Kyber Controls Manual V3

RC Robotic Control System

Kyber Controls

Contents

Table of Contents

Introduction

The Kyber Controls System is a comprehensive RC robotic control system that provides advanced sound playback, servo control, and wireless configuration capabilities. This manual covers firmware version 150 and later.

Key Features

- Support for up to 45 configurable buttons (3 pads of 15 buttons each)
- 24-channel SBUS support (SBUS16 and SBUS24)
- Dual Maestro servo controller support (up to 48 servos)
- WiFi configuration interface with modern responsive design
- Random sounds and animations
- Marcduino integration
- Emergency stop functionality
- Real-time configuration without rebooting

System Requirements

- ESP32-based Kyber main board
- Compatible RC receiver with SBUS output
- DFPlayer Mini or compatible MP3 player module
- SD card with MP3 files
- 7.5V to 36V power supply
- Optional: Pololu Maestro servo controllers

Main Board Layout

Main Board Components:

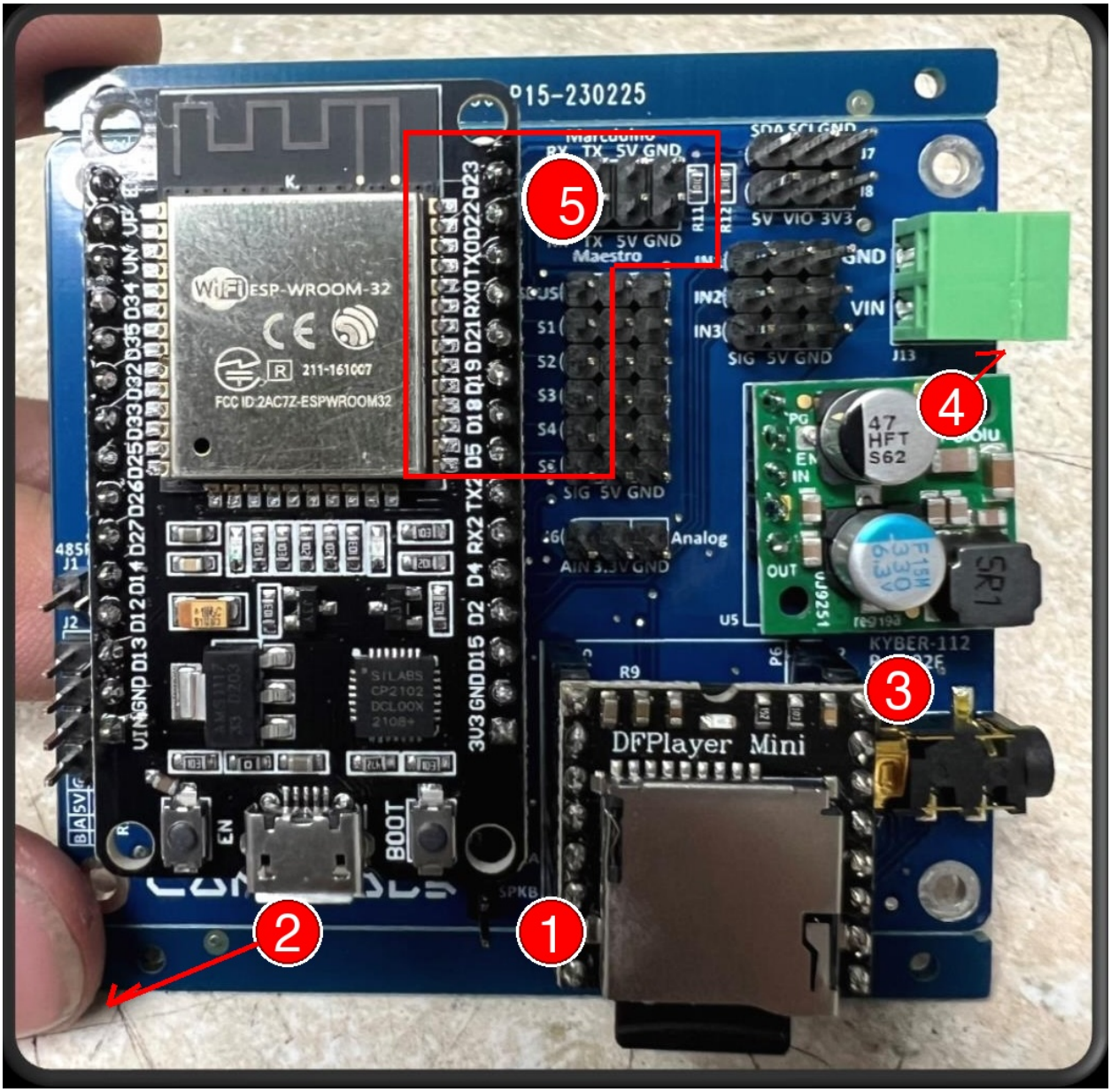


Figure 1: Main Board Layout

1. **SD Card Slot** - Insert SD card with MP3 files
2. **Speaker Output** - 3W maximum (terminal closest to DFPlayer is negative)
   • Terminal closest to the DF Player is the negative output to the speaker
3. **3.5mm Audio Jack** - For external amplifier connection
   • If you experience noise in the system you may need to install a ground loop isolator between the Kyber and the amplifier
4. **Power Input** - 7.5V to 36V DC

    - Input voltage: 7.5V to 36V
5. **Kyber I/O Connections** - 5V output, 2.5A maximum current
    - Output voltage: 5V
    - **WARNING**: 2.5A Max Output Current

Note: The red box highlights the Marcduino/Maestro connection area with TX, RX, and GND pins

Important Warnings:

- Maximum speaker power: 3 Watts
- Maximum output current: 2.5 Amps
- Use ground loop isolator if experiencing audio noise
- Never exceed voltage ratings

Wiring Guides

Basic SBUS Connection

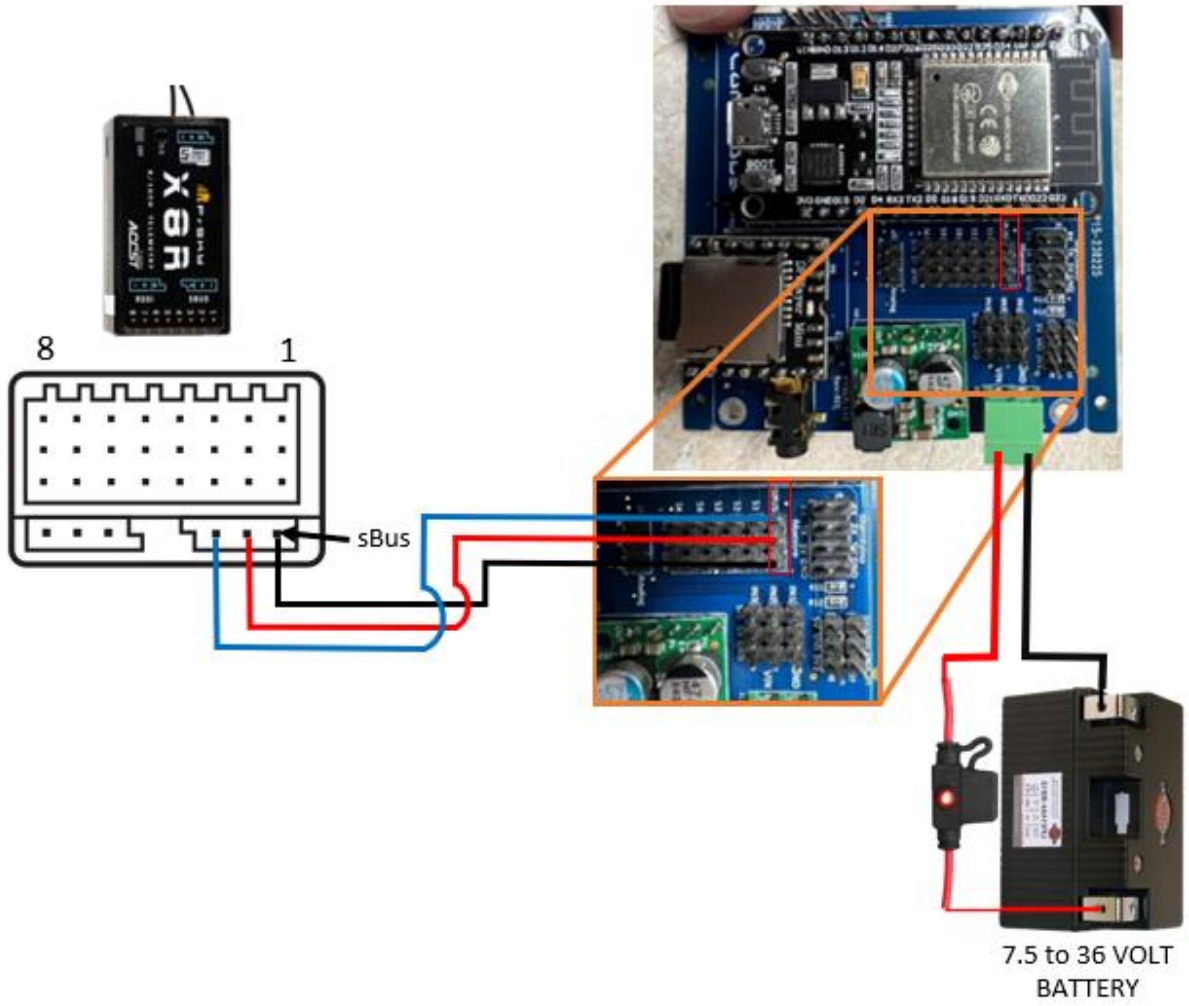


Figure 2: Basic SBUS Wiring

Connect the following: - Receiver SBUS output Kyber SBUS input - Power supply Kyber power input - Speaker or amplifier Audio output

Complete System with Maestros

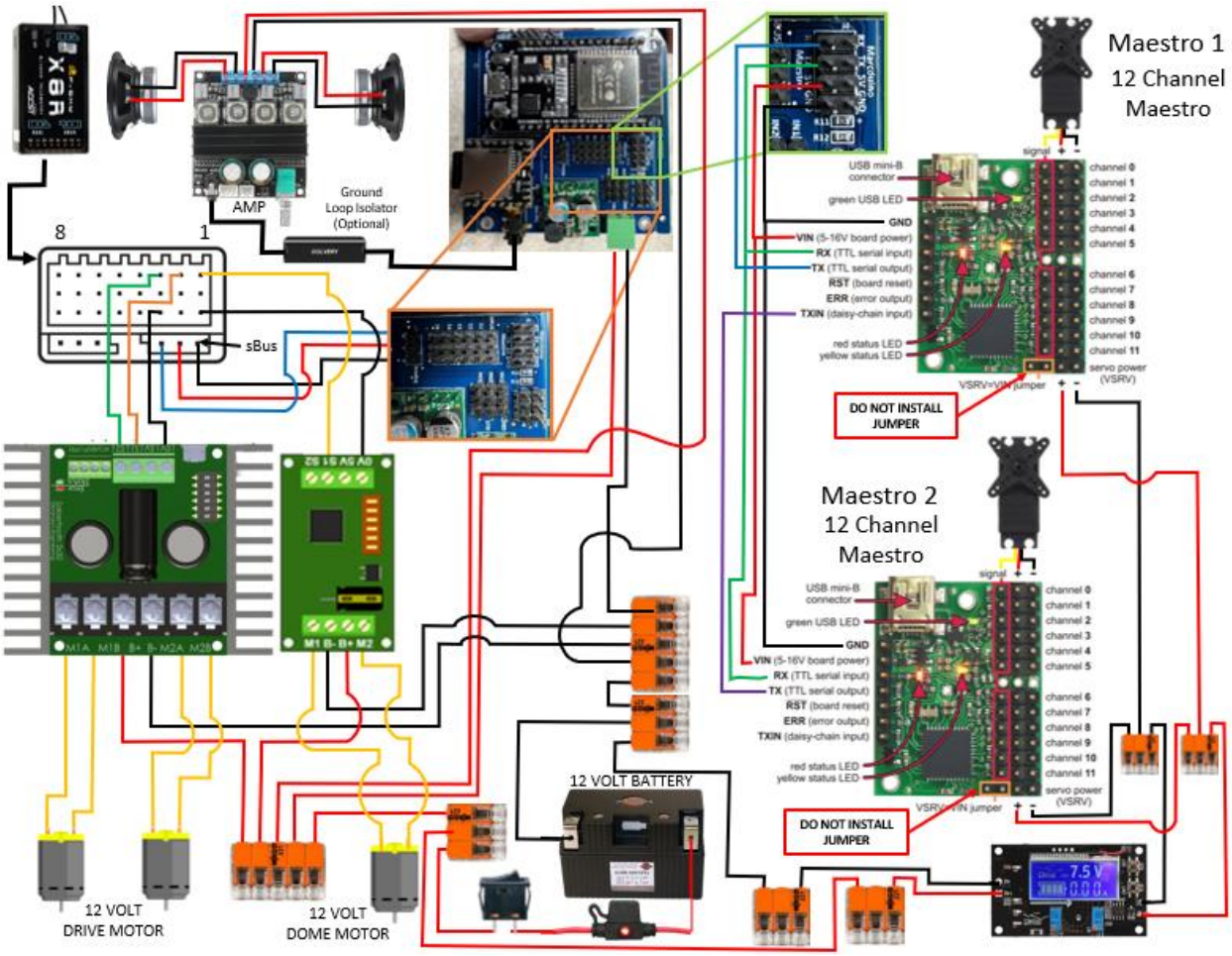


Figure 3: Dual Maestro Wiring

This configuration includes: - SBUS receiver connection - Dual Maestro connections - Audio amplifier - Emergency stop switch (optional) - Drive motor controllers

Connection Notes:

- Ensure proper polarity on all connections
- Use appropriate wire gauge for current requirements
- Keep signal wires away from power wires to reduce interference

---

Sound Files Setup

File Requirements:

- **Format**: MP3 only

- **Naming**: 0001.mp3 through 0255.mp3
- **Location**: Must be in /MP3/ folder on SD card
- **Optional naming**: Files can include descriptive text after the number (e.g., "0001_startup.mp3")

SD Card Structure:

```
SD Card Root/
  MP3/
    – 0001.mp3
    – 0002.mp3
    – 0003.mp3
    – ... up to 0255.mp3
```

Tips:

- Keep a backup of your sound files on your computer
- Create a spreadsheet documenting which sound is which number
- Test playback before final installation
- Use consistent volume levels across all files

Maestro Setup

Maestro Configuration Steps:

For Maestro 1:

1. Connect Maestro to computer via USB
2. Open Pololu Maestro Control Center
3. Navigate to Serial Settings tab
4. Select "UART, fixed baud rate"
5. Enter baud rate: **57692**
6. Set Device Number: **1**
7. **Uncheck** "Enable CRC" and "Never Sleep"
8. Apply settings

For Maestro 2:

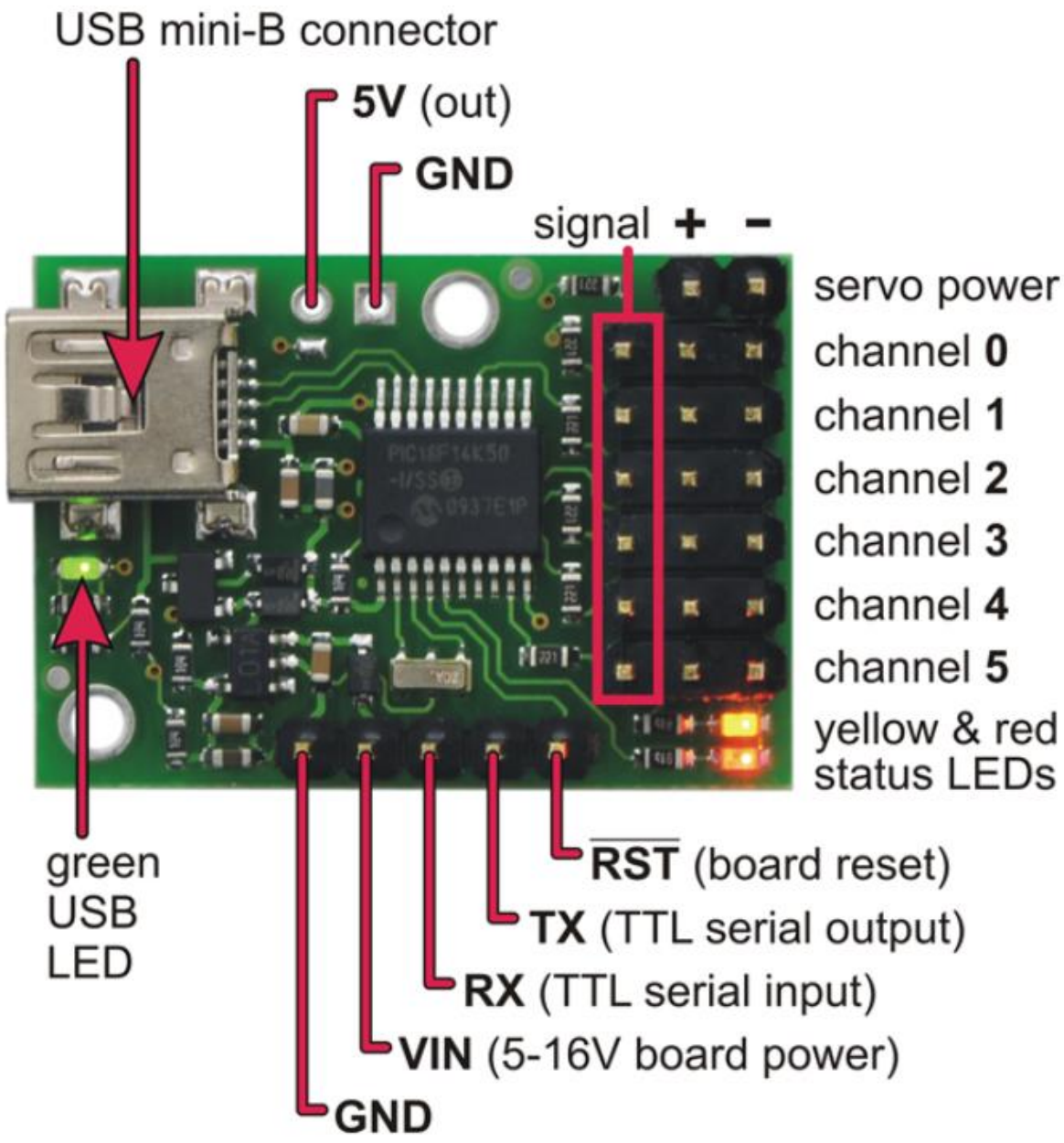
1. Follow same steps as Maestro 1
2. Set Device Number: **2**
3. Apply settings

Supported Maestro Models:

- 6-channel Micro Maestro
- 12-channel Mini Maestro
- 18-channel Mini Maestro

- 24-channel Mini Maestro

Maestro Pinout References:



Micro Maestro 6-channel USB servo controller (fully assembled) labeled top view.

Figure 4: 6-Channel Maestro

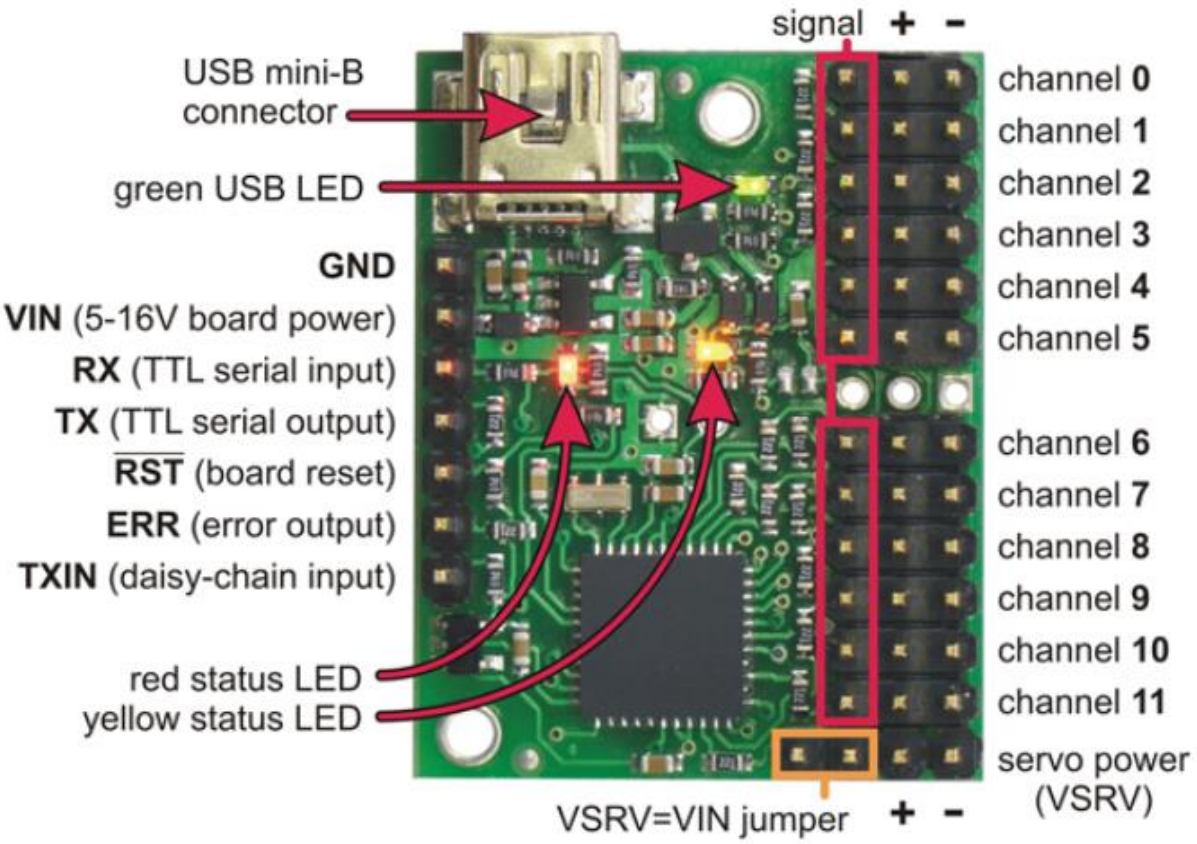


Figure 5: 12-Channel Maestro

For 18 and 24-channel pinouts, refer to the Pololu documentation.

Quick Start Guide

Initial Setup:

1. **Connect Hardware**
   - Connect SBUS from receiver to Kyber
   - Connect speaker or amplifier
   - Connect Maestros (if used)

- Apply power to Kyber board

2. **Connect to WiFi**
   - On your device, find WiFi network “KYBER_[MAC]”
   - Default password: 12345678
   - Open web browser
   - Navigate to: http://192.168.4.1
3. **Configure RC Settings**
   - Click “RC Settings” tab
   - Enter channel numbers for:
     - Button Pad (usually channel 9)
     - Toggle switches
     - WiFi on/off
     - Volume control
4. **Configure General Settings**
   - Click “General” tab
   - Set number of buttons (0, 15, 30, or 45)
   - Set button pad mode (2 or 3-position)
   - Set number of Maestros (0, 1, or 2)
   - Configure stop-all button
   - Save settings
5. **Configure Buttons**
   - Click “Button Configuration” tab
   - Select pad to configure (1, 2, or 3)
   - For each button, enter:
     - Description (optional)
     - Sound number (1-255)
     - Maestro scripts
     - Delays if needed
   - Save configuration

Web Interface Overview

Modern Interface Features:

- Responsive design works on phones, tablets, and computers
- Real-time updates without page refresh
- Streamlined navigation
- Visual feedback for all actions
- Auto-save reminders

Main Navigation Tabs:

- **Home** - System information and status
- **General** - Core system settings
- **RC Settings** - RC channel assignments and configuration
- **Button Configuration** - Unified button pad configuration

- **Buttons RC** - RC button channel configuration
- **WiFi** - Network settings
- **Firmware** - Updates and backup

Figure 6: Web Interface Home Page

General Settings

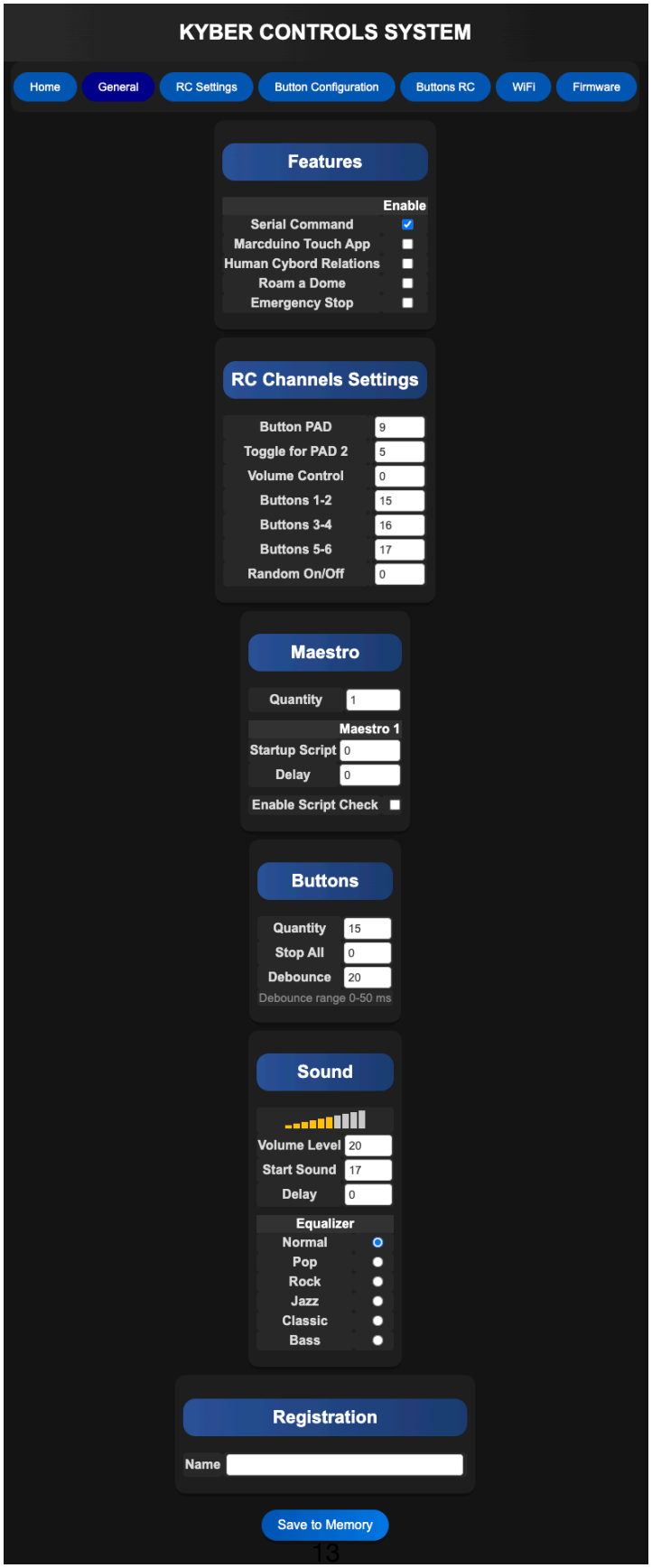


Figure 7: General Settings Page

Maestro Configuration:

- **Quantity**: 0, 1, or 2 Maestros
- **Startup Script**: Script to run on power-up
- **Startup Delay**: Delay before startup script (milliseconds)
- **Script Check**: Enable/disable checking if script is running

Button Configuration:

- **Button Pad Mode**:
  - 2-position (30 buttons total - Pads 1 & 2)
  - 3-position (45 buttons total - Pads 1, 2 & 3)
- **Number of Buttons**: Physical buttons installed (0-15)
- **Stop All Button**: Assign button for emergency stop
- **Debounce**: Adjust if experiencing double-triggers

Sound Features:

- **Volume Level**: Default volume (if not using RC channel)
- **Startup Sound**: Sound to play on boot
- **Startup Delay**: Delay before startup sound
- **Equalizer**: Choose audio profile (Normal, Pop, Rock, Jazz, Classic, Bass)

Additional Options:

- **Owner Name**: Your identification (shown in footer)
- **Marcduino Support**: Enable/disable Marcduino interface
- **E-Stop Support**: Configure emergency stop features

Button Configuration

Unified Configuration Page:

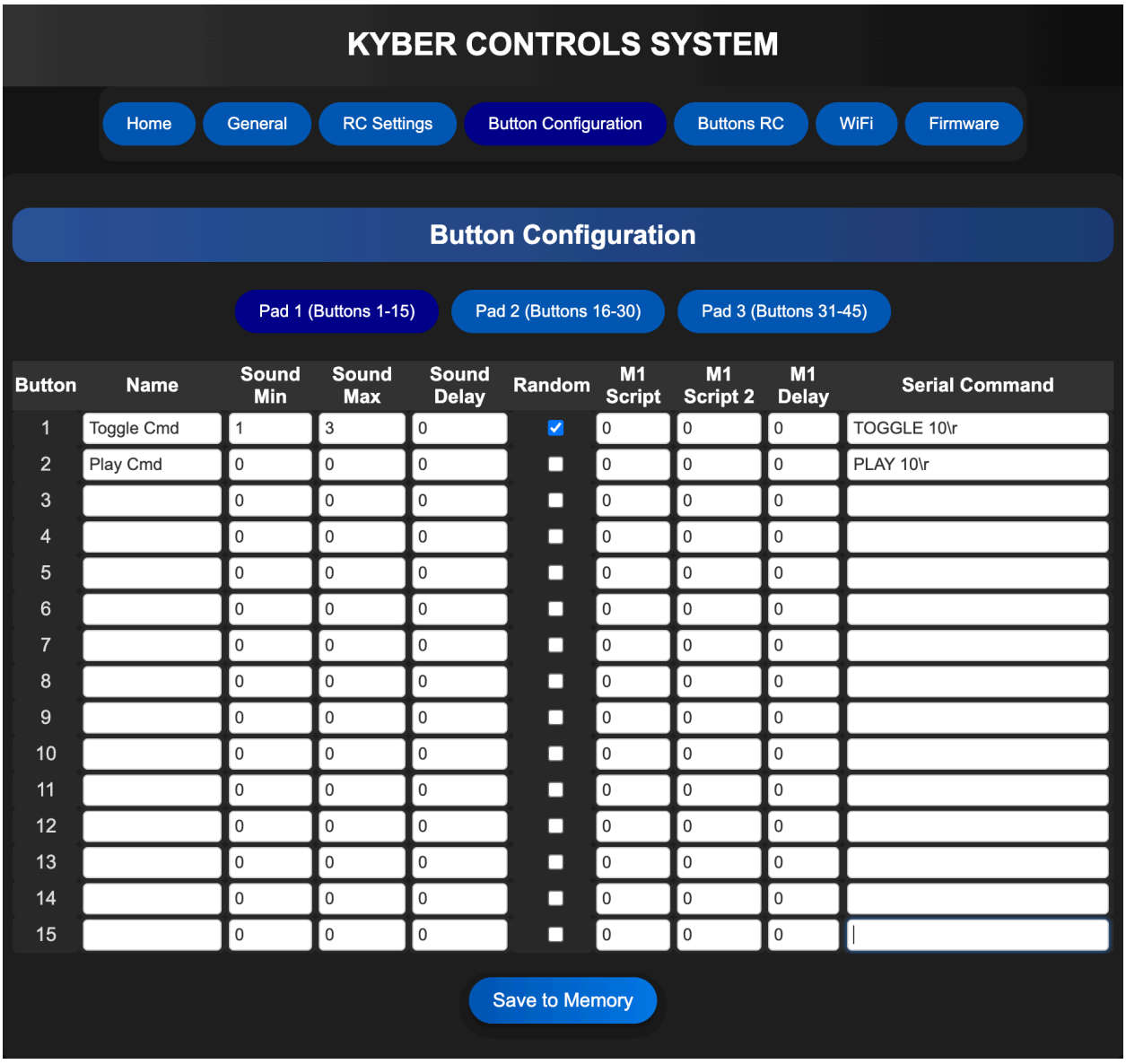


Figure 8: Button Configuration Page

The new button configuration uses a single page with a pad selector:

1. **Select Button Pad**:
   - Pad 1 (Buttons 1-15)
   - Pad 2 (Buttons 16-30)
   - Pad 3 (Buttons 31-45) - Only visible in 3-position mode
2. **For Each Button Configure**:
   - **Description**: Optional name for reference

- **Sound Min/Max**:
  - Same number = play single sound
  - Different numbers = cycle through range
  - Check “Random” for random selection
- **Sound Delay**: Milliseconds before playing
- **Maestro 1/2 Script**: Script numbers (1-100)
  - Min/Max for different scripts on press/release
- **Script Delays**: Timing adjustments
- **Marcduino Command**: Optional Marcduino commands

Button Behavior:

- Buttons can trigger on press and/or release
- Multiple actions per button (sound + scripts + Marcduino)
- Random selection from sound banks
- Sequential playback through ranges

RC Settings

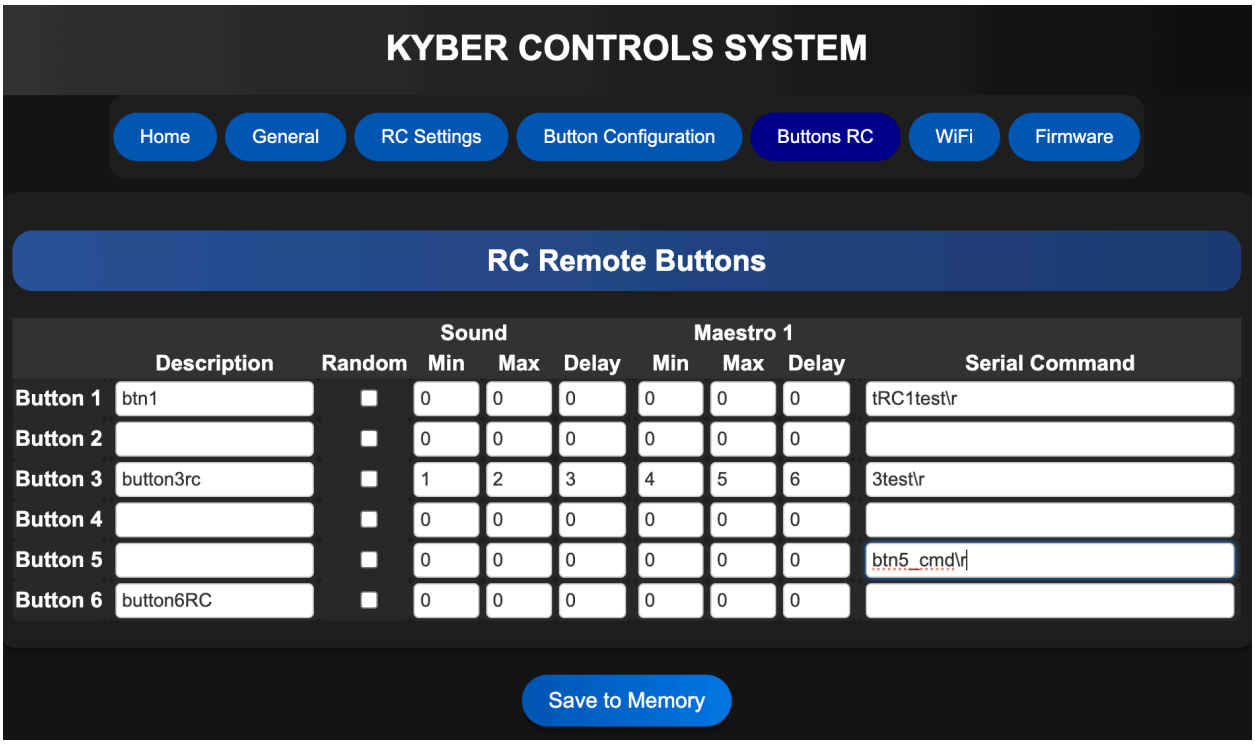


Figure 9: RC Settings Page

Channel Assignments:

Control Channels:

- **Button Pad**: Channel for button pad input
- **Toggle for Pad 2/3**: Switch between button banks
- **WiFi On/Off**: Hardware WiFi control
- **Volume Control**: Real-time volume adjustment

RC Button Channels (1-6):

- Convert 3-position switches to 6 buttons
- Each switch provides 2 button inputs
- Configure sounds and scripts like button pad

Passthrough Channels (24 available):

- **Description**: Name for reference
- **RC Channel**: Input channel (1-24)
- **Maestro ID**: Target Maestro (1 or 2)
- **Maestro Channel**: Servo channel
- **PWM Min/Max**: Range mapping
- **Disable Deadband**: Optional for precise control

Channel Smoothing:

- Adjustable smoothing per channel (0 = disabled)
- LERP function for smooth transitions
- Prevents servo jitter

Random Sounds & Animations

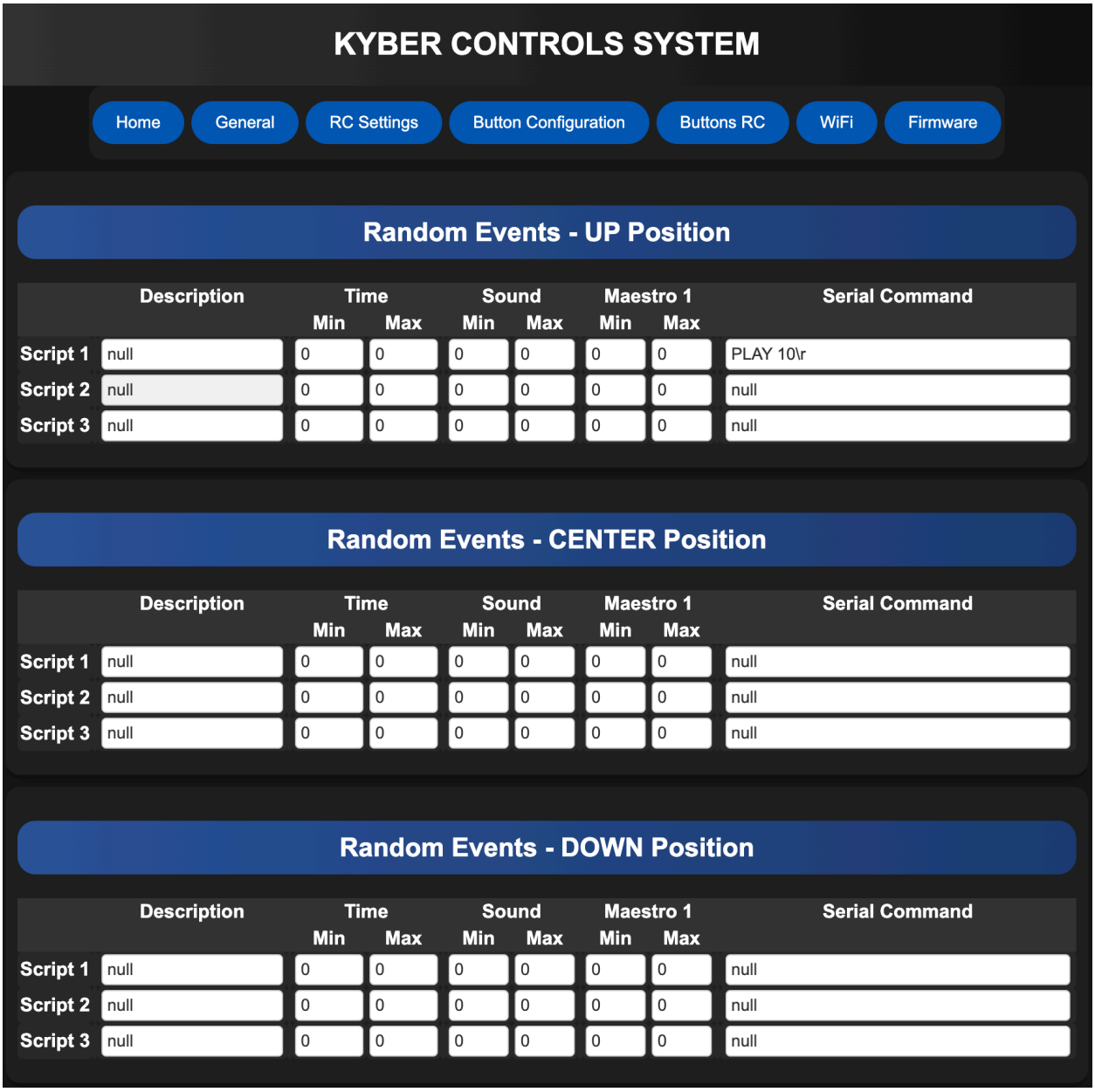


Figure 10: Random Configuration Page

Random Events Configuration:

- **Enable/Disable**: Master switch for random features
- **Min/Max Interval**: Time between random events (seconds)
- **Sound Groups**: Define banks of related sounds
- **Script Groups**: Define banks of related animations
- **Trigger Conditions**: When to allow random events

Script Builder:

- Visual script creation tool
- Combine multiple servo movements
- Set timing and sequences
- Test scripts in real-time
- Save up to 100 custom scripts

WiFi Settings

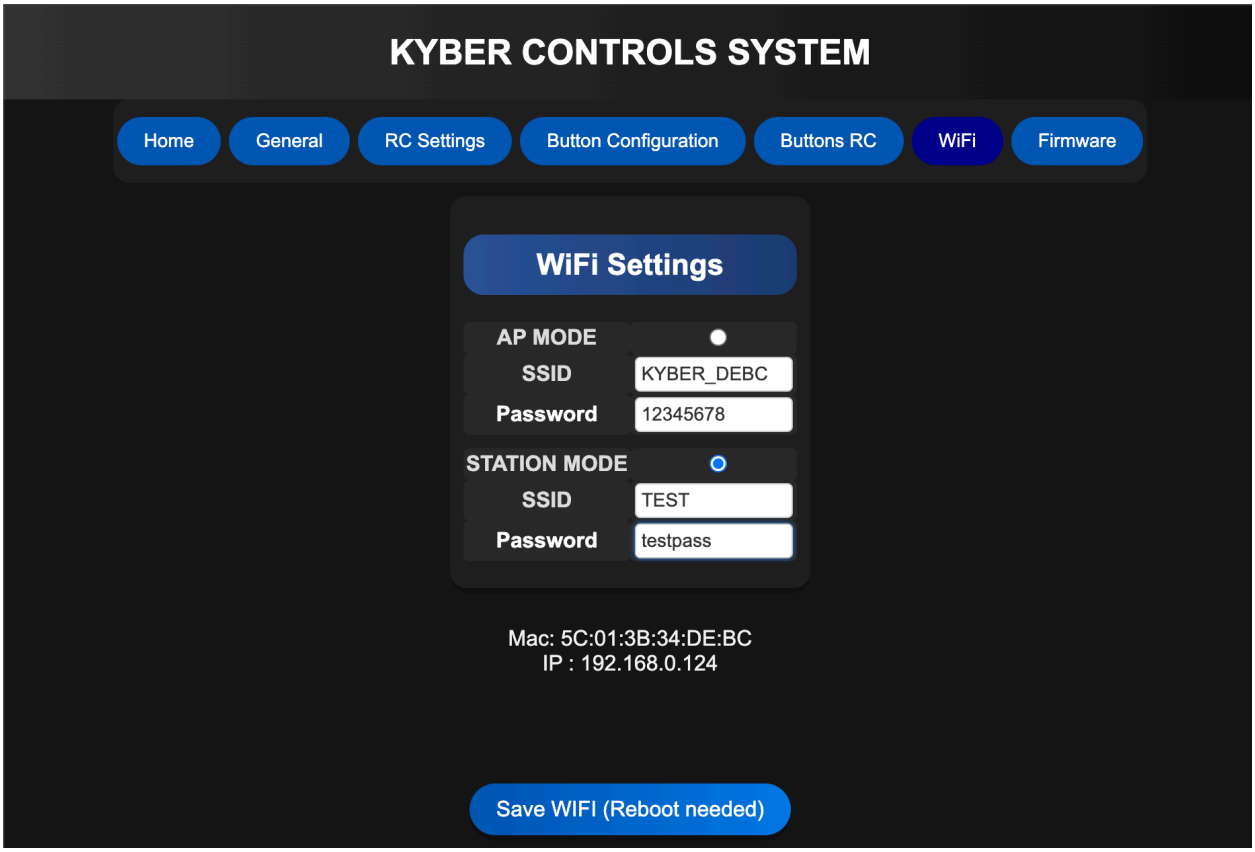


Figure 11: WiFi Settings Page

Access Point Mode (Default):

- **SSID**: Network name (includes MAC for uniqueness)
- **Password**: Network password (8+ characters)
- **IP Address**: Always 192.168.4.1

Station Mode (Optional):

- Connect Kyber to your home network

- **SSID**: Your network name
- **Password**: Your network password
- Automatically falls back to AP mode if network unavailable

Security Features:

- Hardware WiFi switch support
- Double-reset to clear credentials
- MAC address display for identification
- Encrypted configuration storage

Firmware Management

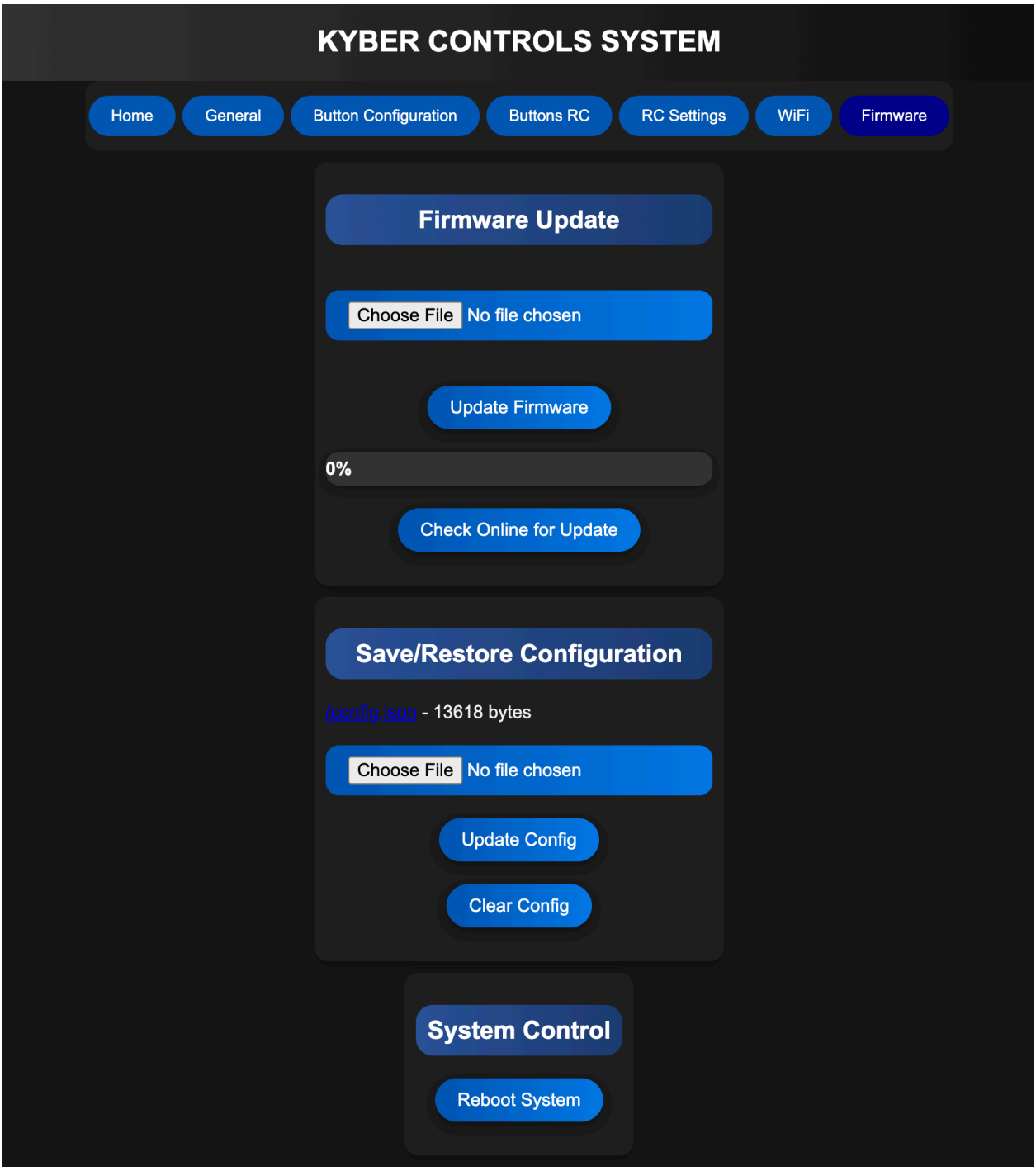


Figure 12: Firmware Management Page

Firmware Updates:

1. **Online Updates**:

- Click “Check for Updates”
- Automatic download and installation
- Progress bar shows status

2. **Manual Updates**:
   - Download firmware file from GitHub
   - Click “Select File”
   - Click “Upload”
   - Wait for completion (don’t disconnect!)

Configuration Backup:

- **Save Configuration**: Download config.json
- **Restore Configuration**: Upload saved config
- **Clear Configuration**: Factory reset
- **Reboot System**: Restart Kyber

Version Information:

- Current firmware version displayed
- Change log available online
- Automatic compatibility checking

Transmitter Setup Examples

FrSky X7/X9 Setup:

Figure 13: FrSky X7 with Button Pad

External Button Pad Installation:

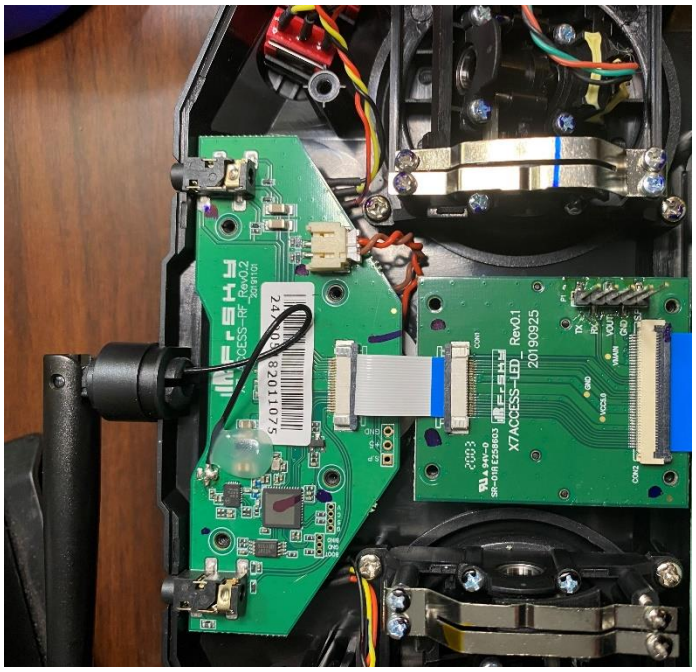


Figure 14: Button Pad Wiring

1. Mount 15-button pad to transmitter
2. Disconnect one potentiometer
3. Wire button pad to potentiometer connections:
    - Red to red (power)
    - Black to black (ground)
    - Signal to signal

Internal Custom Buttons:

Figure 15: Custom Button Board

1. Install momentary switches in desired locations
2. Wire to custom button board
3. Connect board to potentiometer
4. Configure button values in web interface

Channel Assignments Example:

- Channel 1-4: Stick controls
- Channel 5-8: Drive/dome controls
- Channel 9: Button pad
- Channel 10: Pad toggle switch
- Channel 11: WiFi on/off
- Channel 12: Volume control
- Channel 13-16: RC buttons

Advanced Features

Marcduino Integration:

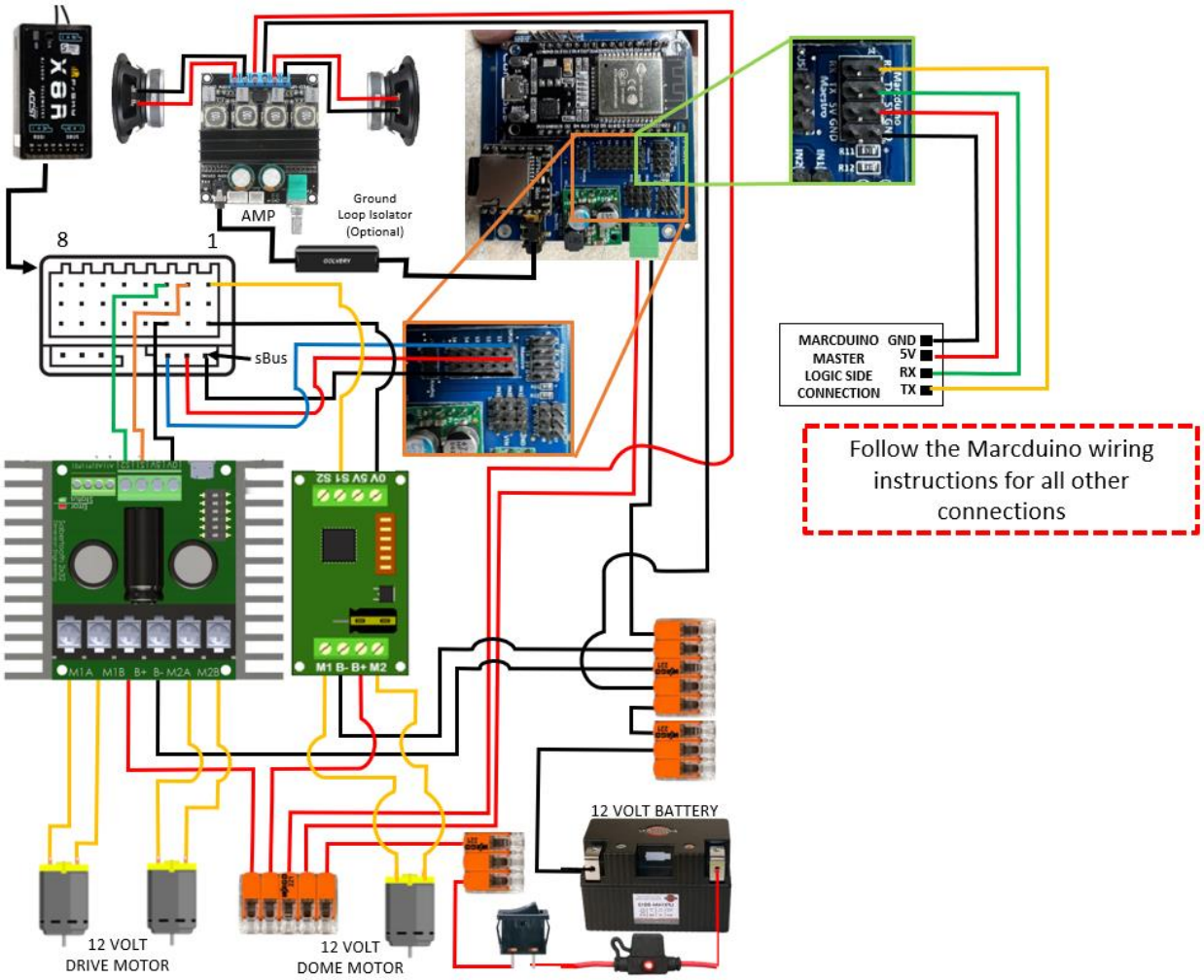


Figure 16: Marcduino Wiring

- Full command support
- WiFi app compatibility
- Vocalizer volume control
- Logic display commands
- Holoprojector sequences

Emergency Stop (E-Stop):

- Hardware bump switches
- Relay control for motors
- Configurable sounds
- Auto-reset timer
- Dual-switch safety

SBUS24 Support:

- Extended channel support
- Compatible with newer receivers
- Backward compatible with SBUS16
- Automatic detection

Vocalizer Support:

- Volume control via RC channel
- Text-to-speech commands
- Sound mixing capabilities
- Real-time adjustments

Troubleshooting

Common Issues:

No WiFi Network Visible:

- Check WiFi is enabled (hardware switch if installed)
- Double-reset to restore default SSID
- Verify power to board

Buttons Not Working:

- Verify button pad channel assignment
- Check button values in PWM page
- Confirm SBUS connection (green = connected)
- Adjust debounce if double-triggering

No Sound Playback:

- Check SD card is formatted FAT32
- Verify MP3 files in /MP3/ folder
- Confirm speaker connections
- Test with known working sound number

Maestro Not Responding:

- Verify baud rate (57692)
- Check device numbers (1 and 2)
- Confirm serial connections
- Test with Pololu software

Configuration Not Saving:

- Always click “Save to Memory”
- Wait for confirmation message
- Don’t navigate away during save
- Backup configuration regularly

Reset Procedures:

Soft Reset:

- Use web interface “Reboot” button
- Preserves all settings

WiFi Credential Reset:

- Double-press reset button quickly
- Restores default SSID and password

Factory Reset:

- Use web interface “Clear Config”
- Or hold reset for 10 seconds
- Returns to default settings

Support Resources:

- GitHub Issues: Report bugs and request features
- Facebook Group: Kyber Updates and community support
- YouTube Channel: Video tutorials and demos
- Email Support: [contact information]

Appendix A: Specifications

Electrical Specifications:

- Input Voltage: 7.5V - 36V DC
- Output Voltage: 5V
- Maximum Output Current: 2.5A
- Logic Level: 3.3V (ESP32)
- Communication: UART, I2C, SPI

Supported Protocols:

- SBUS (16 and 24 channel)
- Serial (57600/115200 baud)
- WiFi 802.11 b/g/n

- Marcduino commands

Memory Limits:

- Sounds: 255 maximum
- Scripts: 100 per Maestro
- Buttons: 45 configurable + 6 RC
- Passthrough: 24 channels

Appendix B: Glossary

- **SBUS**: Serial bus protocol for RC receivers
- **PWM**: Pulse Width Modulation for servo control
- **Maestro**: Pololu servo controller
- **DFPlayer**: MP3 playback module
- **Marcduino**: Droid control protocol
- **AP Mode**: Access Point mode for WiFi
- **Station Mode**: Client mode for WiFi
- **Debounce**: Delay to prevent false triggers
- **LERP**: Linear interpolation for smoothing

License

Manual Version 3.0 - Updated for Firmware V150 © 2024 Kyber Controls