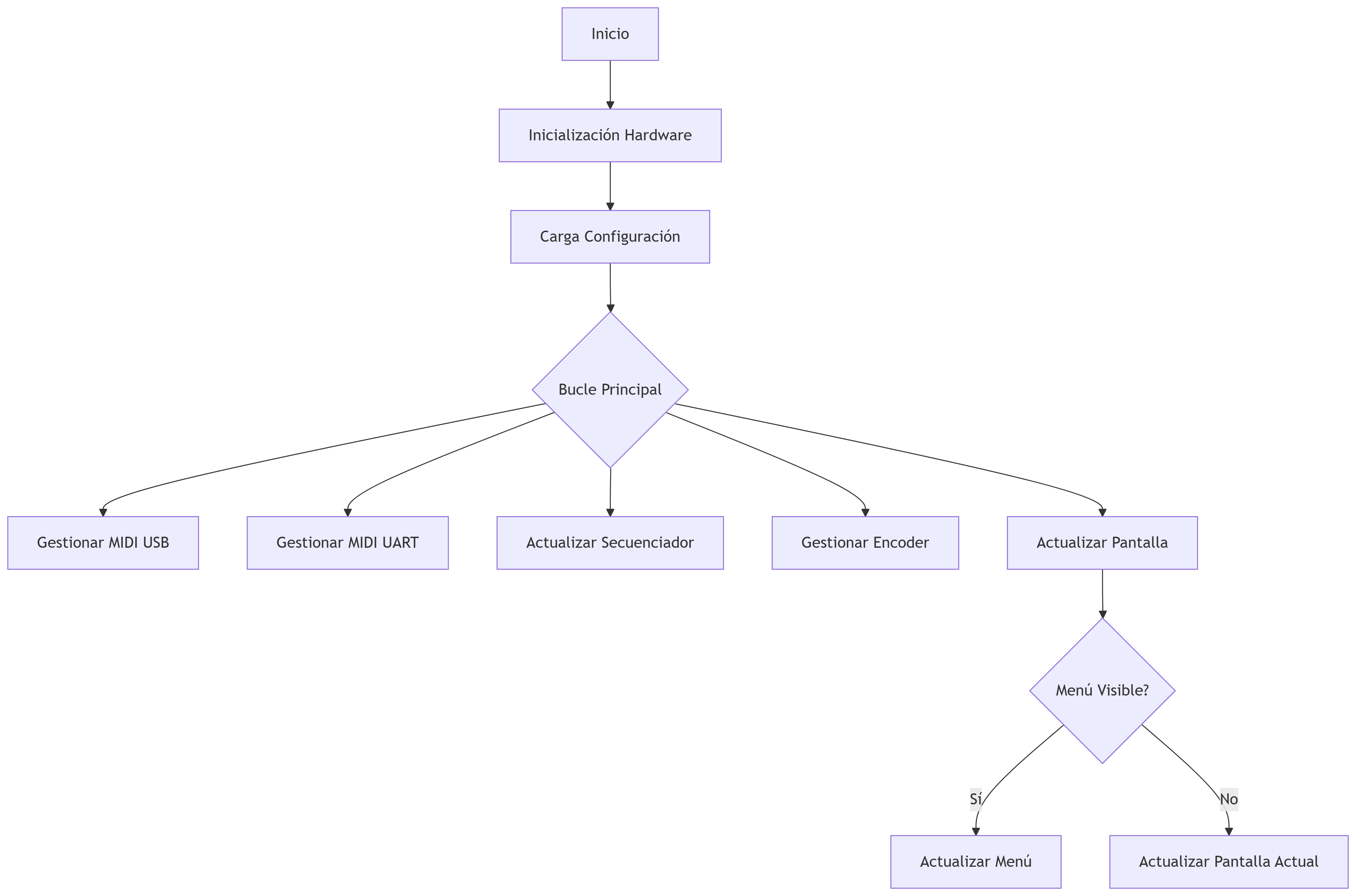


### 9.2 Responsabilidades por Módulo

| **Módulo** | **Responsabilidades Clave** |
| --- | --- |
| pantalla\_navegacion | Gestión de transiciones, eventos encoder, timeout |
| menu\_unico | Lógica de navegación, gestión de estados |
| pantalla\_unica | Renderizado OLED, manejo de colores |
| secuenciador | Reproducción de patrones, sincronización MIDI |
| controles | Mapeo MIDI, gestión de presets |
| configuracion | Variables globales, gestión de EEPROM |
| zona\_menu | Definición de áreas interactivas |

## 10. Flujos del Sistema

### 10.1 Flujo Principal de Ejecución



### 10.2 Flujo de Comunicación MIDI

## 11. Compilación y Uso

### 11.1 Requisitos de Compilación

* **Plataforma**: PlatformIO (recomendado) o Arduino IDE
* **Librerías**:
  + Adafruit\_SSD1351@^1.2.6
  + Adafruit\_TinyUSB@^1.10.0
  + SPI@^1.0
  + SD@^1.0.5

### 11.2 Configuración de Hardware

cpp

*// hardware\_config.h*

#define OLED\_CS 5

#define OLED\_DC 15

#define OLED\_RST 16

#define ENCODER\_A\_PIN 9

#define ENCODER\_B\_PIN 10

#define ENCODER\_BUTTON\_PIN 13

#define SD\_CS 8

#define MIDI\_UART\_TX 43

#define MIDI\_UART\_RX 44

### 11.3 Instrucciones de Uso

1. **Inicialización**:
   * Insertar tarjeta SD con estructura de presets
   * Conectar MIDI IN/OUT
   * Alimentar el sistema
2. **Navegación básica**:
   * Girar encoder para desplazarse por opciones
   * Pulsación corta para seleccionar
   * Pulsación larga para cambiar zona activa
3. **Gestión de presets**:
   * Acceder a menú PRESETS
   * Seleccionar LOAD/SAVE/DELETE
   * Elegir preset de la lista

## 12. Futuras Ampliaciones (Roadmap v0.7.0)

* **Integración con DAWs**:
  + Soporte completo para transporte (START/STOP/CONTINUE)
  + Sincronización visual de compás
  + Integración con Ableton Link
* **Mejoras de Secuenciador**:
  + Guardado de parámetros globales en presets
  + Modificación de ticksPorStep desde menú
  + Patrones de ritmo predefinidos
* **Optimizaciones**:
  + Carga diferida de presets grandes
  + Soporte para fuentes de texto personalizadas
  + Transiciones animadas entre pantallas
* **Expansión Hardware**:
  + Soporte para pantalla secundaria
  + Integración con módulo WiFi
  + Control remoto vía aplicación móvil

## 13. Problemas Conocidos

| **Issue** | **Gravedad** | **Solución Temporal** |
| --- | --- | --- |
| Límite de 108 controles | Media | Expandir multiplexación en futura versión |
| Retardo en carga de presets grandes | Baja | Usar presets más pequeños |
| Texto largo en OLED | Baja | Acortar etiquetas en menús |
| Ocasionales glitches visuales | Baja | Reiniciar sistema |

## 14. Licencia y Contribuciones

**Licencia MIT** - Libre para uso personal y comercial con atribución.

**Contribuciones**:

1. Reportar issues en GitHub
2. Proponer mejoras mediante pull requests
3. Documentar casos de uso
4. Desarrollar nuevos presets

**Versión del Documento**: 1.1  
**Última Actualización**: 16 de Agosto, 2025  
**Firmware Compatible**: v0.6.1  
**Autores**: Equipo de Desarrollo SP1 Mk2