

Car Loading Panel Instruction Manual

Copyright © 2025, McNab Media. All Rights Reserved.

DISCLAIMER

READ CAREFULLY BEFORE PROCEEDING

This project and associated documentation are provided "AS IS" without warranty of any kind, either express or implied, including but not limited to warranties of merchantability, fitness for a particular purpose, or non-infringement.

BY BUILDING AND USING THIS DEVICE, YOU AGREE TO THE FOLLOWING:

- You assume ALL responsibility and liability for the construction, installation, operation, and use of this device
- You acknowledge that this project involves electrical components and soldering that may pose risks of fire, electric shock, or injury if improperly constructed or used
- McNab Media and all associated parties provide NO support, NO warranty, and NO guarantee that this device will function as described
- You are solely responsible for ensuring safe construction practices and compliance with all applicable electrical codes and regulations
- McNab Media and all associated parties shall NOT be liable for any damages, injuries, property damage, or losses arising from the use of this device or documentation
- You use this information entirely at your own risk

If you do not agree to these terms, do not proceed with this project.

What This Device Does

The Car Loading Panel is a model railroad accessory that simulates the time it takes to load a hopper car. When you press a button, it displays a loading progress bar on a screen for 96 seconds (representing the time to load your car). It provides audio feedback when loading starts, completes, or is cancelled.

Key Features:

- 96-second loading timer with visual progress bar

- Audio alerts for different events
 - One-button operation
 - Automatic sleep mode to save power
 - LED indicator for status
-

What You'll Need

Electronic Components

Required:

- 1× Arduino Uno or compatible board
- 1× 16×2 LCD Display with I2C interface (I2C address: 0x27)
- 1× Adafruit Audio FX Sound Board + 2x2W Amp - WAV/OGG Trigger -16MB (Product ID: 2217)
- 1× Momentary push button (normally open)
- 1× LED (any color, 5mm recommended)
- 1× 220Ω resistor (for LED)
- 1 or 2× Speakers (4-8Ω impedance, the 2217 board can drive stereo or mono)
- Jumper wires (male-to-male and male-to-female)
- Breadboard or perfboard for prototyping
- USB cable for Arduino programming
- Micro USB cable for sound board programming

Optional but Recommended:

- Project enclosure
- Screw terminals for easier connections
- Heat shrink tubing
- Labels for connectors

Tools Required

- Soldering iron and solder

- Wire strippers
- Small screwdriver set
- Multimeter (for testing connections)
- Computer with USB port for programming

Software Required

- Arduino IDE (free download from arduino.cc)
 - USB drivers for your Arduino board
-

About the Adafruit Audio FX Sound Board (Product ID: 2217)

The Adafruit 2217 board features 16MB of built-in storage, a 2x2W stereo Class D amplifier, and can drive 4-8 ohm speakers up to 2.2W with only 1% distortion. This specific board is perfect for this project because:

- No SD card needed - Files are stored directly on the board
- Built-in amplifier - Speakers connect directly (no external amp needed)
- UART control - Can be controlled by the Arduino via serial communication
- Simple file naming - Audio files are triggered by sending simple commands

Important Notes:

- This board is for direct-to-speaker connection only and cannot be connected to another amplifier
 - The board comes with header pins and two 2-pin terminal blocks for speaker connections
 - Power requirements: 3-5.5V DC
-

Preparing the Sound Board

Creating Audio Files

You'll need three sound files in OGG format:

- L.OGG - Plays when loading starts
- C.OGG - Plays when loading is cancelled

- D.OGG) - Plays when loading is complete

Audio File Tips:

- Keep files short (1-3 seconds for alerts)
- OGG format is more space-efficient than WAV
- Normalize audio levels so they're not too quiet or distorted
- Mono files use less space than stereo

Loading Files onto the Sound Board

1. Connect the sound board to your computer
 - Use a micro USB cable
 - Connect to the micro USB port on the sound board
 - The board should appear as a USB drive
2. Copy your audio files
 - Open the USB drive that appears
 - Copy your three audio files to the root directory (not in any folder)
 - CRITICAL: Filenames must be EXACTLY as shown above, including uppercase letters
 - Example: L.OGG (not l.ogg or L .OGG)
3. Safely eject
 - Safely eject/remove the USB drive from your computer
 - Disconnect the USB cable

Circuit Assembly

Important: UART Mode Setup

The Adafruit 2217 board must be set to UART mode to work with this project. Connect the UG pin to ground to enable UART mode. This is CRITICAL - without this connection, the Arduino cannot control the sound board.

Pin Connections

LCD Display (I2C)

- VCC → Arduino 5V
- GND → Arduino GND
- SDA → Arduino A4
- SCL → Arduino A5

Push Button

- One side → Arduino A2
- Other side → Arduino GND
- Note: The code uses internal pullup resistor, no external resistor needed

LED with Resistor

- LED Anode (+, longer leg) → 220Ω resistor → Arduino A3
- LED Cathode (-, shorter leg) → Arduino GND

Adafruit Audio FX Sound Board (2217)

- VIN → Arduino 5V
- GND → Arduino GND
- UG → Arduino GND (CRITICAL - enables UART mode)
- RX → Arduino Pin 7
- TX → Arduino Pin 8
- L+ terminal → Left speaker positive (or single speaker)
- L- terminal → Left speaker negative
- R+ terminal → Right speaker positive (optional, for stereo)
- R- terminal → Right speaker negative (optional, for stereo)

Important Wiring Notes:

- TX on the sound board is 3.3V logic output, RX on the sound board accepts 3-5V logic input
- The RX/TX labels refer to the sound board's perspective
- Arduino Pin 7 connects to sound board RX (Arduino transmits to sound board)
- Arduino Pin 8 connects to sound board TX (Arduino receives from sound board)

- Do not connect R+ to L+ and R- to L- to try to get more power - this could damage the amplifier chip

Speaker Connection

The 2217 board includes two 2-pin screw terminal blocks for easy speaker connection:

1. Solder the terminal blocks to the board (at L+/L- and R+/R- positions)
2. Strip speaker wires about 1/4 inch
3. Insert wires into terminals and tighten screws firmly
4. For mono setup: Connect one speaker to either left or right channel
5. For stereo setup: Connect two speakers to both channels

Speaker Recommendations:

- Use 4Ω or 8Ω speakers
 - Each channel can output up to 2.2W at 1% distortion
 - Small speakers (2-3 inch diameter) work well for model railroad applications
-

Step-by-Step Assembly Instructions

SAFETY FIRST: Unplug Arduino from power before making any connections.

1. Prepare the sound board
 - Solder the included header pins to the board if not already attached
 - Solder the screw terminal blocks for speaker connections
 - Load your audio files (see "Preparing the Sound Board" section)
2. Set up your workspace
 - Place your breadboard on a stable surface
 - Keep all components organized
 - Have your wiring diagram handy
3. Connect the LCD Display
 - Insert the LCD into the breadboard or connect via jumper wires
 - Connect VCC to Arduino 5V rail

- Connect GND to Arduino GND rail
 - Connect SDA to Arduino pin A4
 - Connect SCL to Arduino pin A5
4. Connect the Push Button
- Insert button into breadboard
 - Connect one leg to Arduino pin A2
 - Connect the other leg to GND
5. Connect the LED
- Insert LED into breadboard (note polarity!)
 - Connect 220Ω resistor to LED anode (longer leg)
 - Connect other end of resistor to Arduino pin A3
 - Connect LED cathode (shorter leg) to GND
6. Connect the Audio FX Board (CRITICAL STEPS)
- Connect VIN to Arduino 5V
 - Connect GND to Arduino GND
 - Connect UG to GND (This enables UART mode - DO NOT SKIP)
 - Connect RX to Arduino pin 7
 - Connect TX to Arduino pin 8
 - Connect speaker(s) to screw terminals
7. Double-check all connections
- Verify power connections (5V and GND)
 - Verify UG is connected to GND (most common mistake!)
 - Ensure no short circuits
 - Check LED polarity
 - Confirm I2C connections (A4 and A5)
 - Verify RX connects to pin 7, TX to pin 8
-

Installing the Software

Setting Up Arduino IDE

1. Download Arduino IDE
 - Go to arduino.cc/en/software
 - Download and install the appropriate version for your computer
2. Install Required Libraries
 - Open Arduino IDE
 - Go to Sketch → Include Library → Manage Libraries
 - Search for and install:
 - "LiquidCrystal I2C" by Frank de Brabander
 - Note: Wire and SoftwareSerial are built-in, no installation needed

Uploading the Code

1. Open the Code
 - Open the provided .ino file into Arduino IDE
2. Connect Arduino
 - Plug Arduino into computer via USB
 - Select correct board: Tools → Board → Arduino Uno
 - Select correct port: Tools → Port → [Your Arduino Port]
3. Upload
 - Click the Upload button (right arrow icon)
 - Wait for "Done uploading" message
 - Check for any error messages

Troubleshooting Upload Issues

- "Port not found": Install USB drivers for your Arduino
 - "LCD not found": Check I2C address (may need to change from 0x27 to 0x3F)
 - Compilation errors: Ensure libraries are installed correctly
-

Testing Your Device

Initial Power-Up Test

1. Connect power to Arduino (USB or external 5V supply)
2. LCD should remain off - device starts in sleep mode
3. LED should be off
4. No sound should play

UART Mode Verification

If nothing works and the sound board seems unresponsive:

- Verify UG pin is connected to ground - This is the #1 cause of issues
- Power cycle the system (disconnect and reconnect power)
- The UG pin must be tied to ground and the board must be reset to enter UART mode

Button Press Test

1. Press the button once
2. LCD should turn on and display "LOADING HOPPER"
3. Progress bar should appear and slowly fill over 96 seconds
4. LED should blink slowly (once per second)
5. Audio alert should play (PL file)

If no audio plays:

- Check speaker connections
- Verify audio files are properly named and loaded
- Ensure UG pin is grounded
- Try increasing volume (see Customization section)

Completion Test

1. Wait for loading to complete (96 seconds, or modify timing - see Customization)
2. LCD should display "LOADING COMPLETE"
3. LED should blink rapidly

4. Completion audio alert should play (PD file)
5. After 5 seconds, device should return to sleep mode

Cancel Test

1. Press button to start loading
2. Press button again during loading
3. LCD should display "LOADING CANCELED"
4. Cancel audio alert should play (PC file)
5. After 5 seconds, device should return to sleep mode

Troubleshooting Audio Issues

No sound at all:

- Verify UG pin is connected to ground
- Check that audio files are named correctly (uppercase, exact format)
- Test speaker with another audio source
- Verify Arduino pins 7 and 8 are connected correctly
- Check 5V power to sound board

Very quiet sound:

- As of June 2017, the gain is set to 6dB to prevent amplifier damage
- Use external audio editing software to normalize/boost audio files before loading
- Ensure you're using adequate power (external 5V recommended)
- Some users report low volume even at maximum settings

Distorted or crackling sound:

- Ensure using sufficient power supply
 - Check speaker impedance matches (4-8Ω)
 - Reduce audio file volume/levels
-

Operation Guide

Starting a Loading Cycle

1. Press the button once
2. LCD will display "LOADING HOPPER" with a progress bar
3. Audio alert will play
4. Wait 96 seconds for loading to complete
5. Device will alert you when complete

Cancelling a Loading Cycle

- Press the button again at any time during loading
- Device will cancel and display "LOADING CANCELED"
- Cancel audio alert will play

After Completion

- Device automatically returns to sleep mode after 5 seconds
- Press button to start a new loading cycle

Sleep Mode

- LCD turns off to save power and extend display life
- Device remains ready - just press button to wake
- No need to unplug between uses

Customization Options

Changing Load Time

Find this line in the code:

```
const unsigned long LOADING_TIME = 96000; // 96 seconds in ms
```

Change 96000 to your desired time in milliseconds:

- 1 second = 1000 ms
- 60 seconds = 60000 ms
- 120 seconds = 120000 ms

Troubleshooting

LCD Shows Nothing

- Check I2C address (try 0x3F if 0x27 doesn't work)
- Verify I2C connections to A4 and A5
- Adjust LCD contrast potentiometer on back of display
- Check 5V and GND connections

LCD Shows Garbled Characters

- Re-upload the code
- Check for loose connections
- Verify correct I2C library is installed
- Try different I2C address in code

No Sound

- #1 cause: UG pin not connected to ground - Verify this first!
- Verify sound files are properly named (exact uppercase format)
- Check Audio FX board connections
- Test speaker with known working audio source
- Verify 5V power to sound board
- Ensure adequate power supply (not just USB from computer)
- Try power cycling the entire system

Sound Plays But Wrong Files

- Check file naming - must be exact (L.OGG, C.OGG, D.OGG)
- Ensure no extra spaces or characters in filenames
- Verify files are in root directory, not in folders

Button Doesn't Work

- Check button connections to A2 and GND
- Try pressing firmly - some buttons need more pressure

- Verify button is normally-open type
- Test button with multimeter

LED Doesn't Light

- Check LED polarity (swap if needed - longer leg to resistor)
- Verify 220 Ω resistor is connected
- Test LED with multimeter (diode test mode)
- Check pin A3 connection

Device Resets Randomly

- Power issue - use external 5V power supply (not USB)
- Audio amplifier can draw high current spikes
- Add 100 μ F or larger capacitor between 5V and GND near Arduino
- Check for loose connections

Audio Stutters or Cuts Out

- Insufficient power supply - use external 5V with adequate current
- Check for loose connections
- Ensure good contact in screw terminals
- Don't query the sound board too frequently during playback

Mounting and Installation Tips

Enclosure Recommendations

- Choose a case large enough for all components
- Ensure adequate ventilation (Arduino and amplifier generate heat)
- Position LCD where it's easily visible
- Mount button for easy access
- Consider speaker placement for best sound
- Provide access to Arduino USB port (for future updates)
- Provide access to sound board USB port (for changing audio files)

Panel Mounting

1. Cut holes for LCD, button, and LED
 2. Mount speakers in or on enclosure (consider acoustic properties)
 3. Use hot glue or small screws to secure components
 4. Label controls for operators
 5. Ensure screw terminals are accessible for maintenance
-

Technical Specifications

System Specifications

Power Requirements:

- Arduino: 5V DC, ~50mA
- LCD: 5V DC, ~20mA
- Sound Board + Amplifier: 3-5.5V DC, up to 2A peak
- Total System: 5V DC, 2A recommended

Timing:

- Loading Duration: 96 seconds (configurable)
- Complete Display Time: 5 seconds
- Sleep Alert: Initial after 10 seconds
- Debounce Delay: 100ms

Display:

- 16×2 character LCD with I2C interface
- I2C Address: 0x27 (or 0x3F)
- Backlight control: Automatic (on during operation)

Audio:

- Adafruit Audio FX Board 2217
- Format: OGG files
- Storage: 16MB built-in

- Amplifier: 2x2W Class D stereo
- Output: Up to 2.2W per channel at 1% distortion (4Ω)
- Speaker Impedance: 4-8 Ω

Operating Environment:

- Temperature: 0°C to 50°C (32°F to 122°F)
 - Humidity: 20-80% non-condensing
-

Additional Resources

Official Documentation

- Adafruit Audio FX Sound Board Guide: learn.adafruit.com/adafruit-audio-fx-sound-board
- Arduino Official Site: arduino.cc
- LiquidCrystal I2C Library: github.com/johnrickman/LiquidCrystal_I2C

Learning Resources

- Arduino forums for community support: forum.arduino.cc
- Adafruit forums: forums.adafruit.com
- YouTube tutorials for Arduino beginners

Purchasing Components

- Adafruit Industries: adafruit.com (official source for 2217 board)
- SparkFun Electronics: sparkfun.com
- Amazon, DigiKey, Mouser (various retailers)
- Local electronics stores
- Model railroad hobby shops (some carry Arduino accessories)

Sound File Resources

- Free sound effects: freesound.org
- Audio editing: Audacity (free, open-source)
- OGG conversion tools: online-convert.com

Frequently Asked Questions

Q: Can I use the headphone version of the Adafruit board instead of the 2217? A: No, this project specifically requires the amplified version (2217) to drive speakers directly. The headphone version would need an external amplifier.

Q: Why isn't my sound board responding? A: The most common cause is forgetting to connect the UG pin to ground. This **MUST** be done for UART mode to work.

Q: Can I control multiple sound effects? A: Yes, the code can be modified to play different sounds. The 2217 board supports multiple trigger modes and file naming schemes.

Q: The volume is too quiet, how do I fix it? A: Edit your audio files to increase volume before loading them onto the board. The 2217's gain is limited to protect the amplifier.

Q: Can I make the loading time longer or shorter? A: Yes, change the `LOADING_TIME` constant in the code (see Customization section).

Q: Will this work with other Arduino boards? A: Yes, it should work with most Arduino boards that have the required pins. You may need to adjust pin assignments.

Support and Contact

This project is provided without official support. For assistance:

- Code questions: Arduino community forums
- Sound board questions: Adafruit support forums
- Hardware issues: Check connections and consult this manual's troubleshooting section

Remember: You are responsible for all aspects of this project. Build safely, test thoroughly, and enjoy on your model railroad!
