SHOWDUINO FX CONTROLLER

Engineered for Fear. Built for Control.

**Introduction**

Welcome to the future of immersive scare control technology. The Showduino FX Controller is a powerful, precision-engineered device designed to orchestrate your haunt’s most intense, synchronized effects with theatrical precision and intuitive control. Whether you’re managing animatronic scares, lighting sequences, sound effects, or a combination of all three, the Showduino FX Controller is your command center.  
  
Built for haunted attractions, escape rooms, dark rides, immersive theater, and other interactive installations, the Showduino system empowers creators to trigger custom shows, control props remotely, and integrate effortlessly with industry-standard equipment — all from a compact touchscreen interface.

**Core Capabilities:**

* • Touchscreen Control Panel: A vivid, tactile interface allows fast interaction, menu navigation, and cue triggering without a computer.
* • Remote Operation: Compatible with the GoreFX Terminal, enabling wireless show triggering and monitoring.
* • Timeline Playback: Trigger complete show files with audio, lighting, and mechanical prop choreography from a single button press.
* • Expandable: Modular compatibility with Showduino add-ons like the Lantern Light Pulse, R3 Terminal, and future hardware modules.

**Intended Use:**

* • Haunted attractions and scare zones
* • Immersive theater and escape rooms
* • Live-action roleplay (LARP) environments
* • Special effects labs and interactive art exhibits
* • Museums, interactive installations, and seasonal theme parks

⚠️ Note: This controller is not a consumer toy. It is a professional-grade theatrical control system designed for trained operators or haunt engineers.

**Built by Haunters, for Haunters**

Showduino was conceived by FX professionals with years of field experience in haunted houses, escape rooms, and immersive live events. That’s why every function — from the tactile menu buttons to the rugged mounting system — reflects the chaotic, dimly lit, fog-choked reality of the show floor.  
  
Let’s bring your nightmare to life.

**Section 2: Safety & Warnings**

Safety isn't a suggestion — it's the foundation of fear done right.  
  
The Showduino FX Controller is an advanced show-control interface designed for immersive environments. Like all professional haunt-tech gear, it must be handled with care, precision, and respect. This section outlines essential safety practices, electrical limitations, and operational warnings to ensure reliable performance and user protection in live installations.

**⚠️ GENERAL WARNINGS**

* • READ THIS SECTION IN FULL BEFORE INSTALLATION OR USE.
* • For Professional Use Only: This is not a toy. The Showduino FX Controller is intended for use by trained theatrical technicians, haunt operators, or engineers with basic knowledge of electronics and live event safety standards.
* • Disconnect Before Servicing: Always remove power and isolate external inputs before opening enclosures or modifying connections.
* • Indoor Use Only (Unless Housed): Unless placed in an appropriate weatherproof housing, this unit should not be used in outdoor or moisture-prone environments.
* • Do Not Modify Internals: Opening or altering the controller can result in electric shock, void warranties, and damage critical systems.
* • Avoid Overload: Never connect devices that exceed the Showduino’s rated control capacity. Use isolated power and signal amplifiers for high-draw props.

**🔌 ELECTRICAL SAFETY**

The Showduino FX Controller is powered using an external 240V AC power controller. This unit steps down, regulates, and routes power safely to all necessary internal components and output systems.  
  
UNDER NO CIRCUMSTANCES SHOULD MAINS POWER BE CONNECTED DIRECTLY TO THE CONTROLLER.  
The internal circuits and GPIO systems are not rated for raw AC input and will be permanently damaged.

|  |  |  |
| --- | --- | --- |
| Component | Power Rating | Description |
| Main Power Input | 240V AC (via external controller) | Connect only through included or approved power distribution hardware. |
| Relay Triggers | 4 x 5V Relays + 4 x 12V Relays | Integrated, software-controlled, independently isolated. |
| GPIO Ports | Signal-level only | Use with digital trigger logic (not for power driving). |
| USB Port | 5V / Data Only | For data interface and potential firmware updates. |
| Audio Out | Line-level stereo | Use with powered speakers or amplifier only. |

**🧯 FIRE PREVENTION & LOAD MANAGEMENT**

* • Secure all connections: Avoid loose barrel jacks, terminal blocks, and unshielded wiring.
* • Use approved cables: Thin-gauge or underspecified cables may heat under load, risking fire or failure.
* • Keep airways clear: Never block or restrict airflow around the unit. Overheating can cause malfunction or failure.
* • Do not bypass fuses: The controller includes integrated fuse protection. Never short or replace with unapproved types.

**⚙️ MECHANICAL HANDLING & INSTALLATION**

* • Mount securely: Use the included mounting hardware to fasten the unit in a stable, protected location — ideally within a rack, cabinet, or backstage control box.
* • Protect the touchscreen: Avoid pressing with sharp objects or excessive force. Only use a clean finger or capacitive stylus.
* • Prevent vibration: Secure cabling and avoid high-vibration areas that may loosen or disconnect wiring during operation.

**🚫 PROHIBITED PRACTICES**

* • Connect the unit directly to AC wall power.
* • Operate in wet, humid, or unsealed outdoor areas without proper IP-rated housing.
* • Power mechanical props directly through GPIO.
* • Bypass relays or rewire internal boards.
* • Operate the controller in environments exceeding 104°F (40°C) or below 32°F (0°C).
* • Use unverified firmware or unlicensed code on the unit.

**✅ RECOMMENDED PRACTICES**

* • Label all wiring clearly to streamline setup and teardown.
* • Use strain reliefs for all external connections to avoid tugging or pull-outs.
* • Check connections periodically during the season for corrosion or fatigue.
* • Test all cue paths during tech rehearsals using the "Dry Run" feature in the Showduino UI. dry run feature can also be a way to control showduino like a master behind it all
* • Use GoreFX or compatible wireless terminals only after full pairing confirmation.

**💀 SAFETY MANTRA: BUILT FOR FEAR, NOT FAILURE**

“No haunt ever got scarier with a fire hazard backstage. Build it once. Build it safe. Let the ghosts do the killing, not your gear.”  
— The Showduino Engineering Team

**Section 3: Box Contents**

Everything you need to summon the fear—right out of the box.  
  
When you open the box containing your Showduino FX Controller, you’re not just unpacking hardware. You’re unsealing a theatrical arsenal crafted for chaos, built for precision, and ready for showtime.  
  
This section details all components included in a standard shipment of the Showduino FX Controller. Confirm that all items are present before beginning installation. If any parts are missing, contact support immediately (details in the Contact & Support section).

**📦 Standard Package Includes**

|  |  |
| --- | --- |
| Item | Description |
| Showduino FX Controller | Primary control unit with touchscreen, audio, relay ports, and GPIOs. |
| USB-C Cable | For connecting to a computer for updates or direct control when needed. |
| Mounting Bracket (x1) | Durable steel or aluminum bracket with screw slots for stable installation. |
| Mounting Screws (x4) | Hardware to affix the bracket and controller to a wall or panel. |
| Quick Start Card | Laminated card with setup checklist, QR to manual, and key diagrams. |
| Relay Breakout Connector | Pluggable screw terminal or harness adapter for easy relay wiring. |
| Power Controller Unit | 240V AC external unit designed to regulate and deliver power safely. |
| Label Sheet | Pre-printed wiring and port labels for efficient cable identification. |
| Protective Face Film | Touchscreen is shipped with removable protective film to avoid scratches. |
| Showduino Sticker | High-gloss logo decal for gear cases, toolboxes, or your creepy lair. |

**📎 Optional or Add-On Accessories**

|  |  |
| --- | --- |
| Item | Description |
| GoreFX Terminal | Wireless control module for remote prop triggering (WiFi/Bluetooth). |
| Showduino Lantern | RGB LED beacon light add-on that syncs with FX sequences. |
| R3 Terminal | Compact external keypad for live cue triggering away from the main unit. |
| Showduino Expansion Harness | Adds extra I/O support for complex prop integrations. |
| DMX Out Adapter | Converts Showduino relay output to DMX512 signal chain control. |
| Weatherproof Enclosure | Rated IP65, suitable for outdoor or fog-heavy installations. |

📝 Note: Optional items must be purchased separately unless bundled through a promotion or dealer.

**🔍 Inspection Checklist**

* ✅ Controller shows no cracks, warping, or screen damage.
* ✅ Relay breakout connector matches relay pinout labels (5V/12V relays separated).
* ✅ USB cable is free of frays, cuts, or exposed wire.
* ✅ Mounting hardware is accounted for and not stripped or bent.
* ✅ Power controller has correct regional plug (U.S., EU, U.K., etc.).
* ✅ No rattling or loose parts when the unit is gently shaken.

**🧰 Tools You May Need (Not Included)**

* • #1 or #2 Phillips screwdriver
* • Wire strippers and cutters
* • Label maker or sharpie (for additional cable tags)
* • Zip ties or Velcro ties (for cable management)
* • Drill and anchors (for wall mounting in certain materials)

**📌 Shipping Note**

All Showduino FX Controllers are shipped from our fulfillment lab in padded, custom-fit foam to prevent jostling or shipping shock. // with great care // If your package arrives significantly damaged (bent corners, crushed box, open seams), photograph the box before opening and contact customer service for immediate assistance.

**Section 4: Device Overview**

Your haunt's nerve center — decoded.  
  
The Showduino FX Controller is more than a black box of frights. It's a smart, structured interface that puts prop activation, show timing, and remote cue execution all within a few taps. Understanding each feature of the device is key to maximizing its potential — and preventing accidental misfires during peak screams.  
  
This section walks you through each physical feature, port, and indicator so you know exactly what’s at your fingertips — and behind them.

**🔲 Front Panel Overview**

|  |  |
| --- | --- |
| Feature | Description |
| Touchscreen Display | Capacitive touchscreen used for system control, show selection, status monitoring, and manual cueing. |
| Speaker Grille | Integrated front-facing speaker for system sounds or basic audio playback (non-amplified). |
| Status LED | Multicolor LED providing instant visual feedback on system status (power, show ready, error, etc.). |
| Show Control Button | Large, tactile button on the faceplate for manually triggering primary cues. Backlit and programmable. |
| Brand Plate | Metallic Showduino logo plate with serial ID printed below for registration and support reference. |

**🔌 Rear Panel Overview**

|  |  |  |
| --- | --- | --- |
| Port/Component | Label | Function |
| Power Input | POWER IN | Connects to the 240V external power controller (barrel or screw terminal depending on model). |
| USB Port | USB-C | For software updates, serial debugging, or USB-triggered control. |
| Relay Output Bank | RELAY OUT | 8 terminal relays: 4 at 5V and 4 at 12V, addressable via interface or timeline JSON. |
| GPIO Ports | GPIO A/B | General purpose digital input/output ports (for triggers, sensors, etc.). |
| Audio Out | LINE OUT | 3.5mm stereo jack to connect to powered speakers or an audio amp. |
| Serial Ports | SERIAL 1/2 | Used for sending prop commands to external Arduinos, MP3 players, or DMX controllers. |
| WiFi/Bluetooth Module Slot | WIRELESS | Optional insert module slot for GoreFX remote or other terminal integration. |
| SD Card Slot | SHOWCARD | Houses microSD card storing timeline JSON files and system config. |
| Fan Vent / Passive Cooling | n/a | Rear-mounted vent for natural heat dissipation — ensure unobstructed airflow. |

**🧠 Internal Systems Summary**

Though not visible externally, the Showduino’s internal architecture is optimized for responsive performance, low-latency prop triggering, and modular expansion.  
- Core Processor: Custom-tuned microcontroller built for real-time I/O sequencing.  
- Memory Storage: Supports FAT32-formatted microSD cards up to 32GB.  
- Relay Logic Isolation: Fully separate logic per relay bank (5V/12V), improving stability and protection.  
- Redundant Power Logic: Intelligent shutdown and power correction prevents lock-ups or accidental resets.

**🔵 Status LED Indicator Guide**

|  |  |  |
| --- | --- | --- |
| LED Color | Behavior | Meaning |
| Green | Solid | Powered and idle — ready for trigger or show selection. |
| Blue | Blinking | Connecting or transmitting to GoreFX terminal. |
| White | Solid | Show currently playing or timeline in progress. |
| Red | Blinking Fast | System error or relay fault detected. |
| Red | Solid | Critical error — power cycle required. |
| Yellow | Pulsing | USB-connected or firmware update in progress. |

**🧭 Device Dimensions & Mounting**

- Width: 6.25 inches  
- Height: 4.75 inches  
- Depth: 2.5 inches (without bracket)  
- Weight: 1.6 lbs // to be added upon completion  
  
Mount Points: Four M4-threaded inserts allow permanent wall or panel mount with included bracket.  
Touchscreen Surface: 4.3” resistive display (glove-compatible).

**🎛️ Labeling & Diagnostics**

Each Showduino FX Controller is shipped with a factory label on the rear or underside, showing:  
- Serial number (for firmware and support reference)  
- Production batch  
- Voltage range tolerance  
- QR code to download the latest firmware, JSON templates, and UI guide

**Section 5: Installation & Mounting**

From bench to scare-zone — install it like a pro.  
  
Proper installation of your Showduino FX Controller ensures stability, optimal performance, and longevity. Whether it’s flush-mounted into a control panel, rack-mounted backstage, or wall-mounted in your haunt’s control room, the following instructions will help you secure your device with confidence and theatrical-grade precision.  
  
This section walks you through step-by-step instructions to safely and efficiently mount the device, with important environmental and safety considerations.

**🛠️ Pre-Installation Checklist**

* ✅ Read the Safety & Warnings section thoroughly.
* ✅ Verified that all parts from the Box Contents section are present.
* ✅ Chosen a well-ventilated, dry, and vibration-free mounting location.
* ✅ Identified nearby access to power, signal cables, and peripheral devices.
* ✅ Reviewed all relay wiring diagrams and load paths for safety compliance.

**📍 Mounting Options**

You can install the Showduino FX Controller in a variety of configurations:

1. Wall Mount (Standard)  
- Use the included metal bracket and M4 mounting screws.  
- Suitable for flat vertical surfaces — control rooms, booths, backstage zones.

2. Panel Flush Mount  
- Cut a custom opening into a plywood, metal, or ABS panel.  
- Secure from the rear using bracket arms or panel frame screws.

3. Rack Mount (With Adapter)  
- Optional 1U or 2U rack adapter available for 19” standard racks.  
- Ideal for centralized tech control setups or AV closets.

**🔧 Wall Mount Installation Steps**

* • Mark the Mounting Points: Using the bracket as a stencil, mark four mounting holes with a pencil or punch.
* • Drill Pilot Holes: Drill using a bit suitable for your surface (wood, drywall, metal, etc.). For drywall, use anchors.
* • Secure the Bracket: Align and screw the mounting bracket firmly into place.
* • Attach the Controller: Slide the Showduino into place. It should snap-lock or seat into the bracket, depending on the model.
* • Confirm Firmness: Tug gently — the device should not wobble, slide, or rattle. If it moves, re-tighten screws or adjust alignment.

**🧱 Surface Recommendations**

|  |  |  |
| --- | --- | --- |
| Surface Type | Recommended? | Notes |
| Drywall | Yes | Use anchors or toggle bolts to ensure long-term grip. |
| MDF/Wood Panel | Yes | Pre-drill to prevent splitting; use wood screws. |
| Concrete | Yes | Use hammer drill and masonry anchors. |
| Plastic Paneling | Yes (if thick) | Confirm the panel is strong enough to hold 1.6 lb load. |
| Foam Walls | No | Not strong enough for direct mounting. Use backing plates. |
| Metal Surfaces | Yes | Ideal for control enclosures — use sheet metal screws. |

**🔌 Cable Routing Best Practices**

* • Route power, audio, and data cables away from each other to minimize interference.
* • Use strain relief clips or grommets to prevent cable stress at entry points.
* • Leave service loops (extra slack) on cables to allow for repositioning and maintenance.
* • Label all terminal ends before final installation for easy debugging later.

**⚠️ Mounting Safety Tips**

* • Don’t over-tighten screws — may warp or damage casing.
* • Avoid pinching wires behind the unit — this can lead to shorts or loose connections.
* • Ensure the unit is not flush with insulation, foam, or soft prop materials that could retain heat.
* • If you plan to mount near foggers or haze machines, enclose unit in a vented box to avoid fluid ingress.

**📐 Bracket Specifications // recomendations**

|  |  |
| --- | --- |
| Feature | Spec |
| Material | Powder-coated steel or anodized aluminum |
| Mounting Hole Spacing | 4" horizontally, 3.25" vertically (center to center) |
| Bracket Type | Snap-in shelf with optional backstop lip |
| Screws Included | 4x M4 hex-head or Phillips head |

**🧰 Installation Tools Required (Not Included)**

* • Power drill
* • Screwdriver (Phillips or hex depending on bracket screws)
* • Level or laser line (optional but recommended)
* • Wire snips and crimpers (if pre-wiring during mount)
* • Pencil, tape, or markers

**🔍 Post-Installation Inspection**

* ✅ Bracket is flush and tightly secured
* ✅ Showduino sits flat and aligned
* ✅ All cables are cleanly routed and labeled
* ✅ Touchscreen and LED indicators are visible and unobstructed
* ✅ External relays, props, or sensors are connected (or prepped for next step)

**💡 Pro Tip from Field Techs**

“Install it like it’s opening night — because one day, it will be. Get it right the first time, and the gear will do the rest.”  
— Haunt Operator’s Guide, 2023 Edition

**Section 6: Power Supply & Wiring**

The voltage behind the terror.  
  
The Showduino FX Controller is powered through a robust electrical design meant for live environments with complex prop networks and cue systems. To avoid performance issues, electrical faults, or permanent damage, it is crucial to follow proper power configuration and wiring practices.  
  
This section outlines the correct use of the 240V external power controller, details the relay outputs, GPIO terminal usage, and explains how to wire the unit for both simple and multi-prop setups.

**⚡ Primary Power Input**

The Showduino is powered via an external 240V AC power controller unit. This controller is responsible for:

- Converting AC mains power into safely regulated internal voltages

- Isolating high-voltage paths from low-voltage logic

- Providing protected power to relays, processor board, and peripherals

NEVER connect AC power directly into the Showduino unit.

|  |  |  |
| --- | --- | --- |
| Input Type | Rating | Notes |
| External Power Input | 240V AC | Plug into grounded outlet — do not daisy chain with foggers, lights, or amps. |
| Regulated Output | 5V / 12V rails | Internally distributed to relays and controller logic board. |
| Max Relay Load | 10A per bank | Distributed across 4x5V and 4x12V relays — total up to 80A max (theoretical). |
| Fuse Type | Inline blade fuse | Replace with same voltage/amperage as labeled (typically 15A). |

**🔌 Relay Output Terminals**

The Showduino features eight onboard relays, designed to trigger external devices such as lights, pneumatics, or sound effects.

Relay Configuration:  
- 4x 5V Relays  
- 4x 12V Relays  
- Optically isolated from logic  
- Individually addressable via touchscreen or JSON shows

|  |  |  |  |
| --- | --- | --- | --- |
| Relay Channel | Voltage | Terminals | Application Example |
| Relay 1–4 | 5V | NO / COM / NC | Trigger 5V solenoids or logic inputs |
| Relay 5–8 | 12V | NO / COM / NC | Activate 12V lights or relays |

✅ Relay terminals are pluggable screw terminals for fast swapping.

**📈 Relay Wiring Example**

To control a 12V strobe light when a scare is triggered:  
1. Wire the 12V strobe power (+) to a relay NO terminal (e.g., Relay 6).  
2. Connect 12V strobe ground (-) to the common ground rail.  
3. Connect the 12V relay’s COM terminal to the 12V power supply (+).  
4. Use the touchscreen or timeline to trigger Relay 6 — the strobe will flash.

**🧩 GPIO Ports**

|  |  |  |  |
| --- | --- | --- | --- |
| Port Bank | Pins | Mode | Notes |
| GPIO A | 6 pins | Input/Output | Configurable as pull-up/down inputs or digital outputs |
| GPIO B | 6 pins | Input only | Ideal for buttons, IR beams, or pressure mats |
| Voltage | 5V TTL | Logic only | Not intended to drive motors or solenoids directly |

**🧵 Wiring Guidelines**

* • Use 16–22 AWG wire for relay terminals.
* • Keep relay wiring under 10 feet if unshielded.
* • Use shielded cables for GPIO inputs in electrically noisy environments.
* • Always label wires at both ends for maintenance.

**💡 Tips for Clean Wiring**

* • Use color-coded wires (e.g., red for power, black for ground, yellow for signal).
* • Twist data and power pairs to reduce interference.
* • Secure wiring with cable ties or clips — especially in moving show sets.
* • For setups with multiple Showduino units, use cable sleeves or looms to organize bundles.

**🔌 Audio Output Connection**

The 3.5mm audio out provides line-level stereo signal. It must be connected to powered speakers or an amplifier. This output is perfect for voiceovers, jump-scares, or background ambiance triggered by shows.  
- Do not plug headphones directly into this port.  
- If wiring into a central audio system, use a ground-loop isolator.

**🧪 Voltage Testing Points**

The internal board includes labeled voltage test pads:  
- VCC (5V)  
- 12V rail  
- GND  
  
These can be used for:  
- Verifying incoming power  
- Diagnosing relay power drops  
- Ground loop troubleshooting  
  
🛑 Technicians only — testing live power points should only be done with appropriate tools and experience.

**🔄 Power Cycle Behavior**

Upon power-up:  
- Controller will auto-load the last-used show (if enabled)  
- All relays default to OFF state  
- GPIO pins initialize in safe input mode  
- LED indicators show green (ready state)  
  
If power is interrupted during a show:  
- A built-in safe-state timeout prevents stuck relays  
- You can enable "Auto Resume Last Show" from the UI settings (optional)

**⚠️ Wiring Do’s & Don’ts**

|  |  |
| --- | --- |
| Do This | Avoid This |
| Use labeled, fused power rails | Sharing power from fog machines or moving heads |
| Mount wires with slack | Pulling wires taut from terminal to terminal |
| Test relay circuits before show | Hot-plugging relay wires during performance |
| Separate power and data lines | Running them as one zip-tie bundle |

**🔧 Power-Ready Checklist**

* ✅ All power connections securely tightened
* ✅ Relays wired according to voltage and purpose
* ✅ GPIO inputs tested for correct logic behavior
* ✅ Audio connected to powered system
* ✅ SD card loaded with show files
* ✅ No exposed wire or loose conductors

**Section 7: Touchscreen Interface**

Control the show at the tap of a finger.  
  
The Showduino FX Controller is designed to place total show control directly in your hands. At the heart of this system is a responsive capacitive touchscreen interface that lets you navigate, trigger, and manage shows and props quickly — even in the dark, even in the chaos.  
  
This section walks you through every menu, button behavior, and icon in the user interface, offering insight into intuitive control and advanced configurations.

**🖥️ Touchscreen Overview**

|  |  |
| --- | --- |
| Feature | Description |
| Main Menu | Central hub with access to shows, settings, diagnostics, and manual control |
| Show Bank Grid | Grid of buttons representing show files or cue presets |
| Live Trigger Panel | Manually trigger GPIOs or individual relays in real-time |
| Settings Gear Icon | Opens configuration menus: pairing, brightness, autoload, etc. |
| Diagnostics Icon | Shows current device health, error logs, and version info |

**🗂️ Main Menu Navigation**

Upon boot, the Showduino displays the Main Menu screen:  
  
- Show Grid (default view): Displays up to 16 timeline show buttons per screen (more via scroll). Each button can include a custom label and color tag.  
- Live Controls Tab: Access GPIO triggers and relay toggles. Useful for testing or manual override.  
- System Settings: Tap the gear icon ⚙️ in the top right to open configuration options.

**▶️ Show Buttons (Grid Mode)**

Each show button represents a single JSON timeline file stored on the SD card. Tapping a show button will:  
- Begin playback of all programmed cues  
- Activate relays, GPIOs, sound effects, and lights  
- Change LED status to white (active)  
- Lock other buttons until show completes (optional in settings)

|  |  |
| --- | --- |
| Visual Element | Function |
| Label Text | Customizable title pulled from JSON metadata |
| Color Tag | Button color can reflect scare level, zone, or operator preference |
| Lock Icon 🔒 | If enabled, prevents re-triggering during execution |
| Progress Bar | Shows timeline position when enabled |

Long-press a button (2 sec) to open context menu for:  
- Preview audio only  
- Edit label/tag (via touchscreen or terminal)  
- Assign to external R3 Terminal button  
- Set auto-trigger interval (for ambient loops)

**⚡ Live Trigger Panel**

Accessible from the bottom menu bar or side swipe, this panel is ideal for manual control of props:  
  
- Relay 1–8 Toggle Buttons  
- GPIO Pin Triggers  
- “All Off” Panic Button — instantly deactivates all outputs  
- Timed Relay Test — send a pulse to test prop reaction (default: 2 sec)  
  
Each button changes color on state change (green = ON, grey = OFF). You can label relays in settings for easier tracking (e.g., “Air Cannon”, “Torch Flicker”).

**⚙️ Settings Menu Options**

|  |  |
| --- | --- |
| Setting Name | Description |
| Brightness | Adjust screen brightness from 10% to 100% |
| Auto Resume | Automatically reloads last show on reboot |
| Audio Volume | Sets output gain for line-level audio jack |
| Button Lockout | Prevents show overlap or repeated triggering |
| WiFi/Bluetooth Setup | Begin pairing with GoreFX Terminal (more in Section 9) |
| Firmware Update | Initiate USB or SD-based update process |
| Relay Labels | Assign names to relays for Live Trigger UI |
| Theme Selector | Choose between Light, Dark, or “Showfloor Red” visual theme |

**🛠️ Diagnostic Panel**

The diagnostic panel shows you everything under the hood — real-time and historical:  
- Voltage Rail Monitor: Shows 5V/12V supply status  
- Relay Status Map: Visual feedback on current relay states  
- Error Log: Last 10 system or show errors  
- System Uptime: Total time since last boot  
- Storage Status: SD card remaining space and number of valid shows  
  
A “System Reset” button here performs a soft reboot — no power cycle required.

**📊 Status Messages & Icons**

|  |  |
| --- | --- |
| Symbol/Icon | Meaning |
| 🔒 Lock Icon | Show locked — no triggering until current cue ends |
| 🔄 Circle Arrow | Replay last show (if enabled) |
| 📁 Folder | Load JSON show from SD manually |
| ⚠️ Exclamation | Warning or error — open diagnostics for details |
| 🔈 Speaker | Playing audio (with file name tooltip) |
| 🔋 Battery Icon | Only appears if using mobile power supply |

**🎨 Customizing the Interface**

You can customize the look and operation of the UI through the settings or JSON configuration:  
- Rename buttons to match show zones (e.g., “Cell Block Scream”)  
- Assign color codes to signify scare level or team ownership  
- Group buttons by theme (lighting, scare, audio-only)  
- Lock certain buttons from guest-trigger terminals  
  
📝 Pro Tip: Use the included Quick Start Card to record your button layout and cue map for fast reference.

**🤖 Offline vs Terminal Modes**

|  |  |
| --- | --- |
| Mode | Description |
| Offline Mode | All functions local to the Showduino. No GoreFX terminal or wireless input. |
| Terminal Mode | Paired with GoreFX or R3 remote — shows may be triggered wirelessly or remotely. |

**🔧 Troubleshooting UI Behavior**

* • Buttons not responding? Try a reboot from Diagnostics.
* • Frozen UI after show? Verify SD card and show file integrity.
* • Touch not registering? Calibrate in Settings > Touchscreen Calibration.
* • Show list empty? Ensure SD card is inserted and properly formatted (FAT32).

**Section 8: Triggering Props**

Bring the horror to life, one trigger at a time.  
  
The Showduino FX Controller was designed with prop integration at its core. Whether you're launching air cannons, lighting up haunted corridors, or animating skeletal puppets, Showduino offers flexible options to communicate with a wide range of external devices — both smart and simple.  
  
This section explains how to trigger props through Showduino’s relay outputs, GPIO pins, and DMX (if using an add-on).

**📡 Prop Triggering Methods Overview**

|  |  |  |
| --- | --- | --- |
| Method | Description | Best For |
| Relay Outputs | Physical relays activate or deactivate power lines | Simple lights, solenoids, strobes |
| GPIO Digital Pins | Sends logic HIGH/LOW signals | Sensors, buttons, LED strips |
| DMX Output (via Add-On) | Sends DMX512 commands via adapter | Lighting rigs, foggers, servo controllers |

**🔌 Using Relays for Triggering Props**

As detailed in Section 6, the onboard 8 relays can be wired directly to low-voltage props.  
  
Examples:  
- Powering a 12V strobe  
- Completing a 5V trigger circuit  
- Switching a relay board that powers pneumatic valves  
  
✅ Use the Live Trigger Panel to test these relays manually.  
✅ Use JSON timelines to automate relay activation with precise timing.

**🎧 Triggering MP3 Players via Serial**

Many MP3 playback modules use serial commands to trigger sound banks. The Showduino supports sending simple serial text or byte command streams over its internal serial ports for controlling these devices.  
  
📌 Note: Showduino does not support external microcontrollers like Arduino.  
  
Example Serial Command for MP3 trigger:  
{ "time": 0.0, "serial": "Serial1", "command": "PLAY\_01\n" }

**💡 Triggering DMX Fixtures (with Add-On Adapter)**

If you’re using the Showduino DMX Add-On, you can send DMX values to light fixtures, strobes, or foggers.  
  
Example:  
{ "time": 2.0, "dmx\_channel": 1, "value": 255 }  
  
DMX Config Notes:  
- Supports standard 512-channel DMX universe  
- Assign fixture addresses to match timeline values  
- Requires active DMX bus and termination resistor

**🧪 Testing Prop Triggers**

* • Toggle any relay ON/OFF
* • Pulse GPIO pins
* • Send test serial commands (via Developer Mode)

**📦 Sample Use Case: Multi-Device Show**

To create a show that:  
1. Plays an audio scream (via MP3 serial)  
2. Flashes a strobe (Relay 6)  
3. Fires a prop trigger via GPIO  
  
JSON Show File Example:  
[  
 { "time": 0.0, "serial": "Serial1", "command": "PLAY\_01\n" },  
 { "time": 0.2, "relay": 6, "state": "ON" },  
 { "time": 0.3, "gpio": 2, "state": "HIGH" },  
 { "time": 1.0, "relay": 6, "state": "OFF" }  
]

**🛠️ Troubleshooting Prop Commands**

|  |  |
| --- | --- |
| Symptom | Solution |
| MP3 plays wrong file | Confirm byte sequence and address values |
| Relay not clicking | Check power controller and wiring polarity |
| No DMX output | Confirm DMX adapter connection and correct fixture addressing |

**✅ Prop Triggering Checklist**

* ✅ Props are powered and connected
* ✅ Serial device baud rates match
* ✅ Relays click on manual test
* ✅ Timeline shows commands in correct sequence
* ✅ No exposed wires or shorting terminals

**Section 9: Connecting to GoreFX Terminal**

Wireless control for the fearless.  
  
The GoreFX Terminal is a wireless remote interface designed to pair with the Showduino FX Controller, enabling off-board triggering, mobile access to cues, and live control over your show system — from a safe or hidden location.  
  
Whether you’re coordinating timed scares from a control booth or allowing actors to trigger effects from behind a curtain, this integration gives your production team an edge in responsiveness and realism.

**📶 What is the GoreFX Terminal?**

The GoreFX Terminal is a dedicated device that connects to the Showduino via WiFi or Bluetooth, depending on your model. It functions as a remote trigger station, allowing operators to:  
- Activate individual show cues remotely  
- Trigger pre-assigned relay or GPIO effects  
- Monitor basic status indicators (battery, signal, system ready)  
- Adjust timing or intervals in live environments

**🧩 Supported Devices**

|  |  |  |  |
| --- | --- | --- | --- |
| Terminal Model | Connection Type | Display | Features |
| GoreFX R1 Remote | Bluetooth | LED + Button Panel | Up to 6 buttons, assignable |
| GoreFX Touchpad | WiFi | Capacitive Touch | Full screen UI mirroring |
| R3 Terminal | USB or Bluetooth | Keypad Interface | Basic trigger bank |

**🔧 Preparing for Connection**

Before connecting, ensure:  
- ✅ The Showduino is powered on and running firmware v1.2 or later  
- ✅ GoreFX Terminal is charged and within range (up to 40 ft Bluetooth, 100+ ft WiFi)  
- ✅ You have access to the Settings menu on the Showduino touchscreen

**📲 WiFi / Bluetooth Setup Instructions**

1. Open Settings > Wireless Pairing  
2. Select connection type: Bluetooth or WiFi  
3. Tap “Scan for Terminals”  
4. Select your GoreFX device from the list  
5. Tap “Pair” — status LED will turn blue when connected  
6. Assign buttons to shows or triggers via R3 Config Menu

**🎮 Using the Terminal**

Once paired, the GoreFX Terminal offers instant interaction:  
- Tap or press a button to fire the assigned cue  
- Long-press to enter assignment mode (if supported)  
- Use LED indicators for feedback:  
 - 🟢 Green = Ready  
 - 🔴 Red = Cue active  
 - 🔄 Yellow = Waiting for response

**⚙️ Assigning Functions to Terminal Buttons**

From the Showduino touchscreen:  
1. Open Settings > R3 Config  
2. Select the terminal to configure  
3. Tap a button (1–6) and assign:  
 - A timeline show  
 - A specific relay output  
 - A GPIO pulse  
4. Save configuration  
  
All assignments are stored on the Showduino’s SD card in the /CONFIG/ directory.

**📡 Terminal Range & Interference Notes**

|  |  |  |
| --- | --- | --- |
| Environment | Recommended Type | Max Range Estimate |
| Indoor Haunt Maze | Bluetooth | 30–50 ft (depending on walls/props) |
| Outdoor Staging | WiFi | 100–200 ft |
| High-Electromagnetic Area | WiFi (2.4GHz) | Use booster antenna if needed |

**🛠️ Troubleshooting Terminal Connections**

|  |  |
| --- | --- |
| Issue | Solution |
| Terminal not detected | Ensure pairing mode is active and device is within range |
| Buttons do nothing | Reassign functions under Settings > R3 Config |
| Show doesn’t trigger | Check SD card for missing or corrupted show files |
| LED won’t turn blue after pairing | Reboot both devices and retry connection |

**🧪 Live Trigger Use Case: Actor Panic Button**

Many attractions set up a hidden panic trigger for actors to activate a scare when the moment is right.  
  
Example Configuration:  
- GoreFX Terminal Button 1 → "Pop-Up Coffin Show"  
- Button 2 → "Strobe + Scream" sequence  
- Button 3 → “All Off” safety kill  
  
This lets the performer pace the timing of the scare while maintaining show flow.

**✅ Terminal Integration Checklist**

* ✅ Showduino firmware is up to date
* ✅ Terminal is charged and within range
* ✅ Terminal is paired and buttons respond
* ✅ Assigned cues load correctly on Showduino
* ✅ All changes saved to /CONFIG/ folder

**Section 10: Add-On Support**

Modular mayhem. Scalable screams.  
  
The Showduino FX Controller is not just a standalone controller — it’s the central nervous system of a scalable effects network. Through compatible add-ons and accessories, you can dramatically expand your haunt’s capabilities, enhance control precision, and adapt to even the most ambitious room designs or show sequences.  
  
This section outlines the supported add-ons, their features, and how to integrate them seamlessly into your Showduino ecosystem.

**🧩 What Are Add-Ons?**

Add-ons are hardware modules developed specifically for use with Showduino. They plug into designated ports (USB, GPIO, serial, or dedicated expansion headers) and immediately extend the device’s I/O, lighting, sound, or interaction abilities.  
  
They’re hot-pluggable (unless noted), instantly recognized, and configurable through the touchscreen interface or setup JSON files.

**🔌 Supported Showduino Add-Ons**

|  |  |  |
| --- | --- | --- |
| Add-On Name | Connection Type | Functionality |
| Lantern Light Module | GPIO | RGB LED beacon synced with shows or scare states |
| R3 Terminal | Bluetooth / USB | Remote keypad trigger system with 6 programmable buttons |
| Showduino DMX Out | Serial / RJ45 | Converts show timelines to DMX512 signals for lighting & FX control |
| Aux Sound Trigger | Serial / GPIO | Triggers pre-assigned MP3 boards for multi-zone sound |
| Expansion Harness | 10-pin header | Adds additional GPIOs, analog inputs, or alternate relay channels |

**🏮 Lantern Light Module (RGB FX)**

Use case: Ambient feedback, color-coded zones, "scare active" indicators.  
  
Features:  
- RGB controllable via timeline cues or button triggers  
- 5V powered through GPIO or USB rail  
- Custom color scripting via JSON  
  
Example JSON Command:  
{ "time": 0.5, "lantern": "red", "duration": 1.5 }  
  
This will turn the Lantern red for 1.5 seconds during a show.  
  
🎯 Mount near props, room entrances, or actor zones for visibility.

**🎛️ R3 Terminal**

Use case: Give actors, operators, or mobile crew quick access to effects without using the touchscreen.  
  
Features:  
- 6 large, tactile buttons with LED feedback  
- Each button assignable to any relay, GPIO, or full show  
- Wired or wireless connectivity  
  
Setup Path:  
Settings > R3 Config > Select Terminal > Assign Buttons

**💡 Showduino DMX Out Adapter**

Use case: Integrate lighting rigs, strobes, foggers, and moving heads via the DMX512 protocol.  
  
Features:  
- Sends values from JSON timeline to any DMX channel  
- Compatible with 3-pin or 5-pin DMX via adapter  
- Requires dedicated power and addressable fixtures  
  
Example JSON Cue:  
{ "time": 3.0, "dmx\_channel": 7, "value": 255 }  
  
🎚️ Control fade times and strobe bursts with timeline timing precision.

**🔊 Aux Sound Trigger**

Use case: For running multiple MP3 zones simultaneously without overloading one audio output.  
  
Features:  
- Uses serial commands to trigger external MP3 boards  
- Ideal for multi-room haunts with isolated sound effects  
- Works with DFPlayer Mini, WAV Trigger, and similar  
  
Wiring:  
Connect TX pin of Showduino’s serial port to RX on MP3 board. Ground must be shared.  
  
JSON Example:  
{ "time": 1.0, "serial": "Serial2", "command": "SOUND\_FIRE\n" }

**🧠 Expansion Harness**

Use case: More triggers, more props, more options.  
  
Features:  
- Adds up to 8 additional GPIOs  
- Optional relay expansion banks  
- Includes analog input channels for sensors, sliders, or feedback loops  
  
Installation:  
Plug into the Showduino’s Expansion Port (10-pin) and configure under Settings > Expansion  
  
Supported Modes:  
- GPIO Input (triggers, sensors)  
- GPIO Output (digital on/off)  
- Analog Read (light sensors, pressure pads, etc.)

**⚙️ Managing Add-Ons via UI**

Once plugged in, add-ons appear under the Settings > Add-Ons menu:  
- Enable/Disable: Toggle module activity  
- Assign Outputs: Link GPIO or relay behavior to UI or show timelines  
- Rename Add-Ons: Helpful for labeling by room or purpose (e.g., “Basement FX”)  
- Diagnostics: View real-time status, power draw, and communication health

**🚨 Compatibility & Power Notes**

|  |  |  |
| --- | --- | --- |
| Add-On | Power Draw Estimate | Notes |
| Lantern Module | ~250mA @ 5V | Use external 5V rail if running multiple simultaneously |
| R3 Terminal (BT) | ~150mA @ 3.3V | Battery optional, rechargeable via USB |
| DMX Out Adapter | ~80mA @ 5V | Minimal draw, requires signal-ground connection |
| Aux Sound Trigger | ~50–100mA | Shared ground required for reliable triggering |
| Expansion Harness | Varies | Use strain relief — high-density connectors are delicate |

**✅ Add-On Integration Checklist**

* ✅ Add-on securely connected (and not reversed)
* ✅ Powered on (if applicable)
* ✅ Detected in Showduino UI under Add-Ons
* ✅ Assigned to the correct cue, GPIO, or button
* ✅ Tested via Live Panel or timeline trigger

**Section 11: Custom Shows**

Craft the perfect scare, frame by frame.  
  
The Showduino FX Controller lets you choreograph immersive, multi-layered scare sequences with precision through custom show files. These files are written in JSON format and loaded from a microSD card, allowing you to define exact timings for triggering relays, playing sounds, controlling DMX lighting, and more — all in sync.  
  
This section teaches you how to structure, upload, preview, and execute your own timeline show files using Showduino’s system.

**📁 Show File Basics**

Each custom show is a `.json` file located in the `/SHOWS/` directory on your Showduino’s microSD card. These files define cue sequences based on time-stamped commands.  
  
File Naming Rules:  
- Use standard characters (A–Z, 0–9, underscores)  
- File names must end in `.json`  
- Avoid spaces or special characters  
  
Example file name: `coffin\_scare\_loop.json`

**🧱 Basic Structure of a Show File**

[  
 { "time": 0.0, "relay": 1, "state": "ON" },  
 { "time": 0.2, "serial": "Serial1", "command": "SOUND\_FIRE\n" },  
 { "time": 0.4, "dmx\_channel": 1, "value": 255 },  
 { "time": 2.0, "relay": 1, "state": "OFF" }  
]

**🧠 Supported JSON Cue Types**

|  |  |  |
| --- | --- | --- |
| Cue Type | Key | Example |
| Relay Toggle | "relay" | { "time": 0.0, "relay": 1, "state": "ON" } |
| Serial Command | "serial" | { "time": 1.2, "serial": "Serial1", "command": "PLAY\n" } |
| DMX Output | "dmx\_channel" | { "time": 2.0, "dmx\_channel": 3, "value": 128 } |
| GPIO Pulse | "gpio" | { "time": 0.5, "gpio": 2, "state": "HIGH" } |
| Lantern FX | "lantern" | { "time": 1.0, "lantern": "blue", "duration": 2.0 } |

**🛠️ Advanced Cue Features**

|  |  |
| --- | --- |
| Field | Description |
| "label" | Display name shown on touchscreen |
| "loop" | Enables looped playback (true/false) |
| "lockout" | Prevents re-triggering until completion |
| "duration" | How long an effect (like lantern) stays on |

Example of labeled looping show:  
{  
 "label": "Intro Lightning",  
 "loop": true,  
 "cues": [  
 { "time": 0.0, "relay": 5, "state": "ON" },  
 { "time": 1.0, "relay": 5, "state": "OFF" }  
 ]  
}

**🔄 Uploading Shows**

1. Remove the microSD card from the Showduino.  
2. Insert it into a computer.  
3. Open the `SHOWS/` folder.  
4. Paste your `.json` file inside.  
5. Eject the card and reinsert into the Showduino.  
  
✅ The file will automatically appear in the touchscreen grid once loaded.

**▶️ Previewing & Testing Shows**

- Tap the show button from the touchscreen grid.  
- View the progress bar and cue execution live.  
- Use Live Trigger Panel for manual overrides.  
- Long-press a show button to rename or set behavior flags.  
  
[QUESTION: 💡][QUESTION: ][QUESTION: Pro Tip: Use `dry\_run` mode from the Developer Tools menu to simulate a show without activating outputs.]

**💾 JSON Template Generator (Optional)**

If enabled on your system, Showduino includes a basic JSON show builder via the USB interface.  
  
Steps:  
1. Connect USB-C cable to a computer.  
2. Launch the Showduino JSON Builder app (if installed).  
3. Choose “Add Cue” and define:  
 - Cue type  
 - Timing  
 - Target (relay, serial, etc.)  
4. Save as `.json` into the SD card’s `SHOWS/` directory.

**⚠️ Common JSON Mistakes**

|  |  |
| --- | --- |
| Problem | Fix |
| File doesn’t show up | Confirm it’s inside `/SHOWS/`, has `.json` extension, and no typos |
| Show won’t trigger | Verify all time stamps are in seconds and numbers are valid |
| Only part of show plays | Ensure there are no stray commas or unclosed brackets |
| Buttons freeze after trigger | Likely malformed JSON or large file — test with fewer cues first |

**✅ Show Creation Checklist**

* ✅ File is named correctly (`.json`)
* ✅ Located inside `SHOWS/` folder on SD card
* ✅ Includes valid cue types and timing
* ✅ Loads successfully on touchscreen grid
* ✅ All cues trigger their targets as expected

**Section 12: Troubleshooting**

If the screams stop, you don’t.  
  
Even the best-controlled show system can run into issues — power surges, disconnected props, corrupted files, or unforeseen environmental interference. The Showduino FX Controller is equipped with built-in diagnostics, error signaling, and fail-safes to get you back up and scaring in no time.  
  
This section provides a comprehensive breakdown of how to identify, diagnose, and resolve common issues, from hardware anomalies to misbehaving timeline files.

**🚦 Status LED Guide (Recap)**

|  |  |  |
| --- | --- | --- |
| Color | Behavior | Meaning |
| 🟢 Green | Solid | System powered and idle (ready) |
| ⚪ White | Solid | Show currently playing |
| 🔵 Blue | Blinking | Pairing or communicating with remote terminal |
| 🔴 Red | Fast Blink | System or relay error |
| 🔴 Red | Solid | Critical error (manual reboot recommended) |
| 🟡 Yellow | Pulsing | USB or firmware update in progress |

**🛠️ General System Issues**

|  |  |  |
| --- | --- | --- |
| Symptom | Cause | Fix |
| Unit doesn’t power on | No power from controller or fuse blown | Verify 240V supply, check fuse, test with another outlet |
| Screen is dark or flickering | Dimmed in settings or power instability | Adjust brightness in settings; check power wiring |
| Buttons not responding | UI frozen or uncalibrated touchscreen | Reboot device or recalibrate from Settings > Touchscreen |
| No show files appear | SD card missing or improperly formatted | Ensure FAT32 format; file path should be `/SHOWS/` |

**🎭 Show Playback & Trigger Errors**

|  |  |  |
| --- | --- | --- |
| Symptom | Cause | Fix |
| Show plays but props don’t fire | Relay not wired correctly or wrong voltage | Check relay wiring, confirm relay state via Live Panel |
| DMX fixtures unresponsive | Address conflict or no DMX termination | Verify DMX chain config, test fixtures individually |
| MP3 trigger fails | Command syntax incorrect or serial mismatch | Confirm correct port and command format |
| Show freezes midway | Malformed JSON or memory overload | Test shorter version; inspect JSON formatting |
| Buttons lock indefinitely | Button lockout enabled in settings | Disable lockout or allow show to complete fully |

**🔧 Relay & GPIO Issues**

|  |  |  |
| --- | --- | --- |
| Issue | Check | Fix |
| Relay won’t click | Is the relay LED lit when triggered? | Check 5V/12V source and relay state |
| Relay stuck ON | JSON show didn’t turn it OFF | Add OFF state at desired time |
| GPIO unresponsive | Is pin configured as input/output? | Review pin config in UI |
| GPIO echoing multiple times | Shared ground? Debounce issues? | Use pull-down resistors, ground properly |

**📁 SD Card & File Handling Problems**

|  |  |
| --- | --- |
| Problem | Solution |
| SD card not detected | Format card as FAT32 (32GB max recommended), reinsert securely |
| File does not appear | Must be inside `/SHOWS/` folder and end with `.json` extension |
| Device freezes after file load | Malformed JSON — test with validator online or in text editor |
| File appears but won’t run | Review file structure — must start with array or object |

**💡 Diagnostics Menu Features**

* • Relay Test: Toggle each relay manually to check response
* • Serial Monitor: View serial command history
* • Voltage Rails: Displays current 5V/12V input levels
* • Error Log: Last 10 system errors with timestamps
* • Uptime Tracker: Shows how long the unit has run since last reboot
* • Reset All Settings: Wipes stored configs and reloads default behavior

**🧪 Live Debugging Tips**

* • Test each prop via the Live Trigger Panel before a show
* • Assign test shows to empty buttons to isolate cues
* • Use show logs (optional add-on) to capture cue-by-cue playback
* • Enable Dry Run Mode to check sequencing logic without output activation

**⚠️ Common Mistakes to Avoid**

|  |  |  |
| --- | --- | --- |
| Mistake | Consequence | Better Practice |
| Forgetting to power the controller | Unit appears dead | Always verify power LED (green) is on |
| Uploading shows to wrong folder | Nothing appears on touchscreen | All show files must go into `/SHOWS/` folder |
| Overlapping cues on same relay | Relay state flickers or skips | Sequence with brief buffer times (0.1s+) |
| Not saving R3 button assignments | Remote terminal won’t respond | Use “Save Config” in Settings > R3 Config |
| Using “smart quotes” in JSON | File unreadable | Use plain text editors (VSCode, Sublime, Notepad++) |

**🛡️ Failsafe Behavior**

Showduino is equipped with internal routines to prevent dangerous states:  
- Show watchdog timer: If show crashes, relays default to OFF  
- Power reset monitor: Device resets cleanly on brownout  
- Status LED alerts: Display distinct colors for issues  
- Touchscreen recovery: Long-hold top-right corner during boot to reset UI

**☎️ When to Contact Support**

If these fixes don’t resolve your issue:  
- Contact support with the following:  
 - Model and serial number (on back of unit)  
 - Show file name(s) in use  
 - Brief issue description and LED color/state  
 - What you’ve tried already  
  
📧 Reach out via the email and site in the Contact section.

**Section 13: Glossary**

The haunted techie's lexicon.  
  
This glossary defines key technical terms and acronyms used throughout the Showduino FX Controller manual. Whether you're new to haunt tech or a seasoned effects veteran, this section ensures you're never left in the dark — unless the show calls for it.

|  |  |
| --- | --- |
| Term / Acronym | Definition |
| GPIO | General Purpose Input/Output. Flexible pins used for triggering or sensing events. |
| Relay | An electrically controlled switch used to open/close circuits. Relays in Showduino trigger props or devices. |
| NO / NC | Normally Open / Normally Closed. Relay terminal types that define default circuit behavior. |
| Serial | A method of communication that sends data one bit at a time via TX/RX lines. |
| UART | Universal Asynchronous Receiver/Transmitter. The protocol used by Showduino for serial communication. |
| DMX512 | A standard protocol for digital communication between lighting control systems and fixtures. |
| Cue | A timed command executed during a show (e.g., turn on strobe, play scream). |
| JSON | JavaScript Object Notation. A structured file format used by Showduino to define custom show timelines. |
| Terminal | In this context, a GoreFX remote or device that connects wirelessly or via USB to trigger cues. |
| R3 Terminal | A 6-button remote unit assignable to Showduino show functions and relays. |
| Lantern Module | A Showduino RGB LED add-on used for lighting cues or zone feedback. |
| Expansion Harness | Hardware expansion for adding GPIOs, analog inputs, or relays. |
| Touchscreen UI | The interactive display on Showduino used to trigger, configure, and test shows. |
| Live Trigger Panel | Touchscreen-accessible manual control for relays, GPIOs, and test pulses. |
| SD Card | A removable storage medium used to store shows and configuration files. |
| FAT32 | A file system format required for the Showduino to read SD card contents. |
| Dry Run | A test mode where the Showduino simulates cue timings without activating outputs. |
| Show Grid | The main UI display showing all available show buttons for triggering. |
| Lockout Mode | Prevents button re-triggers while a show is playing to avoid cue overlap. |
| Power Controller | External 240V unit that supplies safe, regulated power to the Showduino hardware. |
| Watchdog Timer | A safety feature that resets outputs or the entire system if a crash or stall is detected. |
| Pairing Mode | Wireless setup state allowing Showduino to connect with GoreFX Terminals. |
| Baud Rate | The speed at which serial data is transmitted (e.g., 9600 bps). |
| TX / RX | Transmit / Receive. The two lines used in serial communication. |
| Timeline Show | A structured series of timed cues that define an effect sequence on Showduino. |
| CONFIG Folder | A special directory on the SD card where terminal and system settings are stored. |

**Section 14: Contact & Support**

You scare. We support.  
  
We know how important uptime and reliability are during a show. Whether you’re a haunt owner, tech operator, or a first-time builder, we’re here to help keep your effects system running smoothly. The Showduino FX Controller is supported by a passionate team of engineers and artists committed to responsive service and continuous product evolution.  
  
This section provides essential contact info, documentation access, and support resources.

**🛠️ Technical Support**

For troubleshooting help, installation questions, or firmware inquiries:  
  
📧 Email:  
showduino38@gmail.com  
  
🌐 Website:  
[www.show-duino.com/support](http://www.show-duino.com/support)  
  
🕰️ Support Hours:  
Monday – Friday  
9:00 AM – 6:00 PM (GMT)  
  
Please include the following in your message:  
- Device serial number (printed on back of unit)  
- Description of issue  
- What you’ve already tried  
- Any relevant JSON files or system logs

**📝 Warranty & Repair Policy**

All Showduino FX Controllers come with a 1-year limited warranty covering:  
- Factory defects  
- Component failures  
- Firmware bugs  
  
❌ Warranty does not cover:  
- Water damage  
- Burned relays from overcurrent use  
- Modifications or reverse engineering attempts  
  
🧰 Need a repair? Contact support for an RMA form and instructions.

**📦 Replacement Parts & Accessories**

Need extra SD cards, terminals, or replacement brackets?  
  
Browse accessories at:  
🛒  [HYPERLINK "http://www.show-duino.com/store"www.show-duino.com/store](HYPERLINK%20%22http://www.show-duino.com/store%22www.show-duino.com/store)  
  
Available parts include:  
- R3 Terminals (Bluetooth or USB)  
- Lantern FX Modules  
- DMX Out adapters  
- Mounting brackets  
- Power controllers  
- Expansion harnesses

**📚 Additional Documentation & Tutorials**

Need setup help, JSON templates, or wiring diagrams?  
  
Check our library of guides and videos:  
📘  [HYPERLINK "http://www.show-duino.com/docs"www.show-duino.com/docs](HYPERLINK%20%22http://www.show-duino.com/docs%22www.show-duino.com/docs)  
  
Resources include:  
- Timeline show file examples  
- GPIO & relay wiring schematics  
- SD card formatting walkthroughs  
- Firmware update instructions  
- Add-on integration charts

**📢 Community & Updates**

Join the growing haunt tech community and get sneak peeks at new features:  
  
📣 Instagram: @SHOW.DUINO  
📺 YouTube: ShowduinoFX  
💬 Discord (invite via support portal)  
  
Sign up for firmware update alerts and new feature releases directly through the touchscreen or at:  
🔔  [HYPERLINK "http://www.show-duino.com/updates"www.show-duino.com/updates](HYPERLINK%20%22http://www.show-duino.com/updates%22www.show-duino.com/updates)