

EPub-InkPlate - Installation Guide

Please Note: if you are updating from a version prior to V2.0 of the application, the SD-Card must be re-initialized. See the section on preparing the SD-Card for all the details.

Here is the installation procedure for the EPub-InkPlate application. This procedure can be adapted depending on your requirements.

The installation consists of

- Preparing your computer with the proper applications needed for the installation process.
- Preparing an SD-Card with the appropriate information.
- Uploading the application to the InkPlate device.

The last version of the binaries for the Inkplate are located in release bundles that you will find with the application GitHub project: “<https://github.com/turgu1/EPub-InkPlate/releases>”. This procedure shows how to install it using the *esptool* upload tool. This is the simplest way to install EPub-InkPlate as it does not require to have a full development environment (VSCode + PlatformIO + ESP_IDF) to install the binary version.

(You can also compile and upload the result within a VSCode/PlatformIO development environment. The supplied `platformio.ini` file is already set up such that once the project is loaded into the IDE, you can launch the builder and the uploader.)

Prerequisite

The *esptool* is a Python program that is used to upload an application to an ESP32 (or ESP8266) device. It must be installed on your computer. It is compatible with both *Python* versions 3.7 or newer. Verify that you have *Python* and *pip* installed on your computer (The following link may be useful: “<https://wiki.python.org/moin/BeginnersGuide/Download>”). Then, on Windows, Linux or MacOS, to install *esptool*, the following command must be executed (in a shell window):

```
pip install esptool
```

For more information, please consult the following link: “<https://docs.espressif.com/projects/esptool/en/latest/>”.

The InkPlate device uses a CH340 USB to UART converter. If your computer doesn’t have a driver installed for the CH340, you have to install one. Look at this location for the proper procedure to install it: “<https://soldered.com/learn/ch340-driver-installation-croduino-basic3-nova2/>”.

You then must retrieve the release from the Github repository. Look at this location on GitHub: “<https://github.com/turgu1/EPub-InkPlate/releases>”. The file to download is **release-X.X.X-inkplatezip**. It is located in the **assets**, down under the description text. Extract its content. You will get two folders: **bin** and **SDCard**, the installation document, and the user’s guide document in PDF format.

Preparing the SD-Card

The SD-Card must be formatted with a FAT32 (or MS-DOS or VFAT) partition. This is usually the case with brand new cards. Do not use ExFAT as it is not supported by the application. The

release's **SDCard** folder contains everything required to initialize the card's content. Simply copy the content of the folder (including the sub-folders) to the card as is.

The config.txt file The file **config.txt** located in the card's root folder may be edited to identify your wifi parameters (**wifi_ssid**, **wifi_pwd**, **http_port**), timezone information (**tz**), and Internet date/time server address (**ntp_server**). As these parameters contain text information or number, they are not editable through the EPub-InkPlate application forms. This file is loaded at startup.

The wifi parameters will permit the application to access your local network. This will allow for accessing the InkPlate device from a Web browser to manage the list of books present on the card. This is optional as it's always possible to update the SD-Card content by inserting it into your computer.

Wifi will also be used to access a date/time Internet server to adjust the device clock. A default NTP Server address is already identified (**ntp_server** parameter) and be sufficient for most usage. It can be replaced by your selection. Please note that the date/time adjustment through the Internet must be manually started to update the internal clock. More information is available in the User's Guide.

The timezone information (**tz** parameter) is a specially formatted string supplying proper information to translate the internal clock value (that is managed as a Greenwich UTC value) to the local time value. A file named **timezones.csv** is supplied with the Epub-Inkplate distribution. It contains the timezone values for more than 400 cities in the world. For example, for Zagreb, the timezone value would be "CET-1CEST,M3.5.0,M10.5.0/3". The actual format is described here: "https://www.gnu.org/software/libc/manual/html_node/TZ-Variable.html".

The file **config.txt** contains other parameters that are managed through the application forms. Please do not modify their content by hand. If those parameters are deleted from the file, they will be re-initialized with default values by the application.

Your books As indicated in the previous section, you can add books to the device using the supplied Web server and your Internet browser. Starting the server is described in the User's Guide.

You can also add your books by hand. All books must be in the EPub format and have a filename with the extension **.epub** in lowercase. They must be located in the **books** folder of the SD-Card. This is done by inserting the SD-Card into your computer to access it directly.

If the SD-Card has already been used by the application, you may find other files present in the **books** folder. They are used to manage various information about your interaction with the books. If deleted, these files will be re-initialized by the application at the proper time.

Once done, insert the card into the device.

Uploading the application program

The release's **bin** folder contains the application, the bootloader, and the partitions binaries that must be downloaded to the device. To do so connects the device to a USB port, turn it on, change your current directory to that folder, and execute the following command:

On Linux or MacOS (in a shell window):

```
$ sh upload.sh
```

On MS Windows:

```
.\upload.bat
```

Here is an example output of the execution:

```
$ sh upload.sh
esptool.py v3.0
Serial port /dev/ttyUSB0
Connecting.....
Chip is ESP32-D0WDQ6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse,
  > Coding Scheme None
Crystal is 40MHz
MAC: fc:f5:c4:1b:4e:cc
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 230400
Changed.
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 25136 bytes to 15148...
Wrote 25136 bytes (15148 compressed) at 0x00001000 in 0.7 seconds
  > (effective 297.9 kbit/s)...
Hash of data verified.
Compressed 3072 bytes to 143...
Wrote 3072 bytes (143 compressed) at 0x00008000 in 0.0 seconds
  > (effective 2244.8 kbit/s)...
Hash of data verified.
Compressed 1086128 bytes to 554716...
Wrote 1086128 bytes (554716 compressed) at 0x00010000 in 24.9 seconds
  > (effective 348.5 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
$
```

Once the upload is complete, the device will automatically reboot. Please look in the user's guide for information on how to interact with the application.

Some options on the esptool command may have to be modified depending on your computer:

- The USB device connected to the InkPlate is expected to be named `/dev/ttyUSB0` (That is the case on Linux Mint and Ubuntu) on Linux and MacOS, or `COM3` on MS Windows. If it's

not the case, you must find it and modify the `upload.sh` script for Linux and MacOS, or `upload.bat` in MS Windows.

- Another issue you may have is the download speed that is too high for your computer. Again, you may change it in the `upload.sh` script (or `upload.bat`). The speed (baud rate) is **230400** in the file. You can change it to **115200** baud or lower.