EEZ Studio
User manual

Low-code embedded GUI development tool
T&M automation and management

Table of Contents

1. Legal information

1.1. Definitions

Draft – A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. Envox does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

1.2. Disclaimers

Limited warranty and liability – Information in this document is believed to be accurate and reliable. However, Envox does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. Envox takes no responsibility for the content in this document if provided by an information source outside of Envox.
In no event shall Envox be liable for any indirect, incidental, punitive, special or consequential damages (including – without limitation – lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.
Notwithstanding any damages that customer might incur for any reason whatsoever, Envox' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of Envox.

Right to make changes – Envox reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use – Envox products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an Envox product can reasonably be expected to result in personal injury, death or severe property or environmental damage. Envox and its suppliers accept no liability for inclusion and/or use of Envox products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications – Applications that are described herein for any of these products are for illustrative purposes only. Envox makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.
Customers are responsible for the design and operation of their applications and products using Envox products, and Envox accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the Envox product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.
Envox does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using Envox products in order to avoid a default of the applications and the products or of the application or use by customer's 3rd party customer(s). Envox does not accept any liability in this respect.

Suitability for use in non-automotive qualified products – Unless this data sheet expressly states that this specific Envox product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. Envox accepts no liability for inclusion and/or use of non-automotive qualified products in automotiveequipment or applications.
In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without Envox' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer

uses the product for automotive applications beyond Envox' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies Envox for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond Envox' standard warranty and Envox' product specifications.

Security – Customer understands that all Envox products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by Envox products for use in customer's applications. Envox accepts no liability for any vulnerability. Customer should regularly check security updates from Envox and follow up appropriately.
Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by Envox.
To report a security issue, use the EEZ Studio issue tracker.

1.3. Miscelaneous

Open source license and contributions – EEZ Studio uses the *GPL v3* license. To view a copy of this license, please visit https://www.gnu.org/licenses/gpl-3.0.html. EEZ Studio uses the C4.1 (Collective Code Construction Contract) process for contributions.
This document is released under *open license FDL v1.3* from GNU.org. Therefore you are entitled to freely copy and redistribute it, with or without modifying it, either commercially or non-commercially. For additional details please consult the content of the license.

Terms and conditions of commercial sale – Envox products are sold subject to the general terms and conditions of commercial sale, as published at https://www.envox.eu/company/terms-of-use/, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. Envox hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of Envox products by customer.

Translations – A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Export control – This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Trademarks – All referenced brands, product names, service names, and trademarks are the property of their respective owners.

1.4. Contact information

If you have any problem or requirement when using EEZ products or this manual, please contact Envox:
> Discord server: https://discord.gg/dhYMnCB
> E-mail: support@envox.eu
> Website: www.envox.eu

1.5. Revision history

Date	Version	Changes
2023-03-31	1.0	Inital version

2. The EEZ Studio overview

2.1. Introduction

EEZ Studio was initally developed as a companion application for the in-house developed H24005 programmable power supply and BB3 T&M chassis to address two important tasks: a) remote programming and management and b) simplifying the development of a feature rich embedded GUI for a color touch-screen display.

The development was inspired by the idea of offering an open source alternative to some existing commercial solutions that are used for the mentioned tasks, all in order to overcome the limitations of their closed code, outdated and complex UI or sometimes awkward UX and licensing, which in our case was not in accordance with the open source of the mentioned devices that we have developed.

2.2. Main sections

EEZ Studio consists of two main sections, which are described separately in the manual:

- **Projects** – creating, editing, debugging and building the code for the embedded GUI project for the selected target platform. Generated code can be directly imported into the IDE/toolchain used to build the firmware and accelerate the development process. It enables the rapid development of high quality embedded GUI and also comes with support for the open-source LVGL graphics library. The drag-and-drop editor makes it easy to utilize the many features such as widgets, animations, and styles to create a GUI reducing the coding effort. Additionaly flowchart-like *EEZ flow* programming feature will further save development time and complexity.

- **Instruments** – allows access to one or more T&M instruments using several communication interfaces through which it is possible to manage and collect measurement data and screenshots using SCPI commands and queries. Collected data can be analyzed, searched, annotated and exported to other applications. Automation of test and measurement tasks using JavaScript and *EEZ flow* programming allows it to be used in different scenarios from basic development, calibration, troubleshooting and quality control using multiple devices from different manufacturers that can be in different locations connected to LANs.

In the introductory chapters of the two main sections that follow, all important features will be listed and described in detail. Common features are listed below:

- Modern and attractive UI/UX developed in Electron
- Light / Dark theme
- Cross-platform run-time (Linux, Windows, MacOS)
- Modular design based on plug-ins that can be added/removed depends of scope of the work
- Source/Version control integration (GitHub and gitea.io)

2.3. Known issues and issue reporting

EEZ Studio is continuously developing and improving. A list of known issues can be found on GitHub where you are also invited to leave your suggestions for improvements and new functionality.

When reporting bugs using the GitHub tracking system, please first check if the issue you want to report has already been reported by someone else. When opening a new ticket, the following information can simplify and speed up the resolution:

- Descriptive/detail name of the issue (avoid general descriptions)
- Installed operating system version
- Installed EEZ Studio version
- Steps to reproduce the problem you are reporting

2.4. Donations

As an open source project, EEZ Studio has been largely developed thanks to donations primarily from NLnet foundation as well as a number of smaller individual donors. If you want to contribute to further development with your donation, you can use Liberapay.

3. Installation

3.1. System requirements

EEZ Studio is a 64-bit application. Therefore the minimum requirement for installation is a personal computer with a 64-bit operating system installed which has enough RAM and disk space for smooth operation.

Installation packages for supported operating systems for all versions of EEZ Studio are available for download at https://github.com/eez-open/studio
It is the official download page and we recommend that you get the latest version for the first installation. You will be able to check for future updates by using the option provided for that, as described below. If EEZ Studio becomes available on the websites of our partners, this information will be published on the Envox official website.

3.2. Linux

Depending on your linux distribution, choose one of the listed packages (.deb, .rpm) and start the installation using the associated installer.
In addition, there is a self-executing .AppImage version that, after downloading, needs to enable the `Allow executing file as program` option under file `Permissions` (Fig. 1) before starting it.
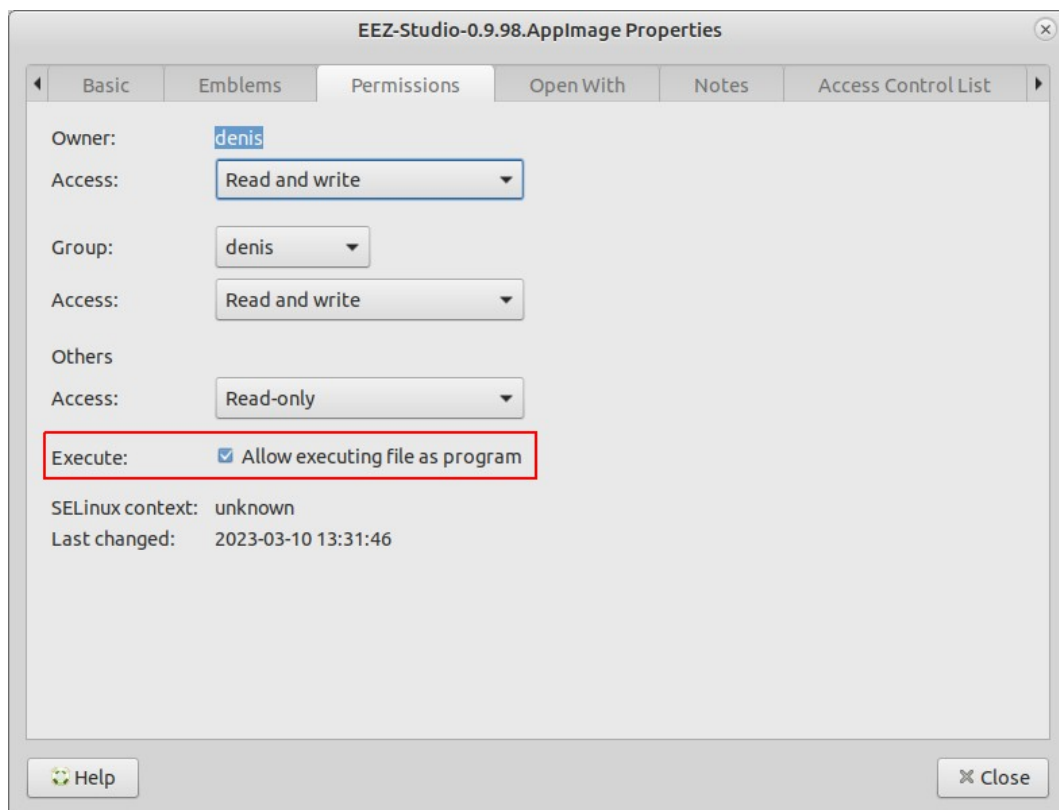

Fig. 1: .AppImage file permission

If you encounter a problem running the .AppImage version on your Linux distribution, try running it using the `--no-sandbox` option: `./EEZ-Studio-[version].AppImage --no-sandbox`

3.3. Mac

Download *eezstudio-mac.zip*, unpack and move *eezstudio.app* to Applications.

3.4. Windows

Download and start *EEZ_Studio_setup.exe*.

3.5. Nix package manager

The Nix [flake](#) provides a derivation for EEZ Studio or an overlay that provides that derivation. It can be used to install the project using [Nix package manager](#).

3.6. Build and run from source (all operating systems)

In addition to using ready-made installation packages, it is possible to build and run EEZ Studio directly from the source code located in the GitHub repository. Below is the procedure to be followed:

- Install *Node.JS 14.x* or newer
- Install *node-gyp*, more information at https://github.com/nodejs/node-gyp#installation

3.6.1. Linux only

```
sudo apt-get install build-essential libudev-dev
```

3.6.2. Raspbian only

Install *Node.js 16* and *npm* on Raspberry Pi: https://lindevs.com/install-node-js-and-npm-on-raspberry-pi/

```
sudo apt-get install build-essential libudev-dev libopenjp2-tools ruby-full
sudo gem install fpm
```

3.6.3. All platforms

In the folder where you want to build the project, it will be necessary to clone the GitHub project repository, and start project building as follows:

```
git clone https://github.com/eez-open/studio
cd studio
npm install
npm run build
```

Start with:

```
npm start
```

Create distribution packages (except [Raspbian](#)):

```
npm run dist
```

3.6.4. On Raspbian:

```
npm run dist-raspbian
```

3.6.5. Nix

To build:

```
nix build 'github:eez-open/studio'
```

To start:

```
nix run 'github:eez-open/studio'
```

3.7. USB TMC

The USB TMC driver must be installed if you want to access the T&M instrument using the USB-TMC interface from EEZ Studio *Instrument* section.

3.7.1. Windows

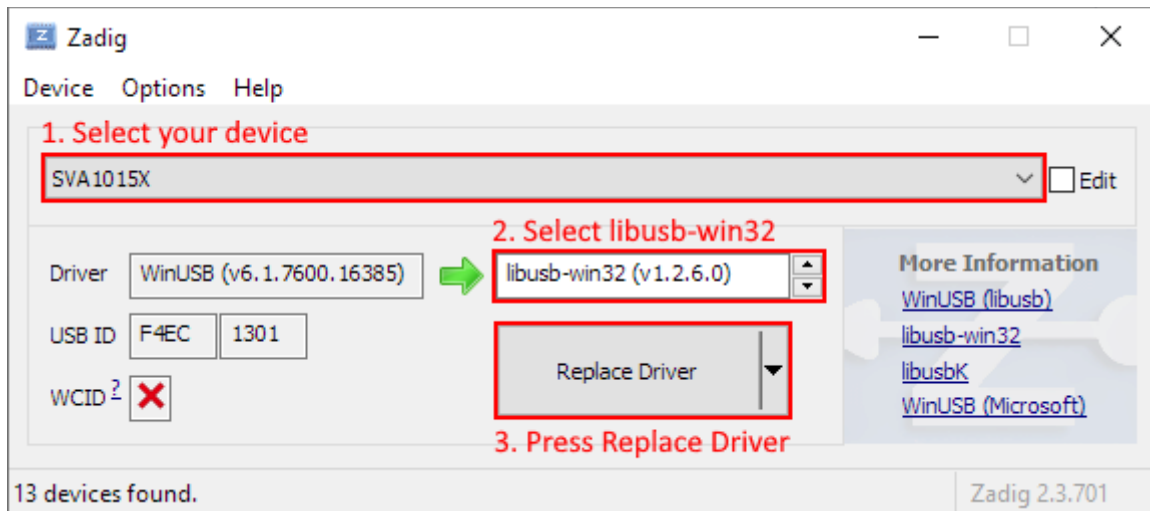Download and start [Zadig](). Select your device, select libusb-win32 and press "Replace Driver" button:

Fig. 2: Zadig driver settings

3.7.2. Linux

You will probably need to add your Linux account to the `usbtmc` group before you can access the instrument using EEZ Studio. Connect your instrument with a USB cable and turn it on. Wait until booting is complete. Now check the instrument group name by entering the following command:

```
ls -l /dev/usbtmc*
```

In case it is *root*, enter the command:

```
sudo groupadd usbtmc
```

Now, add your account (<username>) to the group:

```
sudo usermod -a -G usbtmc <username>
```

A reboot is required. After that, the *gid* of `/dev/usbtmc0` should be set to `usbtmc` and you are ready to use your instrument via USB-TMC interface.

3.8. FAQ

Q: Where is the database file by default?
A: Depending on the operating system, it can be:

- Linux: `~/.config/eezstudio/storage.db`
- Mac: `~/Library/Application\ Support/eezstudio/storage.db`
- Windows: `%appdata%\eezstudio\storage.db`

The default created database as well as its location can be changed later through the options in the *Settings* section of EEZ Studio.

Q: Where are the IEXTs (Instrument EXTensions) used to access T&M instruments stored?
A: Depending on the operating system, it can be:

- Linux: `~/.config/eezstudio/extensions`
- Mac: `~/Library/Application\ Support/eezstudio/extensions`
- Windows: `%appdata%\eezstudio\extensions`