



Loadupch

Release 0.0.1

Unit Electronics

Jun 18, 2024

CONTENTS:

1	Installation	3
1.1	Requirements	3
1.2	Loadupch Installation	3
1.3	Driver Installation	3
1.4	Uninstallation	4
2	Usage	5
3	Upload firmware	7
3.1	Verify firmware loaded	8
4	Examples	9
4.1	Upload Firmware Using Loadupch	9
5	Unpaired Device	13
6	Indices and tables	17

Warning: Loadupch is freely available for use. However, the software is currently under development and may contain bugs.

Loadupch is a software tool specifically designed for uploading firmware to the CH552 microcontroller. Developed in Python, this application leverages the PyUSB library to facilitate communication with the device.

Loadupch is based on the loader implementation by Stefan Wagner, available at [Stefan Wagner's GitHub](#). Originally inspired by the `chprog.py` project found at [chprog.py on GitHub](#), Loadupch enhances the original by introducing a graphical user interface, making it significantly more user-friendly.

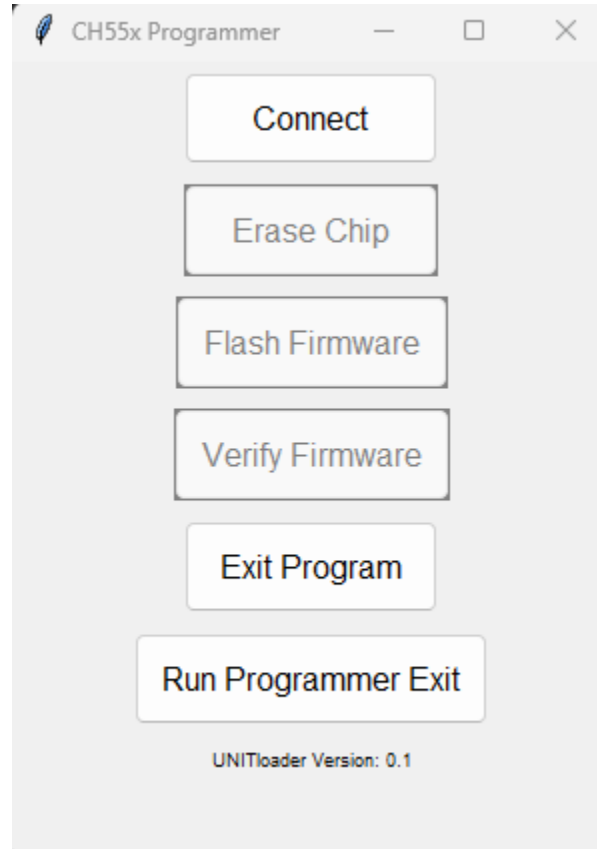


Fig. 1: Loadupch interface

This documentation is divided into the following sections:

INSTALLATION

Warning: Loadupch is freely available for use. However, the software is currently under development and may contain bugs.

1.1 Requirements

Loadupch requires [Python 3.6 or higher](#) and the following libraries:

```
pip install pyusb
```

```
pip install tkinter
```

1.2 Loadupch Installation

For easy installation, you can use the following command:

```
pip install loadupch
```

1.3 Driver Installation

Windows

To install the driver, you can use the Zadig software. Download the latest version of Zadig. You can download it from the official website.

Ubuntu

If you use Linux, you may need to install the following packages:

```
sudo apt-get install libusb-1.0-0-dev libudev-dev
```

1.4 Uninstallation

In the case that software is not useful for you, you can uninstall it using the following command:

```
pip uninstall loadupch
```


USAGE

To use the Loadupch, you can run the following command:

```
python -m loadupch
```

The software will open a window with the interface. You can select the device press the “Connect” button.

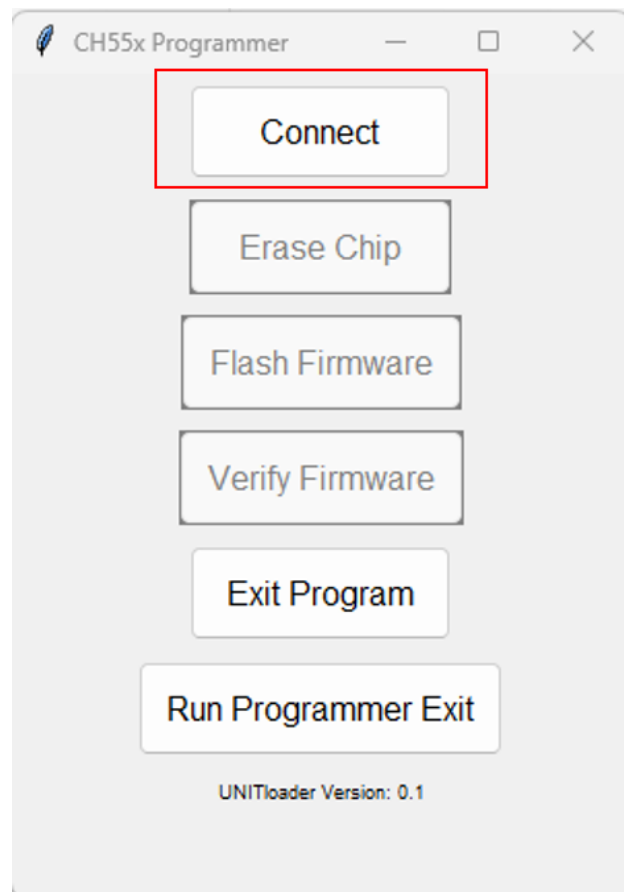


Fig. 2.1: Loadupch interface

Note: Before connect the device, press the “boot” button on the CH552 board.

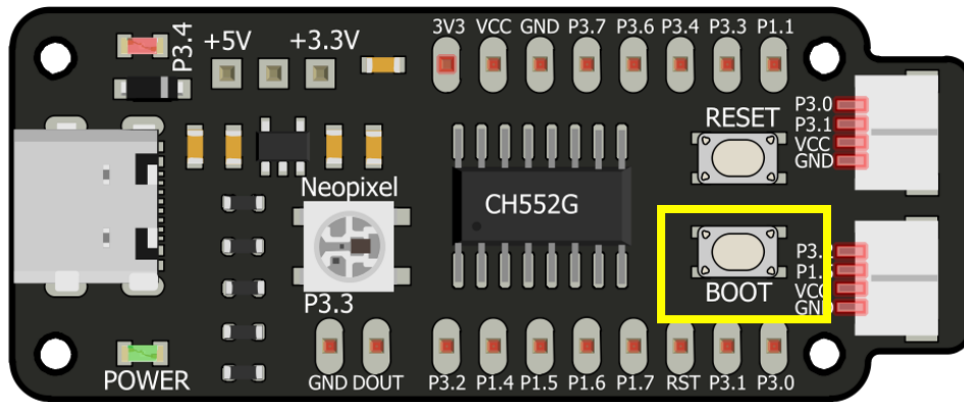


Fig. 2.2: Cocket Nova CH552 button boot

The interface will show the message “Device connected” and you can select the firmware file and press the “flash firmware” button.

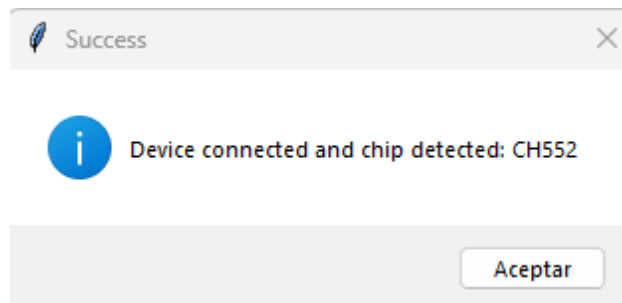


Fig. 2.3: Device found message

UPLOAD FIRMWARE

For the upload firmware, press the “flash firmware” button.

Note: Only .bin files are supported.

Automatically the software open the file dialog, you can select the firmware file and press the “open” button.

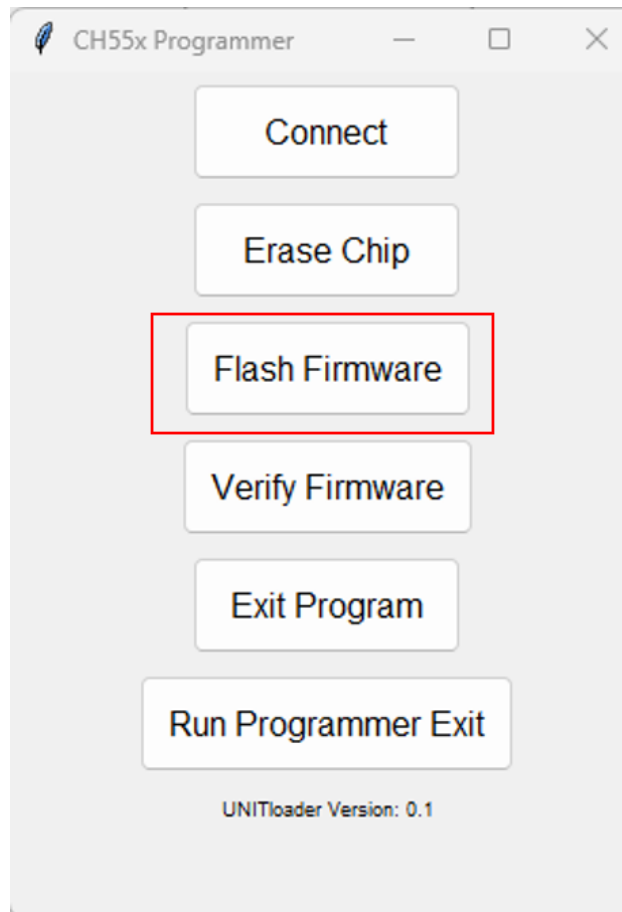


Fig. 3.1: Flash firmware dialog

The software will show the message “Firmware flashed” and the device will be ready to use.



Fig. 3.2: Firmware flashed message

3.1 Verify firmware loaded

To verify the firmware loaded, press the “verify firmware” button.



Fig. 3.3: Verify firmware dialog

The software will show the message “Firmware verified” if the firmware is loaded correctly.

EXAMPLES

Note: There exists a repository with examples for the Cocket Nova CH552 board, which you can find here:

The repository provides examples for developing software in C using the SDCC compiler for the CH552 microcontroller. It serves as an excellent resource for both beginners and experienced developers, offering versatile and affordable solutions for code development.

Clone the repository and follow the instructions to compile and upload the firmware to the CH552 microcontroller:

4.1 Upload Firmware Using Loadupch

To upload the firmware using Loadupch, follow the steps below:

1. Clone the repository with the examples:

```
git clone https://github.com/UNIT-Electronics/CH55x_SDCC_Examples.git
```

2. Open the Loadupch software:

```
python -m loadupch
```

3. Press the `boot` button on the CH552 board and connect the device to the USB port of your computer.
4. Press the `Connect` button on the Loadupch interface.
5. Select the firmware file from the Software/examples/Blink repository and press the `flash firmware` button.
6. The software will show the message `Firmware flashed` and the device will be ready to use.
7. Finally, press the `run programmer Exit` button on the Loadupch interface.

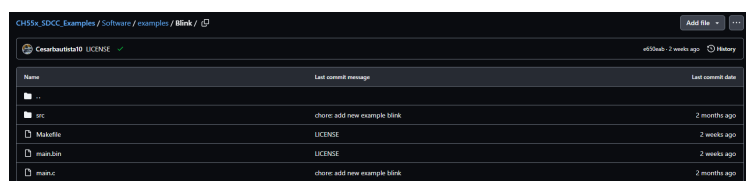
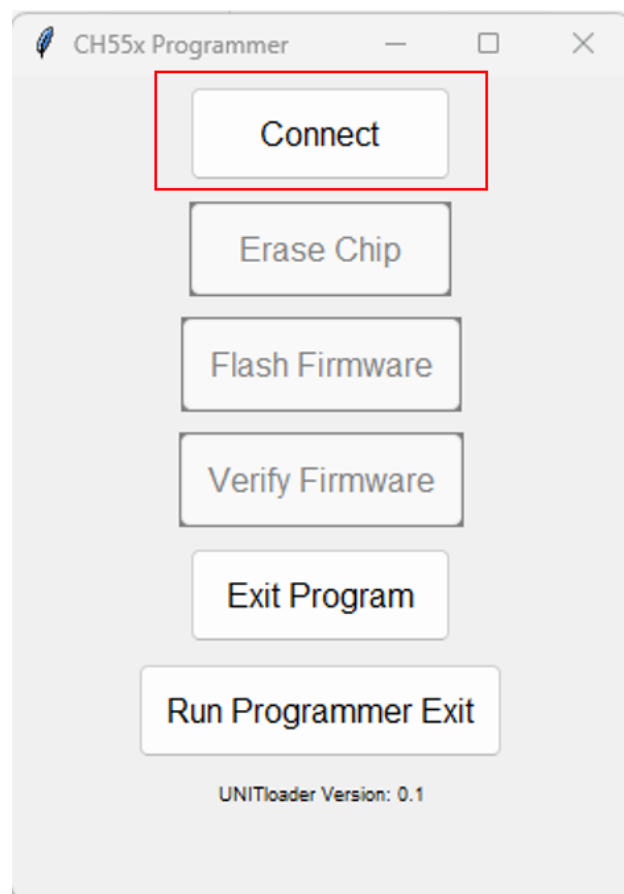
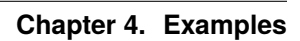


Fig. 4.3: Path to example blink

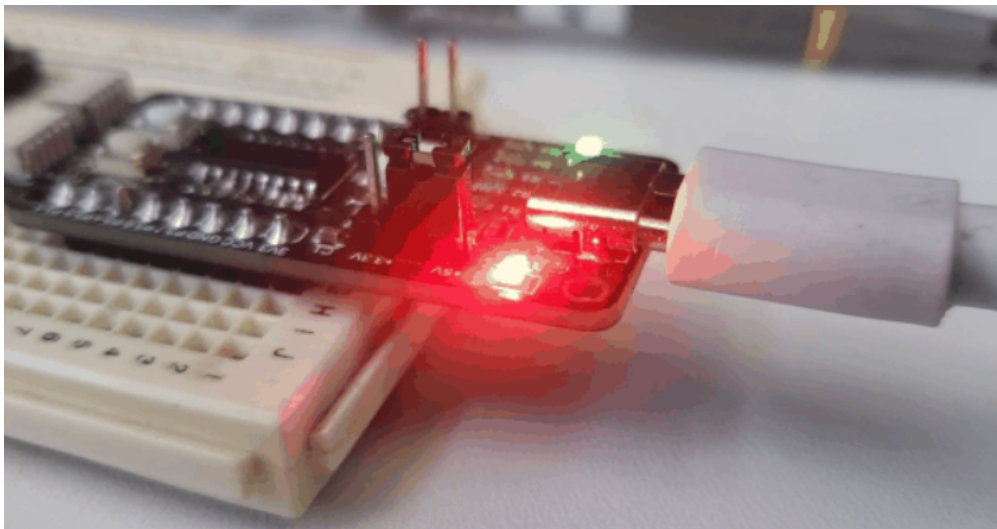


Fig. 4.4: Blink example

UNPAIRED DEVICE

There are several reasons why the device might not be connected:

- The device is not connected to the USB port.
- The device is not in boot mode.
- The system does not have the necessary permissions to access the device.
- The driver is not installed.

To solve the problem, you can try the following steps:

1. Check if the device is connected to the USB port.
2. Check if the device is in boot mode.
3. Check if the system has the necessary permissions to access the device.

If the problem persists, you can try the following steps:

1. Check if the driver is installed.

Use Zadig to install the driver:

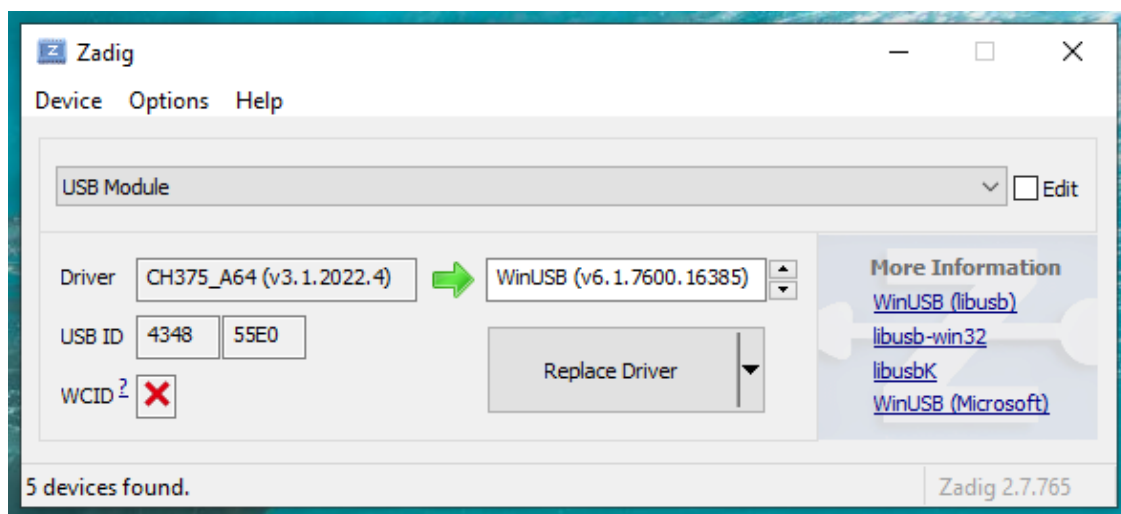


Fig. 5.1: Zadig interface

1. Go to the “Options” menu and select “List All Devices”.
2. Locate the device with USB MODULE VID = 0x4348 and PID = 0x55E0.
3. Select the driver “libusb-win32 (v1.2.6.0)” and press the “Install Driver” button.

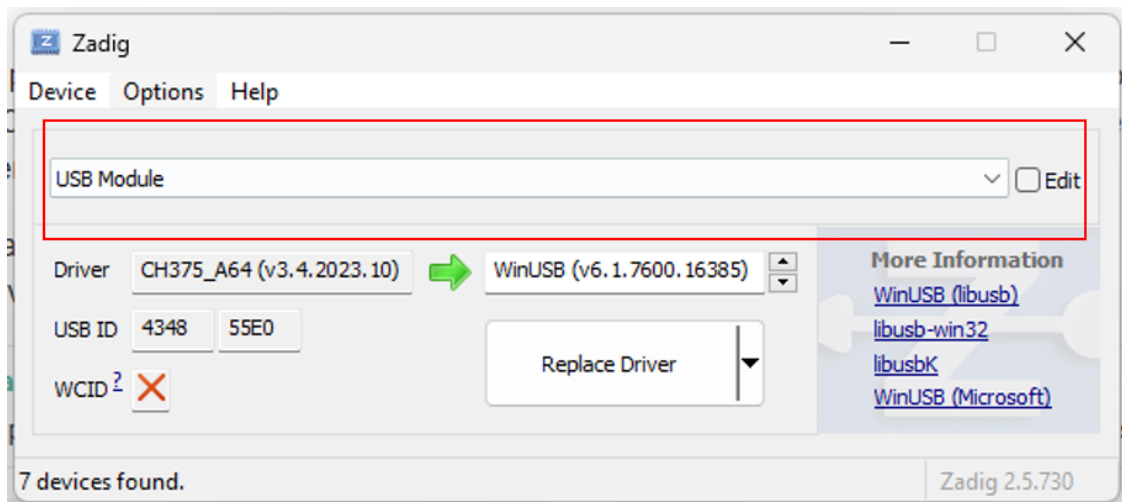


Fig. 5.2: Device selected

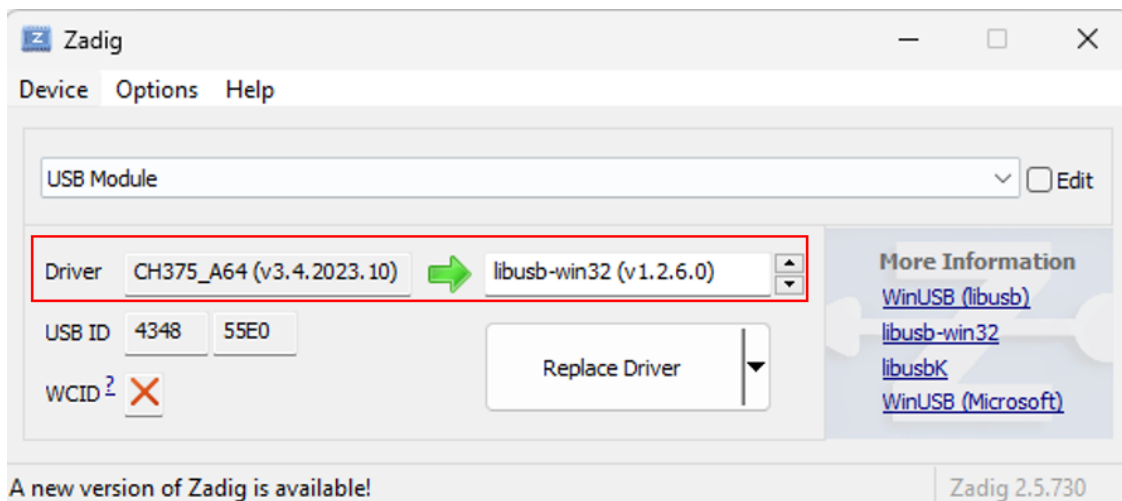


Fig. 5.3: Driver selected

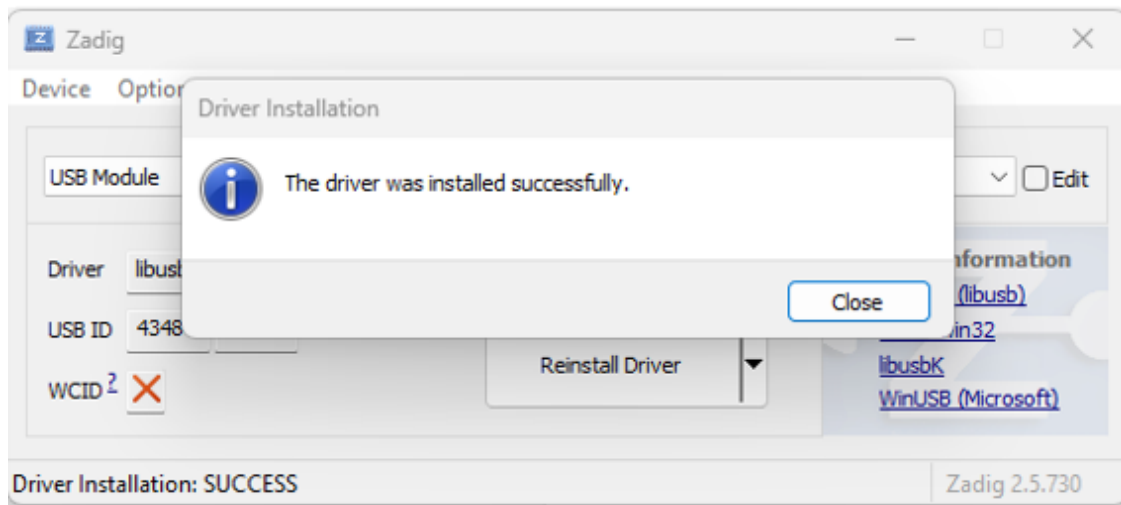


Fig. 5.4: Device installed

After that, the device will be recognized by the system.

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`