X-Plane Dataref Bridge - User Manual

Table of Contents

Introduction

X-Plane Dataref Bridge is a powerful application that enables bidirectional communication between X-Plane flight simulator and Arduino devices. It allows you to:

- Monitor and control X-Plane datarefs through your custom hardware
- Map physical inputs (buttons, switches, joysticks) to X-Plane commands and datarefs
- Create complex logic blocks for advanced functionality
- Generate Arduino code automatically for your custom hardware

The bridge acts as an intermediary, translating between X-Plane's dataref system and your Arduino devices, providing a flexible and user-friendly interface for creating custom cockpit hardware.

Getting Started

System Requirements

- Windows operating system (tested on Windows 10/11)
- Python 3.8 or higher
- X-Plane 11 or newer
- Arduino IDE (for uploading firmware to your devices)
- USB cable for connecting Arduino devices

Installation

1. **Download the application** from the official repository
2. **Extract the files** to a folder of your choice
3. **Install Python dependencies** by running:

pip install -r requirements.txt

4. **Launch the application** by running main.py

First-Time Setup

1. **Configure X-Plane Connection**:
    - Go to Settings panel
    - Set X-Plane IP (default: 127.0.0.1 for local connection)
    - Set X-Plane Port (default: 49000)
    - Set Receive Port (default: 49001)
2. **Connect to X-Plane**:
    - Click the "Connect to X-Plane" button in the main window
    - Verify connection status in the status bar
3. **Connect Arduino Devices**:
    - Connect your Arduino device via USB
    - Go to Arduino panel

- Select the port and click "Connect"

Interface Overview

The main application window consists of several panels:

- **Dataref Panel**: Browse and subscribe to X-Plane datarefs
- **Arduino Panel**: Manage connected Arduino devices
- **Input Panel**: Configure input mappings from physical devices
- **Output Panel**: Configure output mappings to Arduino devices
- **Monitor Panel**: View real-time dataref values
- **Logic Panel**: Create and manage logic blocks
- **Settings Panel**: Configure application settings

Working with Datarefs

Understanding Datarefs

Datarefs are X-Plane's way of exposing internal variables to external applications. They can represent various aircraft parameters such as:

- Flight controls (ailerons, elevators, rudder)
- Engine parameters (RPM, fuel flow, temperature)
- Instrument readings (airspeed, altitude, heading)
- System states (lights, gear position, flaps)

Browsing Datarefs

1. Navigate to the Dataref Panel
2. Use the search bar to find specific datarefs
3. Filter by category using the dropdown menu
4. Click on a dataref to view its details

Subscribing to Datarefs

1. Select a dataref from the list
2. Click the "Subscribe" button
3. The dataref will appear in the Monitor Panel with its current value

Creating Custom Datarefs

1. Click the "Add Custom Dataref" button
2. Enter the dataref name, type, and description
3. Specify if the dataref is writable
4. Click "Save" to add it to your database

Arduino Integration

Connecting Arduino Devices

1. Connect your Arduino device via USB
2. Go to the Arduino Panel
3. Select the port from the dropdown list
4. Click "Connect" to establish communication

Supported Devices

- Arduino Uno, Nano, Mega
- ESP32-S2, ESP32-S3
- Other Arduino-compatible boards

Communication Protocol

The application uses a custom serial protocol to communicate with Arduino devices:

- **PC → Arduino**: SET <KEY> <VALUE> - Send data to Arduino
- **Arduino → PC**: INPUT <KEY> <VALUE> - Send input data from Arduino
- **Arduino → PC**: CMD <COMMAND> - Execute X-Plane command

Uploading Firmware

1. Open the Arduino IDE
2. Load the example sketch from the "Arduino libraries/examples" folder
3. Select your board and port
4. Upload the firmware to your device

Input Mapping

Creating Input Mappings

1. Go to the Input Panel
2. Select a device from the list
3. Click "Add Mapping"
4. Configure the mapping in the dialog:
    - Select the input trigger (button, switch, axis)
    - Choose the action type (Command, Dataref Set, etc.)
    - Set the target (X-Plane command or dataref)
    - Configure parameters (values, increments, etc.)
5. Click "Save" to create the mapping

Action Types

- **Command**: Execute an X-Plane command
- **Dataref Set**: Set a dataref to a specific value
- **Dataref Toggle**: Toggle a dataref between 0 and 1
- **Dataref Increment/Decrement**: Increase or decrease a dataref value
- **Axis**: Map an axis input to a dataref with calibration options
- **Sequence**: Execute multiple actions in sequence
- **Custom**: Execute custom logic

Axis Calibration

1. Create an axis mapping
2. Click "Calibrate" to open the calibration wizard
3. Follow the on-screen instructions to set min/max values
4. Adjust deadzone and response curve as needed
5. Save the calibration settings

Output Configuration

Assigning Output IDs

1. Go to the Output Panel
2. Add a dataref to monitor
3. Enter an Output ID (e.g., "GEAR_LED")
4. The application will automatically send updates to Arduino devices

Generating Arduino Code

1. Configure your output mappings
2. Click "Generate Code" button
3. Copy the generated code to your Arduino IDE
4. Upload to your device

Dataref Types

The application supports various dataref types:

- **Scalar**: int, float, double, bool
- **Array**: int[n], float[n], byte[n]
- **String**: byte[n] used as strings
- **Command**: Execute actions without values

Writing to Datarefs via Output IDs

Users can write to datarefs using their assigned OUTPUT IDs:

[redacted]

SET <OUTPUT_ID> <value>

For example:

[redacted]

SET GEAR_LED 1.0

For array datarefs:

[redacted]

SET ENGINE_THRUST[0] 0.5

SET ENGINE_THRUST 0.5,0.6,0.7,0.8

For string datarefs:

SET AIRCRAFT_ICAO C172

For command datarefs:

CMD GEAR_TOGGLE

Logic Engine

Creating Logic Blocks

1. Go to the Logic Panel
2. Click "Add Logic Block"
3. Configure the logic block:
    - Set conditions (datarefs, variables, etc.)
    - Define the logic operator (AND, OR, etc.)
    - Set outputs (variables, commands, etc.)
4. Save the logic block

Logic Operators

- **AND**: All conditions must be true
- **OR**: At least one condition must be true
- **NAND**: Not all conditions are true
- **NOR**: No conditions are true
- **XOR**: Exactly one condition is true
- **XNOR**: Either all or no conditions are true

Using Variables

1. Create a variable in the Logic Panel
2. Set its initial value and update conditions
3. Use the variable in logic blocks or mappings

Profiles Management

Saving Profiles

1. Configure your mappings and settings
2. Go to File → Save Profile
3. Enter a profile name
4. Click "Save"

Loading Profiles

1. Go to File → Load Profile
2. Select a profile from the list
3. Click "Load"

Exporting/Importing Profiles

1. Go to File → Export Profile to save a profile to an external file
2. Go to File → Import Profile to load a profile from an external file

Auto-Loading Aircraft Profiles

The application can automatically load profiles based on the aircraft ICAO code:

1. Name your profile with the aircraft ICAO code (e.g., "B737.json")
2. Enable "Auto-load aircraft profile" in settings
3. The application will automatically load the matching profile when you change aircraft in X-Plane

---

Advanced Features

HID Device Integration

The application supports HID devices (joysticks, gamepads):

1. Connect your HID device
2. Go to the HID Panel
3. View device information and inputs
4. Create mappings for HID inputs

Custom Dataref Editor

1. Select a dataref in the Dataref Panel
2. Click "Edit" to open the editor
3. Modify values based on dataref type (scalar, array, string)
4. Save changes to X-Plane

Debugging

1. Enable debug logging in settings
2. View log messages in the application log
3. Use the Monitor Panel to track dataref values in real-time

---

Troubleshooting

Connection Issues

- **X-Plane not connecting**: Check IP and port settings in the Settings panel
- **Arduino not connecting**: Verify the correct port is selected and the device is properly connected
- **Datarefs not updating**: Ensure you're subscribed to the datarefs and X-Plane is running

Value Issues

- **Incorrect values**: Check dataref type and format in the Dataref Panel
- **Values not updating**: Verify the dataref is writable and you have the correct permissions

Performance Issues

- **Slow response**: Reduce the number of subscribed datarefs
- **High CPU usage**: Decrease update frequency for datarefs

Arduino Issues

- **Device not recognized**: Ensure the correct firmware is uploaded
- **Communication errors**: Check baud rate settings and cable connections
- **Incorrect values**: Verify the Arduino code matches your output configuration

Getting Help

- Check the application log for error messages
- Review the example Arduino sketches in the "Arduino libraries/examples" folder
- Consult the PROTOCOL.md file for detailed communication protocol information
- Visit the project repository for documentation and community support

---

This manual provides a comprehensive guide to using the X-Plane Dataref Bridge application. For more specific information about each feature, refer to the relevant sections or explore the application's built-in help system.