



TEKTELIC Communications Inc.
7657 10th Street NE Calgary, Alberta
Canada, T2E 8X2

USER GUIDE

Document Type:	User Guide
Document Number:	T0006377_UG
Document Version:	0.7
Product Name:	Custom Display Tablet
Product Codes & Revisions:	T0006086 –Battery Powered Digital Signage, NA T0006093 – Externally Powered Digital Signage, NA T0006749 –Battery Powered Digital Signage, EU T0006750 – Externally Powered Digital Signage, EU
Issue Date:	April 04, 2022

Revision History

Version	Date	Editor	Comments
0.1	July 13, 2021	T Makarov	Initial Draft
0.2	July 23, 2021	T Makarov	General updates and template updates
0.3	August 20, 2021	T Makarov	Custom Configurations section updates
0.4	August 27, 2021	T Makarov	Uplink Configuration File section updates
0.5	September 23, 2021	T Makarov	Custom Configurations section updates
0.6	September 28, 2021	T Makarov	Custom Configurations section updates
0.7	April 04, 2022	T Makarov	Custom Configurations section updates

Table of Contents

List of Tables	5
List of Figures	6
1 Product Description	7
1.1 Overview	7
1.2 Specifications.....	8
1.3 Physical Interfaces.....	10
2 Commissioning	12
2.1 Power Up/Down Procedure	12
3 Installation	13
3.1 Included Product and Installation Material	13
3.2 Safety Precautions.....	13
3.3 Unpacking and Inspection	13
3.4 Equipment Required for Installation.....	14
3.5 Custom Display Tablet mounting	14
3.6 Battery Replacement.....	16
4 Custom Configurations	17
4.1 Screen Element Configuration File.....	17
4.2 Example Screen Element Configuration File	20
4.3 Text Configuration File	23
4.4 Example Text Configuration File	24
4.5 Style Configuration File	25
4.6 Example Style Configuration File.....	27
4.7 Uplink Configuration File.....	28
4.8 Example Uplink Configuration File	30
4.9 Downlink Configuration File.....	32
4.10 Example Downlink Configuration File	33
4.11 Image Files	34
4.12 Online Image Converter	35
4.13 Font Files.....	36

4.14	Offline Font Converter	36
5	Operation, Alarms, and Management.....	39
5.1	Configuration.....	39
5.2	Default Configuration.....	39
6	Firmware Upgrade Feature	40
6.1	Overview	40
6.2	Setup/Requirements	40
6.3	Steps to Complete Firmware Upgrade.....	41
6.4	Touchscreen Firmware Upgrade	43
7	Compliance Statements.....	45
8	Bibliography	47

List of Tables

Table 1: Custom Display Tablet Models	7
Table 2: Custom Display Tablet Specifications	9

List of Figures

Figure 1-1: Custom Display Tablet	8
Figure 1-2: The Tablet (external) interface layout.	10
Figure 1-3: The Tablet (Battery) interface layout.	11
Figure 2-1: Custom Display Tablet's infrastructure	12
Figure 3-1: Wall mount plate.	15
Figure 3-2: Tablet back side	16
Figure 4-1: Online Image Converter	35
Figure 6-1: USB A to Micro-B Cable	40
Figure 6-2 Tablet with power plug and USB cable connected	40
Figure 6-3 Com Power Selection	41
Figure 6-4 Serial Port and Speed Selection	41
Figure 6-5 Ready for Firmware Note	42
Figure 6-6 Progress Bar	42
Figure 6-7 Firmware file upload aborted	43
Figure 6-8 Firmware upgrade successful	43

1 Product Description

1.1 Overview

The Custom Display Tablet is a LoRaWAN enabled interactive signboard. The Tablet features 6" E Ink screen with capacitive touch, Front light, Accelerometer and Battery monitor. Table 1 presents the Custom Display Tablet models.

Table 1: Custom Display Tablet Models

Product Code	Description	RF Region	Tx Band (MHz)	Rx Band (MHz)
T0006086	MODULE, DIGITAL SIGNAGE, BATTERY POWERED, NA	US915	923-928	902-915
T0006093	MODULE, DIGITAL SIGNAGE, EXTERNALLY POWERED, NA	US915	923-928	902-915
T0006749	MODULE, DIGITAL SIGNAGE, BATTERY POWERED, EU	EU868	863-870	863-870
T0006750	MODULE, DIGITAL SIGNAGE, EXTERNALLY POWERED, EU	EU868	863-870	863-870

The main features of the Custom Display Tablet (Tablet) are the following:

- **6" E Ink screen:** 1024(H) x 758(V) pixels with 16 levels of gray.
- **Touch screen:** Capacitive touch screen.
- **Front light:** Uniform front lighting for low light environment.
- **LoRa:** Air interface capable of long range at low power.
- **Battery Powered (option):** Powered by 4xAA, Up to 1-year battery life.
- **Externally Powered (option):** 5V DC or PoE (48V) powering option.
- **Easy Installation:** Removable wall mount plate simplifies the installation and alignment of the Tablet.
- **Safety Screw:** Hidden screw on the top that requires a special tool to unlock ensuring the devices safety.
- **Hidden Cables:** If powering the device externally there are grooves to hide the cables leaving a clean finish.

- **Landscape or Portrait mode¹:** The device can be mounted in horizontal or vertical orientation.
- **Battery Monitor:** Monitors the battery level and provides a low battery warning for timely replacement.
- **Deep Sleep mode:** Accelerometer allows the device to save power when the Tablet is not in use and wake up on when a Double-tap is detected on the screen.

Figure 1-1 illustrates the Custom Display Tablet.



Figure 1-1: Custom Display Tablet

1.2 Specifications

The Custom Display Tablet specifications are listed in Table 2.

¹ Portrait mode is not supported in current version of firmware
Custom Display Tablet User Guide
TEKTELIC Communications Inc.

Table 2: Custom Display Tablet Specifications

Attribute	Specification
Use Environment	Indoor only
Enclosure	Plastic, IP30
Operating Temperature	5°C to 40°C
Storage Temperature	-25°C to 70°C
Operating Relative Humidity	5% to 95%, condensing
Size	4.5 (L) x 6 (W) x 1(H) inch
Weight	340g(0.75lb) with batteries 280g(0.62lb) without batteries
Display	Size: 6-inch E Ink screen (3:4) Resolution: 1024(H) x 758(V) pixels Color: 16 levels of gray (monochrome) Front light
Touch screen	Capacitive touch screen ± 5 mm accuracy
Power Source	Externally powered Option: -DC 5V -PoE 48VDC (IEEE 802.3af Mode A or B or 4-pair Mode) Battery powered option: -4x AA Lithium Batteries -Front light and RGB LEDs not supported
Power Consumption	3 W maximum
Battery Lifetime	1-year battery life for typical use case ²
Network technology/Frequency band	LoRaWAN in several variants (Table 1): US915, EU868
Air Interface	LoRa
Maximum Tx Power	15 dBm
Ethernet	TBD
USB MicroB	USB2.0 Debug port
Temperature Measurement Accuracy	<±5°C

² Active 10 hour/week day (280 Rx packets, 15 Tx packets at 15dBm, 15 screen updates), in Deep sleep otherwise

1.3 Physical Interfaces

Figure 1-2 and Figure 1-3 illustrates the customer accessible interfaces of the Tablet. All models share the same layout; however, some functions are not available in some models. The externally powered model shown in Figure 1-2 has RGB LEDs on the corners while the battery powered version shown in Figure 1-3 cannot use the LEDs.

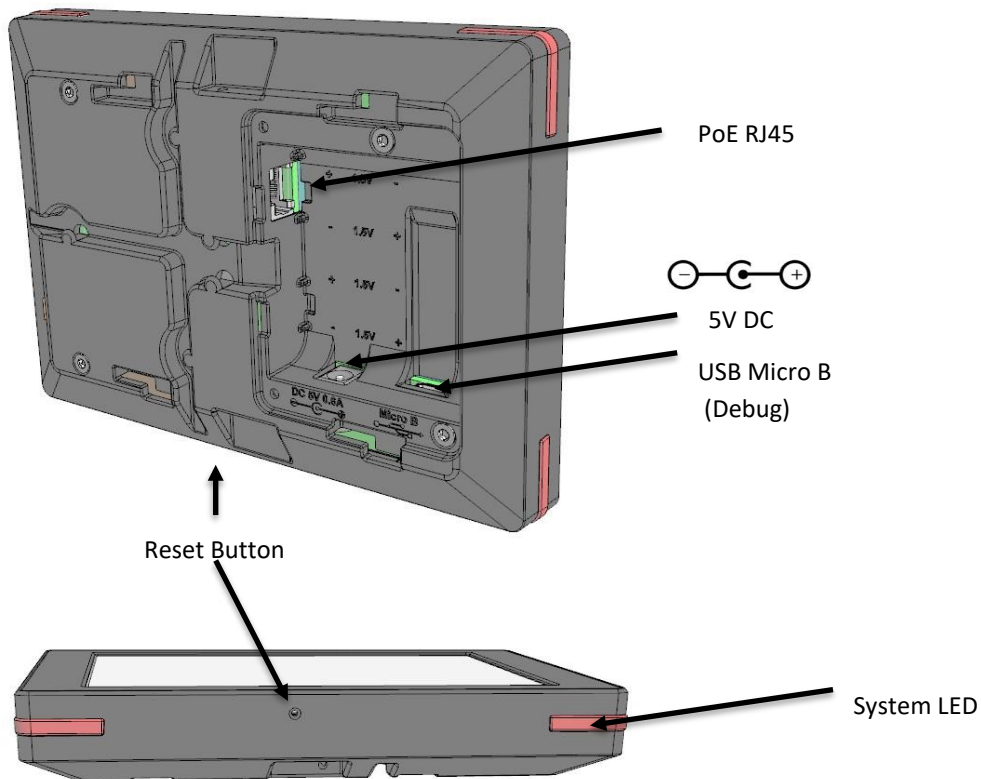


Figure 1-2: The Tablet (external) interface layout.

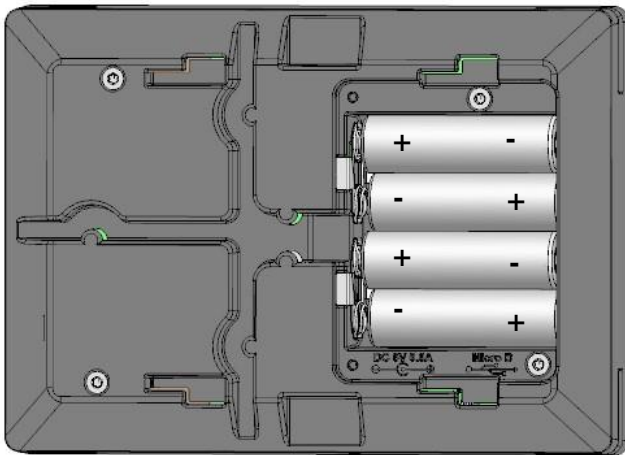
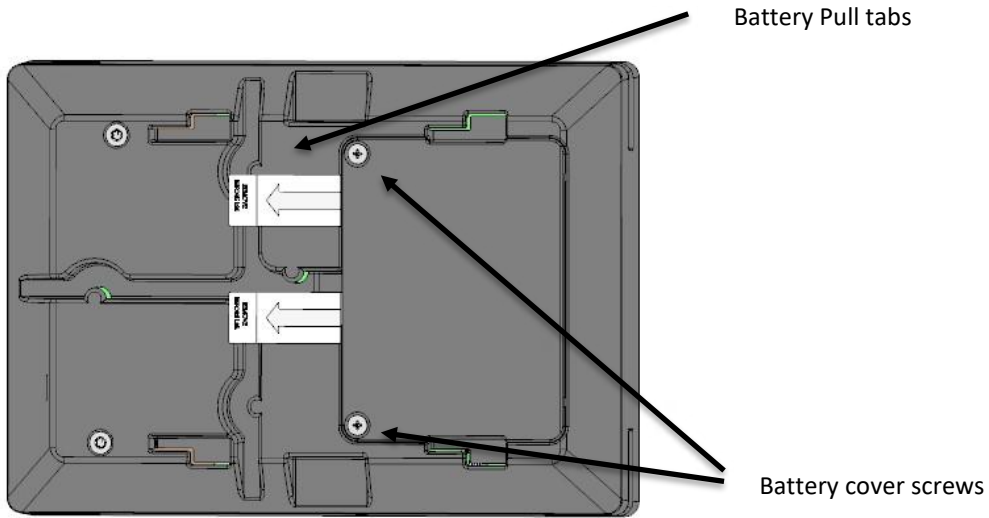


Figure 1-3: The Tablet (Battery) interface layout.

2 Commissioning

The steps for commissioning the Custom Display Tablet are managed through a versatile Application. Application or user registers the Tablet onto the Network server using the unique identifier (EUI) of the Tablet. A high-level diagram of the Custom Display Tablet's infrastructure is shown in Figure 2-1.

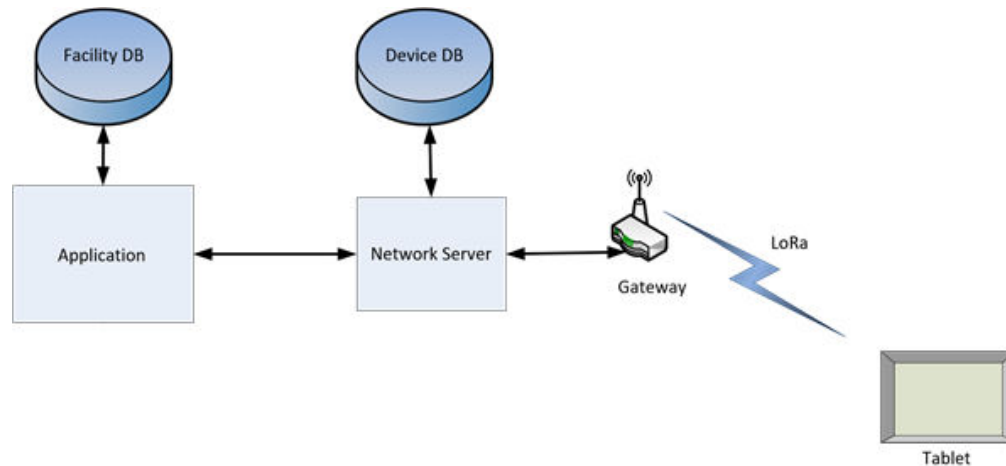


Figure 2-1: Custom Display Tablet's infrastructure

2.1 Power Up/Down Procedure

- Battery powered version of Tablet is shipped with the batteries installed along with pull tabs that prevent the Tablet from turning on during shipment.
- In case of necessity to apply custom configurations, images, fonts or firmware for touch controller (see section 4 and 6.4 for details):
 - Remove batteries from Tablet.
 - Plug connected from one side to computer through USB cable to Tablet USB Micro B and press Reset Button (see Figure 1-2).
 - Wait till the system on computer connect Tablet as Mass Storage Device.
 - Add or update configuration files, images, fonts and file with firmware for touch controller on Tablet.
 - Unplug USB cable from Tablet.
- Once the Tablet is configured with configuration files and on the Application, turn on the Tablet by removing the battery pull tabs or providing external power depending on the model and press Reset Button.
- To turn off the Tablet the batteries or external power must be removed. The unit must remain un-powered for 1 minute to completely reset.
- The reset switch of the Tablet is accessible through a pin hole in the bottom side of the Tablet as shown in Figure 1-2.

3 Installation

3.1 Included Product and Installation Material

The following items are shipped with each Tablet:

- Custom Display Tablet
- 5V AC-DC power adapter (optional)
- Four AA batteries (optional)
- Wall mount plate
- Security/Lock screw (T6)
- Product Manual

3.2 Safety Precautions

The following safety precautions should be observed:

- The Tablet is for indoor use only. Do not connect Tablet to any outdoor cables.
- The Tablet has no internal field serviceable parts other than the batteries. Other than installing or replacing the batteries, the Tablet must only be opened by an approved TEKTELIC service center.
- All installation practices must be in accordance with the local and national electrical codes.
- Ensure that the Tablet is located to eliminate any physical hazard to people or property.
- The Tablet shall be powered from the supplied AC-DC power adaptor or through Power over Ethernet (PoE) or 4xAA batteries. Simultaneous application of power through more than one input may result in unexpected operation and shall be avoided.
- Keep batteries away from the reach of children.
- Do not mix old and new batteries.
- If the Tablet is not expected to be used for extended periods of time, the batteries should be removed before storage to avoid any leak.

3.3 Unpacking and Inspection

The following should be considered during the unpacking of a new Custom Display Tablet:

- Inspect the shipping carton and report any significant damage to TEKTELIC.
- Unpacking should be conducted in a clean and dry location.

- Do not discard the shipping box or inserts as they will be required if a unit is returned for repair.

3.4 Equipment Required for Installation

The following tools are required to install the Custom Display Tablet:

- 1) Screwdriver
- 2) 4x M4 Screws (Choose screw type based on mounting surface)
- 3) Spirit level
- 4) T6 Torx driver for the security/lock screw

3.5 Custom Display Tablet mounting

Custom Display Tablet features a removable mounting plate that simplifies the installation and leveling of the Tablet. Once the mount plate is installed the Tablet slides onto the four hooks on the plate. Tablet can be mounted in Landscape or Portrait³ mode.

Note: The mounting surface must be capable of holding > 15 kg [33 lbs].

Mounting the wall plate

- Thread the Ethernet and/or Power cables through the rectangular opening in the wall plate. Ensure that the lock screw feature marked as “C” in Figure 3-1 is facing away from the mounting surface.
- Secure the mount plate on the mounting surface using 4 x M4 screws in locations A or B as shown in the Figure 3-1.
 - Double sided adhesive tape could be used when mounting on glass
- The slots provided on mount plate allows for adjustment to level the mount plate.
- Using a spirit level, ensure that the mounting plate is plumb before tightening all the screws.

Connect cables

- With the externally powered version, connect the necessary cables to the Tablet (Ethernet, 5V DC and Debug USB port are accessible under the battery cover).
- No cables are required on the battery powered version.
- Cables can be routed through the channels on the back of the Tablet.

³ Portrait mode is not supported in current version of firmware

Install the Tablet

- Take a note of the MAC-ID/ EUI shown on the label on the back of the Tablet.
- To install the Tablet, first align the hooks on the mounting plate to the slots on the back of the Tablet (See features marked as “D” in Figure 3-1 and Figure 3-2), then push the Tablet onto the plate. Slide the Tablet to fix the Tablet in place.
- It is recommended to proceed with the commissioning steps detailed in Commissioning before completing the final step of securing the Tablet with the lock screw.

Secure the Tablet using the lock screw

- With the battery powered version, ensure that the battery shipment tabs are pulled out before securing the Tablet.
- A T6 Torx screw is provided for the security lock. Tighten the lock screw using hand to secure the Tablet.

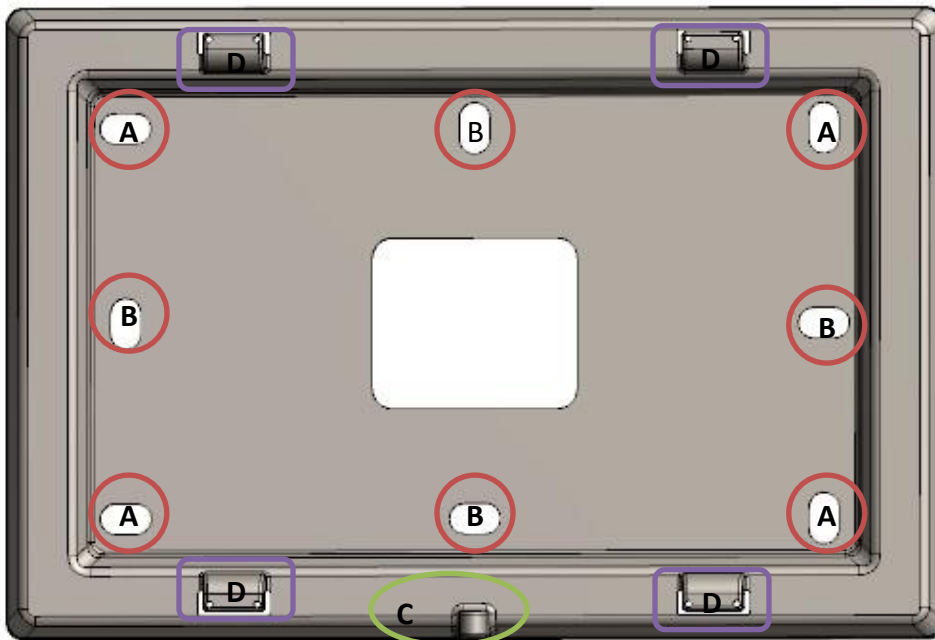


Figure 3-1: Wall mount plate.

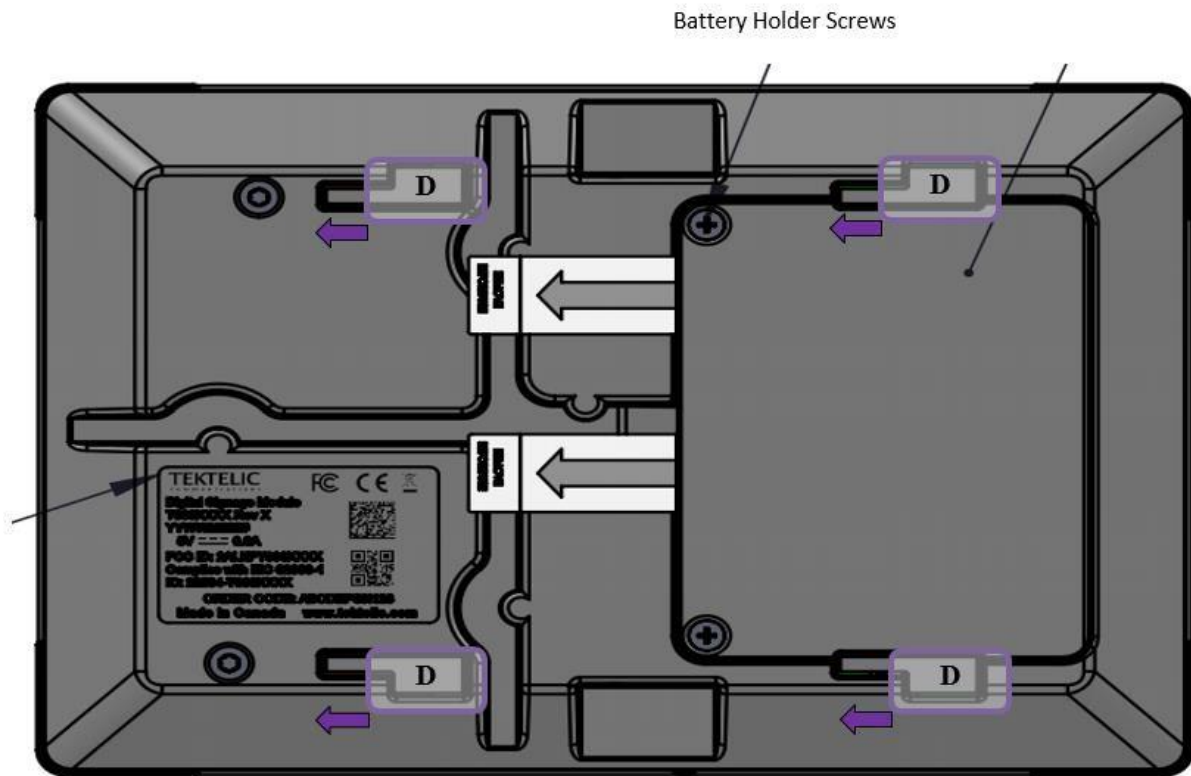


Figure 3-2: Tablet back side

3.6 Battery Replacement

The following tools are required to install the Custom Display Tablet:

- 1) A Philips Screwdriver (#2)
- 2) 4x AA batteries (LiFES2 – Energizer, part number: L91) Steps for replacing the batteries are as follows:

- Removed the battery cover by removing two philips screws shown in Figure 1-3
- Remove all the old batteries from the device.
- To turn off the device the batteries must be uninstalled and remain un-powered for 1 minute.
- Replace with all new batteries of the recommended type.
- Install the battery cover and philips screws to secure the batteries.
- Recycle the used batteries when possible by dropping them off at a participating retailer.

4 Custom Configurations

It is allowed to configurate Tablet graphical user interface, styles, text constants, images, fonts, uplinks and downlinks, firmware for touch controller. Most of those configurations are done by configuration files in YAML format. All configuration and image, font or touch controller firmware files should be upload on Tablet before startup (see section 2.1 for details). It is required to reboot Tablet with Reset Button (see Figure 1-2) after new files uploading.

Configuration and image, font and touch controller firmware files are discussed next in this section.

IMPORTANT: Do not use symbols “#”, “!” and “@” which are special symbols in YAML (for example “#” is a symbol of the beginning of commentary in YAML) for values! This may cause to unrecognizable configuration file or its part. You may use “!” and “@” symbols (but not “#” symbol) for string values inside double quotes. For example:

```
value: "Do not enter!"  
value: "mail@mail.com"
```

4.1 Screen Element Configuration File

To configure screen elements, it is required to use elements.yml (this name is mandatory) configuration file. Screen element configuration file is based on YAML language. The structure of this file is:

```
---  
Elements:  
-  
  <screen element 1 configurations>  
-  
  <screen element 2 configurations>  
...  
-  
  <screen element N configurations>
```

<screen element 1 configurations>, <screen element 2 configurations> and <screen element N configurations> – sets of configurations for screen elements 1, 2 and N respectively.

This document starts with “---”.

“Elements:” is on the next line.

The configurations of the screen elements are placed after “Elements:” line.

The configurations of every screen element start with “-” on separated line.

Mandatory fields for all screen element configurations are:

- *id* – unique identifier of the screen element. There are predefined identifiers of the screen elements in Tablet. Possible values of *id* field are (any other value will be ignored by Tablet): *lab_productName*, *lab_fwVer*, *lab_bootloadVer*, *lab_apploadVer*, *lab_iteFwVer*, *lab_loraStatus*, *img_logoBlack*, *pnl_bottom* for *init_scr* (see section 4.9 for details about identifiers of screens) and for rest of screens it is used such format of *id* `<type symbol>_custScr<number of the screen>_<number of the element with this type on the screen>` (for example *lab_custScr1_1*, *img_custScr3_2*), where:
 - `<type symbol>` – first 3 letters show the type of element (*lab* for labels, *but* – button, *pnl* – panel, *img* – image).
 - `<number of the screen>` – shows the number of custom screen (from 1 to 4).
 - `<number of the element with this type on the screen>` – shows the number of current screen element with current type on current screen (from 1 to 10 for labels, 1 – 6 for buttons, 1 – 5 for panels and 1 – 5 for images).

If there are more than one set of configurations for screen element with the same value of *id* field then Tablet will use configurations from the last set of configurations with such value in *id* field.

- *type* – type of screen element. Possible values of *type* field are (any other value will be ignored by Tablet):
 - *label* – text label (the value of screen element *id* includes “*lab*” at the begin).
 - *button* – screen button (the value of screen element *id* includes “*but*” at the begin).
 - *panel* – rectangular area of the screen (the value of screen element *id* includes “*pnl*” at the begin).
 - *image* – image from Tablet disk drive (the value of screen element *id* includes “*img*” at the begin).
- *x* – horizontal coordinate of the left bottom corner of the screen element (coordinates start from the left top edge of the screen). The value of this field is limited by the resolution of the screen (value should be in 0 – 1024 range).
- *y* – vertical coordinate of the left bottom corner of the screen element (coordinates start from the left top edge of the screen). The value of this field is limited by the resolution of the screen (value should be in 0 – 768 range).
- *visible* – screen element visibility state. Possible values for this field are:
 - *0* – screen element should be invisible.

- 1 – screen element should be visible.

Optional fields are related to screen element type (that is set by field *type*).

Optional fields for the label are:

- *width* – the width of the label in pixels. This field is optional. If this field is not set then width of the label will set automatically equal to the width of content (text).
- *style_id* – identifier of styles that are set in style configuration file *styles.yml* (see section 4.5 for details). Possible values of *style_id* field are (any other value will be ignored by Tablet) values of the *id* fields from style configuration file.
- *font* – the name of the font file that is the source of font. The length of the name of the file has to be not more than 8 symbols and the length of the file extension has to be not more than 3 symbols. This file has to be in binary format (see section 4.13 for details). Tablet uses default font in case if this field is not set or Tablet could not upload the font that is set in this field.

NOTE: Default font is font with height size equal to 20. It is support Basic Latin, Latin-1 Supplement, Cyrillic, Arabic, Arabic Presentation Forms-A and Arabic Presentation Forms-B symbols.

- *text_id* – identifier of the text that is viewed by label. Those identifiers are related to *id* field in *text.yml* file (see section 4.3 for details).
- *calc_id* – identifier of the calculations the result of which is viewed by label. If *calc_id* and *text_id* both are present in configurations of the same label then *text_id* is as more priority field is used and *calc_id* field is ignored. Possible values of *calc_id* field are (other value will be ignored by Tablet):
 - *regStr_calc* – firmware version with regional belongings string.
 - *bootloadVer_calc* – bootloader version string.
 - *apploadVer_calc* – application loader version string.
 - *iteVer_calc* – ITE firmware version string.
 - *loraStatus_calc* – the state of the connection to LoRa network string.
 - *tsFwVer_calc* – touch screen firmware version string.

Optional fields for the panel are:

- *style_id* – identifier of styles that are set in style configuration file *styles.yml* (see 4.5 for details). Possible values of *style_id* field are the same as for labels.
- *width* – the width of the panel in pixels.
- *height* – the height of the panel in pixels.

Optional field for the image is:

- *image* – the name of the image file that is the source of image. The length of the name of the file has to be not more than 8 symbols and the length of the file extension has to be not more than 3 symbols. This file has to be in binary format (see section 4.11 for details).

Optional fields for the buttons are:

- *style_id* – identifier of styles that are set in style configuration file styles.yml (see 4.5 for details) for button. Possible values of *style_id* field are the same as for labels.
- *label_style* – identifier of styles that are set in style configuration file styles.yml (see 4.5 for details) for label that is viewed inside of button. Possible values of *style_id* field are the same as for labels.
- *font* – the name of the font file that is the source of font for label viewed inside of the button. The length of the name of the file has to be not more than 8 symbols and the length of the file extension has to be not more than 3 symbols. This file has to be in binary format (see section 4.13 for details). Tablet uses default font in case if this field is not set or Tablet could not upload the font that is set in this field.
NOTE: Default font is font with height size equal to 20. It is support Basic Latin, Latin-1 Supplement, Cyrillic, Arabic, Arabic Presentation Forms-A and Arabic Presentation Forms-B symbols.
- *width* – the width of the button in pixels.
- *height* – the height of the button in pixels.
- *text_id* – identifier of the text that is viewed by label that is placed inside of the button. Possible values are the same as for labels.

4.2 Example Screen Element Configuration File

Screen element configuration file example is:

```
---
Elements:

# Product Name label
-
  id: lab_productName
  type: label
  style_id: blackLab_stl
  font: chivo40.bin
  x: 0
  y: 135
  width: 1024
  visible: 1
```

```
text_id: meetingRoomDisplayTablet_txt

# Company logo black image
-
  id: img_logoBlack
  type: image
  x: 681
  y: 50
  image: LOGO_B.bin
  visible: 1

# Bottom panel
-
  id: pnl_bottom
  type: panel
  style_id: btnPanel_stl
  x: 0
  y: 586
  width: 1024
  height: 179
  visible: 1

# Update product name button
-
  id: btn_custScr1_1
  type: button
  style_id: bookNowBtn_stl
  label_style: whiteLab_stl
  font: chivo40.bin
  x: 5
  y: 588
  width: 1014
  height: 164
  text_id: bookNow_txt
  visible: 1
```

This example file includes configurations for 4 different screen elements that is understandable from the *type* field (it has *label*, *image*, *panel* and *button* values).

All these screen elements are headed with commentary started with # sign. These commentaries give short brief about the screen element that is configured below and they are optional.

All these screen elements may be identified by their *id* field value (*lab_productName*, *img_logoBlack*, *pnl_bottom*, *btn_custScr1_1*). First 3 letters of the *id* value are additional information about screen element type (lab for labels, img for images, pnl for panels and btn for buttons).

All screen elements have their position in configurations that are presented by *x* and *y* fields (these fields are coordinates in pixels (starting from left top corner) of left bottom corner of the element).

Also, all these elements have *visible* field that value is 1 for all of them. This means that all these elements should be visible on the screens that they belong to.

Most of these screen elements (except Company logo black image with *id* equal to *img_logoBlack*) has *style_id* field in their configurations. Update product name button with *id* that is equal to *btn_custScr1_1* has also *label_style* field in its configurations. These fields configure the style of text, lines, different colors (of background, of text, of filling) etc. All styles are configured in style configuration file *style.yml* (see 4.5 for details).

Product name label (with *lab_productName* as the value of *id* field) and Update product name button (with *btn_custScr1_1* as the value of *id* field) have *text_id* in their configuration. This field is related to *id* field in *texts.yml* file. The text that should be viewed by label and button in this case may be found by *text_id* value (*meetingRoomDisplayTablet_txt* and *bookNow_txt*) in *texts.yml* file.

Also, these elements (Product name label (with *lab_productName* as the value of *id* field) and Update product name button (with *btn_custScr1_1* as the value of *id* field)) have *font* in their configurations to set the font files that are set fonts for texts in these elements.

Product name label (with *lab_productName* as the value of *id* field) has *width* in its configurations. This optional field is width of the label. This field is useful for example if style of the label content *text_align* option as *center* (see 4.5 for details) and it is necessary to place the text in the middle of the screen or panel.

Company logo black image (with *img_logoBlack* as the value of *id* field) includes image field with value *LOGO_B.bin* in its configurations. This means that image should use *LOGO_B.bin* file as source of picture. This file (*LOGO_B.bin*) should be uploaded to the Tablet disk drive. If this file is not on the Tablet disk drive or this file is corrupted then “No data” text will be shown instead of picture in image field.

Company logo black image (with *img_logoBlack* as the value of *id* field), Bottom panel (with *pnl_bottom* as the value of *id* field) and Update product name button (with *btn_custScr1_1* as the value of *id* field) have *width* and *height* in their configurations. These fields are dimensions of screen element and is mandatory for panels and buttons. Images do not require dimension fields *width* and *height* because they use their content dimensions.

4.3 Text Configuration File

To configure text constants, it is required to use texts.yml (this name is mandatory) configuration file. Text configuration file is based on YAML language. The structure of this file is:

```
---
Texts:
-
  <text 1 configurations>
-
  <text 2 configurations>
...
-
  <text N configurations>
```

<text 1 configurations>, <text 2 configurations> and <text N configurations> – sets of configurations for text constants 1, 2 and N respectively.

This document starts with “---”.

“Texts:” is on the next line.

The configurations of the text constants are placed after “Texts:” line.

The configurations of every text constant start with “-” on separated line.

Mandatory fields (there are no optional fields in text configurations) for all text constants configurations are:

- *id* – unique identifier of the text constant. If there are more than one text constant with the same value of *id* field then the last of such constants will be used by the Tablet and rest of such constants will be ignored. These values of *id* field may be used as value for *text_id* field in screen elements configuration file *elements.yml* (see section 4.1 for details). The values of this field have such format: *<string name>_txt* (for example *room_txt*), where:
 - *<string name>* – one of the possible string names that are (any other value will be ignored by Tablet): *meetingRoomDisplayTablet*, *euRegionStr*, *naRegionStr*, *asRegionStr*, *auRegionStr*, *krRegionStr*, *inRegionStr*, *ruRegionStr*, *dnRegionStr*, *fwVer*, *bootloadVer*, *apploadVer*, *iteVer*, *waitToJoin*, *joinedSuccess*, *waitRoomInfo*, *waitRoomStatus*, *tapToWakeUp*, *finishMeeting*, *requestPending*, *bookNow*, *noNextBooking*, *zero*, *doubleZero*, *nextBookingAt*, *am*, *pm*, *by*, *available*, *persons*, *book*, *b15min*, *b30min*, *b45min*, *b1h*, *meeting*,

extendMeeting, futureMeeting, extend, finish, bookNext, till, occupied, bookNextDots, cancel, extendDots, finishDots, yes, no, meetingSpace, bookingSpace, extended, successfully, failedToBe, room, booked or cust<number>, where number is value from 1 to 30 (for example cust15).

- *value* – text constant.

NOTE: Put text constant in "" to use special symbols in text (for example spaces or ':' symbol that will interpreted as YAML-language special symbol without putting in "").

NOTE: Do not use single (' and ') and double (" and ") quotes symbols in text (these symbols are using in YAML-language).

NOTE: Use '\n' to switch the text to the next line.

4.4 Example Text Configuration File

Text configuration file example is:

```
---
Texts:

# Custom Display Tablet
-
  id: meetingRoomDisplayTablet_txt
  value: Custom Display Tablet

# Firmware Version:
-
  id: fwVer_txt
  value: "Firmware Version:      "
```

This example file includes configurations for 2 text constants. They have unique *id* field values (*meetingRoomDisplayTablet_txt* and *fwVer_txt* for our case). These values of *id* field are usable as value of *text_id* field in elements.yml file.

All these screen elements are headed with commentary started with # sign. These commentaries give short brief about the text constant that is configured below and they are optional.

Custom Display Tablet (with *meetingRoomDisplayTablet_txt* as the value of *id* field) *value* field is *Custom Display Tablet* and Firmware Version (with *fwVer_txt* as the value of *id* field) *value* field is *"Firmware Version: "*. A string in the *value* field for *fwVer_txt* is placed in "" because this string has ':' inside and it is necessary to show to the Tablet that this symbol is part of the string but not the special symbol of the YAML.

4.5 Style Configuration File

It is possible to add, delete and configure styles. It is required to use `styles.yml` (this name is mandatory) configuration file for this purpose. This file should be uploaded to the Tablet through USB.

Style configuration file is based on YAML language. The structure of this file is:

```
---
Styles:
-
  <style 1 configurations>
-
  <style 2 configurations>
...
-
  <style N configurations>
```

`<style 1 configurations>`, `<style 2 configurations>` and `<style N configurations>` – sets of configurations for styles 1, 2 and N respectively.

This document starts with “---”.

“Styles:” is on the next line.

The configurations of the styles are placed after “Styles:” line.

The configurations of every style start with “-” on separated line.

Mandatory field for all styles configurations is:

- *id* – unique identifier of the style. It should be literal name that may consist of Latin letters, “_” and digits (first symbol has to be a letter) and its length should not exceed 20 symbols. It is desirable to add “_stl” at the end of the *id* of the style (do not forget to include these 4 symbols in *id* length limit).

Optional fields in style configurations are:

- *text_color* – a color of the text of label or label in button. Default value is *white*. Possible values of *text_color* field are (other value will be ignored by Tablet):
 - *white*
 - *light_grey*

- *dark_grey*
 - *black*
- *text_align* – the alignment of the text in label. Text alignment is used by label only if the width of the label is more than the width of label content text (see width property for Default value is *auto*. Possible values of *text_align* field are (other value will be ignored by Tablet):
 - *auto* – text align according to symbols that it contents (for example left alignment for Latin and right alignment for Arabic).
 - *left*
 - *center*
 - *right*
- *letter_space* – the space in pixels between the nearest letters in the same word. The range of this field is 0 – 10 (0 means that this option is disabled). Default value is 0 pixels.
- *bg_color* – a color of the background of the screen element. Possible values of *bg_color* field are the same as for *text_color*. Default value is *white*.
- *border_color* – a color of the borders of the screen element. Possible values of *border_color* field are the same as for *text_color*. Default value is *black*.
- *border_opa* – the opacity of the borders of the screen element. Default value is *opa_cover*. Possible values of *border_opa* field are (other value will be ignored by Tablet):
 - *opa_transp* – transparent border
 - *opa_20* – 20% opacity
 - *opa_40* – 40% opacity
 - *opa_60* – 60% opacity
 - *opa_80* – 80% opacity
 - *opa_100* – 100% opacity
 - *opa_cover* – full opacity (the same as *opa_100*)
- *border_side* – the side of the screen element where border should be placed. Default value is *none*. Possible values of *border_side* field are (other value will be ignored by Tablet):
 - *none*
 - *bottom*
 - *top*
 - *left*
 - *right*
 - *full*
- *border_width* – the width of the border in pixels. The range of this field is 0 – 50 (0 means that this option is disabled). Default value is 0.

- *shadow_color* – a color of the shadows of the screen element. Possible values of *shadow_color* field are the same as for *text_color*. Default value is *white*.
- *shadow_width* – the width of the shadow in pixels. The range of this field is 0 – 50 (0 means that this option is disabled). Default value is 0.
- *pad_hor* – horizontal padding in pixels. The range of this field is 0 – 20 (0 means that this option is disabled). Default value is 0.
- *pad_ver* – vertical padding in pixels. The range of this field is 0 – 20 (0 means that this option is disabled). Default value is 0.
- *pad_gap* – gap padding in pixels. The range of this field is 0 – 20 (0 means that this option is disabled). Default value is 0.
- *radius* – the radius of every corner in pixels. The range of this field is 0 – 20 (0 means that this option is disabled). Default value is 0.

NOTE: The number of configured styles in styles.yml file should not exceed 50.

4.6 Example Style Configuration File

Style configuration file example is:

```
---
Styles:

# Dark grey label style
-
  id: dGreySpaceLab_stl
  text_color: dark_grey
  text_align: center
  letter_space: 1

# Generic button style
-
  id: genericBtn_stl
  bg_color: white
  radius: 5
  pad_hor: 0
  pad_ver: 0
  pad_gap: 0
  border_color: dark_grey
  border_width: 10
  border_opa: opa_cover
  border_side: full
  shadow_color: dark_grey
  shadow_width: 5
```

This example file includes configurations for 2 styles (with names *dGreySpaceLab_stl* and *genericBtn_stl* that are set in *id*).

All these styles are headed with commentary started with # sign (string after # is interpreted as commentary in YAML). These commentaries give short brief about the style that is configured below and they are optional.

Dark grey label style (with *dGreySpaceLab_stl* as the value of *id* field) is useful for label or label in button because it sets the configurations of text. These configurations are *text_color* that sets the color of text (*dark_grey* that means dark grey color in this case), *text_align* that sets the alignment of text (*center* that means centered text in this case) and *letter_space* that sets the space between letters in one word (1 pixel in this case).

Generic button style (with *genericBtn_stl* as the value of *id* field) may be useful for button or for panel because it sets the configurations of background, padding, borders and shadows:

- *bg_color* sets that background should be white (value *white*) here
- *radius* sets that the radius of every corner should be equal to 5 pixel (value *5*) here
- *pad_hor*, *pad_ver* and *pad_hor* set that the screen element that use this style should not have paddings (values *0*). These fields may be removed from this style configurations because default values for padding properties are *0* and it is not mandatory to set the properties if it is necessary to use default values
- *border_color* sets that border should be dark grey (value *dark_grey*), *border_width* sets that border width should be equal to 10 pixel (value *10*), *border_opa* sets that border opacity should be full (value *opa_cover*), *border_side* sets that border should be along all sides of the screen element that use this style (value *full*)
- *shadow_color* sets that shadow should be dark grey (value *dark_grey*), *shadow_width* sets that border width should be equal to 5 pixel (value *5*).

4.7 Uplink Configuration File

It is possible to add and change configurations for custom uplinks that are correspond to application in *uplinks.yml* configuration file. This file should be uploaded to the Tablet through USB.

Uplink configuration file is based on YAML language. The structure of this file is:

```
---
Uplinks:
-
```

```
<uplink 1 configurations>
-
<uplink 2 configurations>
...
-
<uplink N configurations>
```

<uplink 1 configurations>, <uplink 2 configurations> and <uplink N configurations> – sets of configurations for uplinks 1, 2 and N respectively.

This document starts with “---”.

“Uplinks:” is on the next line.

The configurations of uplinks are placed after “Uplinks:” line.

The configurations of every uplink start with “-” on separated line.

It is allowed to configurate different uplinks with the same *header_id* or any other field.

Mandatory fields in uplink configurations are:

- *header_id* – 1 byte message ID. It is allowed to set the value of this field in Decimal (should be set as usual number) and in Heximal (should be set with 0x or 0X prefix). It is not possible to set *header_id* equal to values that are message IDs for predefined uplinks (from 0x00 to 0x13, from 0x24 to 0x27, from 0x40 to 0x42, 0x70 and 0x72): tablet ignore uplink with such predefined message ID.
- *port* – LoRa port number. It is not possible to set *port* 20 (if *port* is set to 20 then uplink with such configuration will be ignored by tablet).
- *size* – payload size (with message ID). If *size* is set to 0, tablet will increase it to 1 (message ID size) internally.
- *source* – “something” that cause sending the uplink. Possible values (any other value will be ignored) for this field are:
 - *timer* – uplink will be sent periodically. If *source* is set as *timer*, *period* is also mandatory field.
 - *element* – uplink will be sent after pressing certain screen button. If *source* is set as *element*, *element_id* is also mandatory field.
 - *event* – uplink will be sent after some event that can happen during tablet runtime. If *source* is set as *element*, *event_id* is also mandatory field.

Optional fields in uplink configurations are:

- *period* – number of core ticks between two periodical uplinks. This field will be ignored if *source* is not set as *timer*.

- *element_id* – identifier of the button that cause sending the uplink. All button identifiers could be found in *elements.yml* file (see section 4.1 for details about screen elements configuration file) that configurate screen elements (any other value will be ignored). This field will be ignored if *source* is not set as *element*.
- *event_id* – identifier of the event that cause sending the uplink. Possible values (any other value will be ignored) for this field are:
 - *join_evt* – uplink will send right after tablet join to LoRa network.
- *retry* – this field is show if uplink should be resent in case of absence of response downlink to it (if *retry* value set equal to 1, uplink should be resent if there were no response to it; if *retry* value set equal to 0 or this field is absent in uplink configurations, uplink do not require response). If *retry* value is set equal to 1, repeat uplink will be sent every 30 sec in case of absence of response downlink (dummy uplinks will also be emitted by tablet every 10 sec in this case). This field will be ignored if *response_dl* field value is set equal to 0 or if there are no *response_dl* in uplink configurations.
- *response_dl* – message ID of response downlink. The downlink with such message ID (any downlink with such message ID) stops uplink retry cycle and marks uplink request as responded. This field will be ignored if *retry* field value is set equal to 0 or if there are no *retry* in uplink configurations. It is allowed to set the value of this field in Decimal (should be set as usual number) and in Heximal (should be set with 0x or 0X prefix).
- *payload* – 1 byte payload value. It is allowed to set the value of this field only in Heximal (should be set with 0x or 0X prefix). This field is proceeding only if *size* is value is not less than 2. It is possible to use different payloads for uplinks with the same header.

NOTE: The number of configured uplinks in *uplinks.yml* file should not exceed 254.

4.8 Example Uplink Configuration File

Uplink configuration file example is:

```

---
Uplinks:

# Get Main Screen request (0x58)
-
  header_id: 0x58
  port: 10
  size: 1
  source: event
  event_id: join_evt
  retry: 1
  response_dl: 0x58

```

```
# Get Temperature request (0x53)
```

```
-
```

```
  header_id: 83  
  port: 10  
  size: 1  
  source: timer  
  period: 2  
  retry: 1  
  response_dl: 0x53
```

```
# Switch Screen request (0x54)
```

```
-
```

```
  header_id: 0x54  
  port: 10  
  size: 2  
  source: element  
  element_id: btn_custScr2_3  
  payload: 0x05  
  retry: 1  
  response_dl: 0x54
```

This example file includes configurations for 3 uplinks (with message IDs *0x58*, *83* (that is *0x53*) and *0x54* that are set in *header_id*).

All these uplinks are headed with commentary started with # sign (string after # is interpreted as commentary in YAML). These commentaries give short brief about the uplink that is configured below and they are optional.

All these uplinks are configured to be sent on port *10*.

Get Main Screen request with *header_id 0x58* and Get Temperature request with *header_id 83* have *size 1* that means that these requests are consist of 1 byte message ID only. Switch Screen request with *header_id 0x54* has *size* equal to 2 that means that this uplink consists of 1 byte message ID and 1 byte payload. This payload is defined by *payload (0x05* in this case).

All these uplinks configured to be resendable (*retry* field value is *1*) till getting the response (response downlink message ID is defined with *response_dl*).

Get Main Screen request has *event* as its *source*. That means that Get Main Screen request uplink is emitted by tablet after special event in tablet. This event is defined in *event_id* field (*join_evt* in this case that mean that this request is emitted by tablet after successful joining to LoRa network).

Get Temperature request has *timer* as its *source*. That means that Get Temperature request uplink is emitted periodical by tablet every 2 core ticks that is set in *period*.

Switch Screen request has *element* as *source*. That means that Switch Screen request uplink is emitted by tablet after pressing the screen button that is set in *element_id* field (*btn_custScr2_3* in this case).

4.9 Downlink Configuration File

It is possible to add and change configurations for custom downlinks that are correspond to application in *dlinks.yml* configuration file. This file should be uploaded to the Tablet through USB.

Downlink configuration file is based on YAML language. The structure of this file is:

```
---
Downlinks:
-
  <downlink 1 configurations>
-
  <downlink 2 configurations>
...
-
  <downlink N configurations>
```

<downlink 1 configurations>, <downlink 2 configurations> and <downlink N configurations> – sets of configurations for downlinks 1, 2 and N respectively.

This document starts with “---”.

“Downlinks:” is on the next line.

The configurations of downlinks are placed after “Downlinks:” line.

The configurations of every downlink start with “-” on separated line.

It is allowed to configurate different downlinks with the same *header_id* or any other field.

Mandatory fields in downlink configurations are:

- *header_id* – 1 byte message ID. It is allowed to set the value of this field in Decimal (should be set as usual number) and in Heximal (should be set with 0x or 0X prefix).
- *port* – LoRa port number.

Optional fields in downlinks configurations are:

- *handler_id* – identifier of the handler that processes downlink payload. Possible values are:
 - *updateElement_hdl* – handler for Update element downlink.
 - *eodSleep_hdl* – handler for End-of-day sleep downlink.
- *screen_id* – identifier of the screen that tablet should switch to after receiving configured downlink. Possible values are:
 - *init_scr* – initialization screen that consist of 6 labels, 1 panel and 1 image. This screen is designed to view base Custom Display Tablet version information.
 - *custom1_scr, custom2_scr, custom3_scr, custom4_scr* – screens that is designed to have possibility to configure the custom screen. They consist of 10 labels, 6 buttons, 5 panels and 5 images on each screen.

NOTE: All custom screens do not have cross-screen connections between labels, panels and images.
- *ack_screen_id* – identifier of the screen that tablet should switch to after receiving acked configured downlink. This field is ignored if *screen_id* is present in downlink configurations. This field is used as *screen_id* configuration if *nack_screen_id* and *screen_id* fields are absent in downlink configurations. Possible values are the same as for *screen_id*.
- *nack_screen_id* – identifier of the screen that tablet should switch to after receiving nacked configured downlink. This field is ignored if *screen_id* is present in downlink configurations. This field is used as *screen_id* configuration if *ack_screen_id* and *screen_id* fields are absent in downlink configurations. Possible values are the same as for *screen_id*.

NOTE: The number of configured downlinks in dlinks.yml file should not exceed 254.

4.10 Example Downlink Configuration File

Downlink configuration file example is:

```

---
Downlinks:

# Get Main Screen response (0x58)
-
  header_id: 0x58
  port: 103
  screen_id: custom1_scr

# Get Temperature response (0x53)
-
  header_id: 83

```

```
port: 102
handler_id: updateElement_hdl
ack_screen_id: custom2_scr
nack_screen_id: custom3_scr

# Deep Sleep Management response (0x59)
-
header_id: 89
port: 104
handler_id: eodSleep_hdl
```

This example file includes configurations for 3 downlinks (with message IDs *0x58*, *83* (that is *0x53*) and *89* (that is *0x59*) that are set in *header_id*).

All these downlinks are headed with commentary started with # sign. These commentaries give short brief about the downlink that is configured below and they are optional.

Get Main Screen response downlink (with *header_id 0x58*) is the first downlink in this configuration file. It is configured to port *103*. This mean that optional configurations (*screen_id* in this case) is used only if tablet received downlink on port that is set in *port* field and with message ID that is specified in field *header_id* (*port 103* and *header_id 0x58* in this case). There is *screen_id* with *custom1_scr* as its value in configurations of this downlink. This mean that tablet switches its view to first custom screen after receiving of this downlink.

Get Temperature response is configured as downlink with message ID *83* (or *0x53* in Heximal) that is set in *header_id* field and is waiting on port *102* that is set in *port* field. The payload of this downlink is processes as Update Element downlink that is set in *handler_id* by value *updateElement_hdl*. The view of the tablet switches to the second custom screen after receiving this acked downlink (that is set in field *ack_screen_id* by value *custom2_scr*) and to the third custom screen after receiving this nacked downlink (that is set in field *nack_screen_id* by value *custom3_scr*).

Deep Sleep Management response is configured as downlink with message ID *89* (or *0x59* in Heximal) that is set in *header_id* field and is waiting on port *104* that is set in *port* field. The payload of this downlink is processes as End-of-day sleep downlink that is set in *handler_id* by value *eodSleep_hdl*. The view of the tablet does not switch to any screen because there no *screen_id*, *ack_screen_id* or *nack_screen_id* in configurations for this downlink.

4.11 Image Files

It is possible to upload through USB files with images to view them on Tablet screen.

The Tablet support images that are converted to binary format with file extension “.bin” (do not use capital letters in extension). To convert picture from popular format (.jpg, .bmp, .png) it is possible to use online converter (see section 4.12 for details).

The picture is used as source for screen element with *type* field that is equal to *image* (see section 4.1 for details). To set picture as source for screen element add the name of file with picture to field *image*.

NOTE: It is important to use right register in the names of files in the Tablet.

NOTE: The length of the name of the file has to be less than or equal to 8 symbols and the length of the file extension has to be less than or equal to 3 symbols.

WARNING! Total size of all used as sources files (images and fonts) should be less than 220 kB. Some part or all your image files will not be viewed on the Tablet screen and some part of fonts will not be applied to the screen elements if you exceed this size.

4.12 Online Image Converter

Online image converter is a free tool that is accessible through the link: <https://lvgl.io/tools/imageconverter>

It is possible to create binary files from images (.png, .jpg and .bmp).

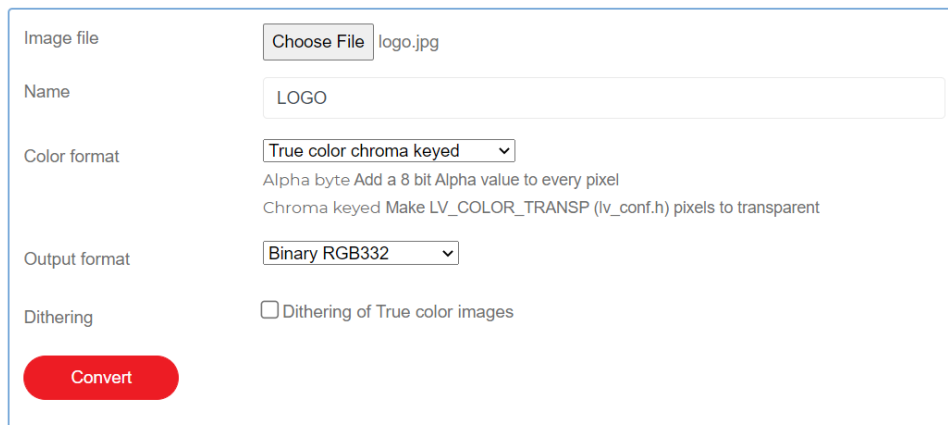


Figure 4-1: Online Image Converter

To convert a picture to binary format it is necessary to:

1. Choose an Image (png, jpg, or bmp) (see Figure 4-1).
2. Give a Name to the output file. E.g. "LOGO"
3. Specify the Color format. It is recommended to use “True color chroma keyed” format. Some format may convert your picture wrong.

4. Set the output format to Binary RGB332, Binary RGB565 or Binary RGB888. Do not use “C array”. It is recommended to use “RGB332” format to decrease image file size.
5. Click the Convert button and to download the converted image.

Short explanations about image converter and about color formats can be find on the same page.

4.13 Font Files

It is possible to upload through USB files with fonts to set the font that is presented by font file to screen element with screen element configuration file (see section 4.1 for details).

The Tablet support fonts that are converted to binary format with file extension “.bin” (do not use capital letters in extension). To convert font from popular format (.ttf, .otf and .woff) it is necessary to use offline converter (see section 4.14 for details).

The font is used by screen element with *type* field that is equal to *label* or *button* (see section 4.1 for details). To set font for screen element add the name of file with font to field *font*.

NOTE: It is important to use right register in the names of files in the Tablet.

NOTE: The length of the name of the file has to be less than or equal to 8 symbols and the length of the file extension has to be less than or equal to 3 symbols.

WARNING! Total size of all used as sources files (images and fonts) should be less than 220 kB. Some part or all your image files will not be viewed on the Tablet screen and some part of fonts will not be applied to the screen elements if you exceed this size.

4.14 Offline Font Converter

Offline font converter is a free tool that is require node.js 10+ (node.js may be downloaded from <https://nodejs.org/en/download/>).

To install globally Font Converter (“lv_font_conv”) use your command prompt and these commands:

```
# install release from npm registry
npm i lv_font_conv -g
# install from github's repo, master branch
npm i lvgl/lv_font_conv -g
```

It is possible to use lv_font_conv command in command prompt after that.

The command with -h key shows help message and the meanings of the keys that is possible to use with lv_font_conv:

```
> lv_font_conv -h
usage: lv_font_conv.js [-h] [-v] --size PIXELS [-o <path>] --bpp {1,2,3,4,8} [--lcd | --lcd-v] [--use-color-info]
                        --format {dump,bin,lvgl} --font <path> [-r RANGE] [--symbols SYMBOLS] [--autohint-off]
                        [--autohint-strong] [--force-fast-kern-format] [--no-compress] [--no-prefilter] [--no-kerning]
                        [--lv-include <path>] [--full-info]

optional arguments:
  -h, --help            show this help message and exit
  -v, --version          show program's version number and exit
  --size PIXELS         Output font size, pixels.
  -o <path>, --output <path>
                        Output path.
  --bpp {1,2,3,4,8}    Bits per pixel, for antialiasing.
  --lcd                 Enable subpixel rendering (horizontal pixel layout).
  --lcd-v              Enable subpixel rendering (vertical pixel layout).
  --use-color-info     Try to use glyph color info from font to create grayscale icons. Since gray tones are emulated
                        via transparency, result will be good on contrast background only.
  --format {dump,bin,lvgl}
                        Output format.
  --font <path>        Source font path. Can be used multiple times to merge glyphs from different fonts.
  -r RANGE, --range RANGE
                        Range of glyphs to copy. Can be used multiple times, belongs to previously declared "--font".
                        Examples:
                        -r 0x1F450
                        -r 0x20-0x7F
                        -r 32-127
                        -r 32-127,0x1F450
                        -r '0x1F450=>0xF005'
                        -r '0x1F450-0x1F470=>0xF005'
  --symbols SYMBOLS    List of characters to copy, belongs to previously declared "--font".
Examples:
  --symbols ,.0123456789
  --symbols abcdefghijklmnopqrstuvwxyz
  --autohint-off       Disable autohinting for previously declared "--font"
```

```
--autohint-strong  Use more strong autohinting for previously declared "--font" (will break kerning)
--force-fast-kern-format
                    Always use kern classes instead of pairs (might be larger but faster).
--no-compress      Disable built-in RLE compression.
--no-prefilter     Disable bitmap lines filter (XOR), used to improve compression ratio.
--no-kerning       Drop kerning info to reduce size (not recommended).
--lv-include <path> Set alternate "lvgl.h" path (for --format lvgl).
--full-info        Don't shorten "font_info.json" (include pixels data).
```

For example, if it is necessary to create binary bold_font_30.bin font file with Latin symbols from Chivo-Bold.ttf file and with Arabic symbols from DejaVuSans-Bold.ttf file with font height equal to 30 pixels than you may use this command:

```
> lv_font_conv --no-compress --no-prefilter --bpp 1 --size 30 --font Chivo-Bold.ttf -r 0x20-0x7F --font DejaVuSans-Bold.ttf -r 0x600-0x6FF -r 0xFB50-0xFDFF -r 0xFE70-0xFEFF --format bin -o bold_font_30.bin --force-fast-kern-format
```

More examples and details about Offline Font Converter see at:

https://github.com/lvgl/lv_font_conv.

5 Operation, Alarms, and Management

5.1 Configuration

The Tablet supports a range of Over-the-Air (OTA) configuration options. Specific technical details are available in the Technical Reference Manual [1]. All configuration commands need to be sent OTA during a Tablet's downlink windows.

5.2 Default Configuration

The default configuration on the Tablet is:

- Report the battery voltage every 1 (one) hour.

6 Firmware Upgrade Feature

6.1 Overview

The Custom Display Tablet has a bootloader in the flash allow for firmware updates in the field. See the steps outlined below to update the firmware of your device.

6.2 Setup/Requirements

In order to complete a Firmware Upgrade on the Tablet you require the following setup:

- Windows 10 Laptop/PC Required. Drivers for DS may be required to install on the Laptop.
- Teraterm
- Micro USB Cable (see Figure 6-1)
- Custom Display Tablet (see Figure 6-2)

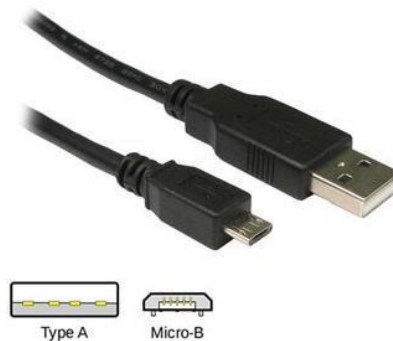


Figure 6-1: USB A to Micro-B Cable

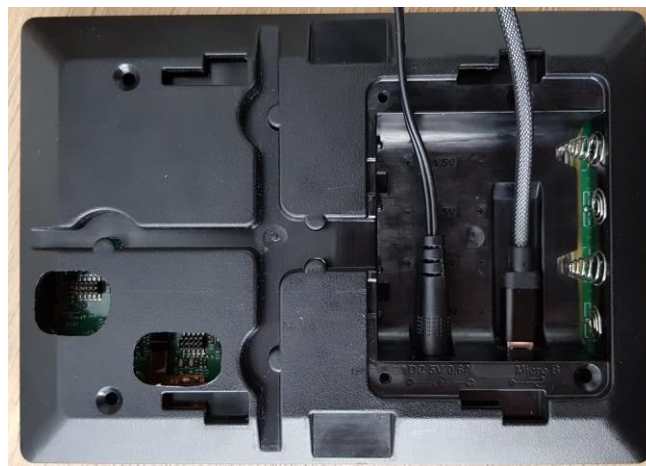


Figure 6-2: Tablet with power plug and USB cable connected

6.3 Steps to Complete Firmware Upgrade

1. Power Up the Tablet.
2. Connect the USB cable to Tablet then to Laptop/PC.
3. Then launch the Teraterm application, then select the serial and select the correct COM Port and select “OK”. (see Figure 6-3)

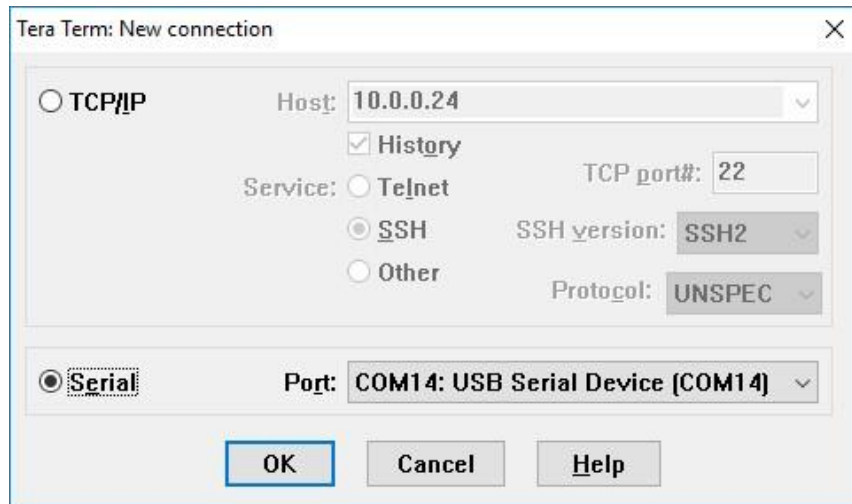


Figure 6-3: Com Power Selection

4. Then click “Setup” menu and then click “Serial Port” then set the “Speed” to 115200. Then click “OK”.

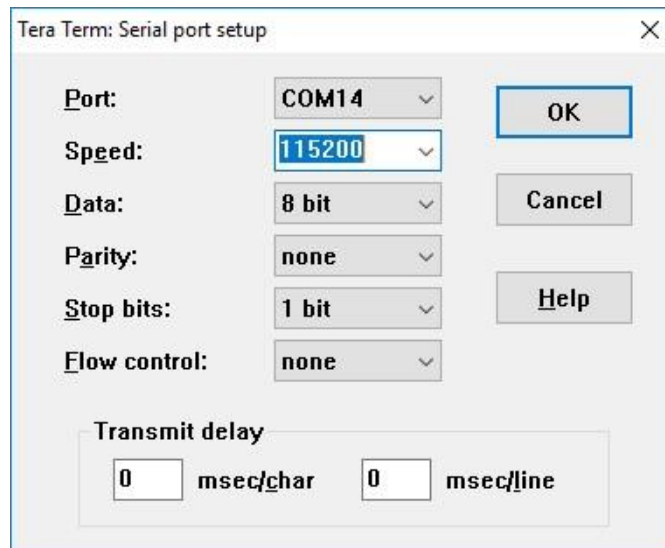


Figure 6-4: Serial Port and Speed Selection

5. Then “reset” the Digital Signage by pressing the reset button using pin. Then Teraterm serial console would display “Ready for the firmware file” message along with continuous “C”.

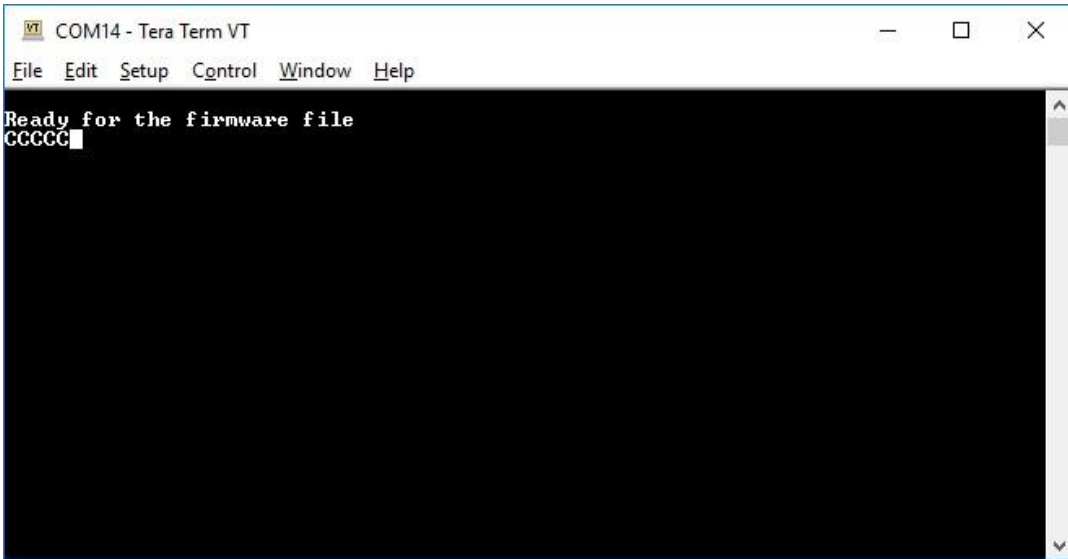


Figure 6-5: Ready for Firmware Note

- Then go to "File->Transfer->XModem->Send". Then browse and select the FW file (.hex file). Then select "Open". Then Firmware Upgrade would start. The progress bar would be displayed.

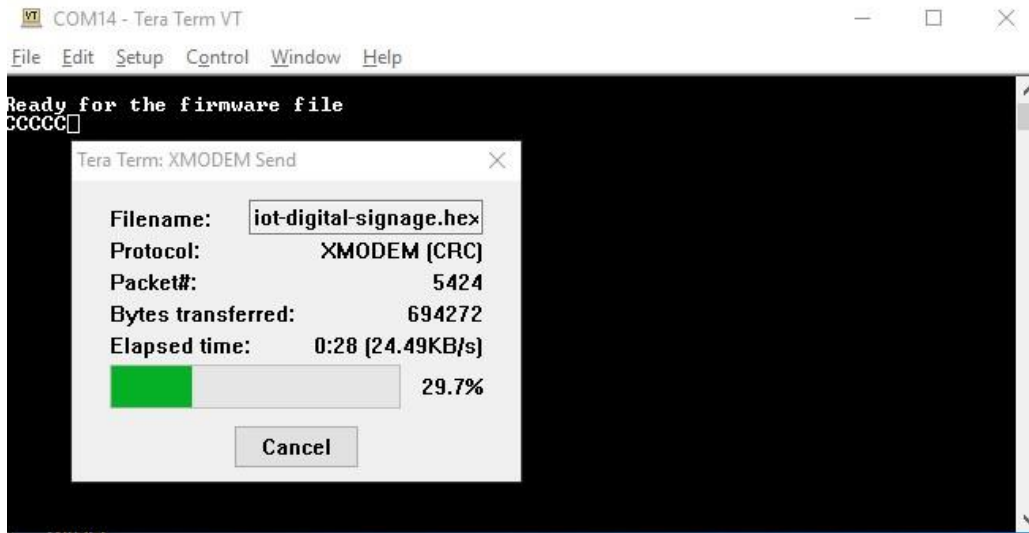


Figure 6-6: Progress Bar

NOTE: Starting the FW Upgrade should happen within a minute. Otherwise, the USB operation would timeout and go to the actual image. These messages would be displayed when it is timed out.

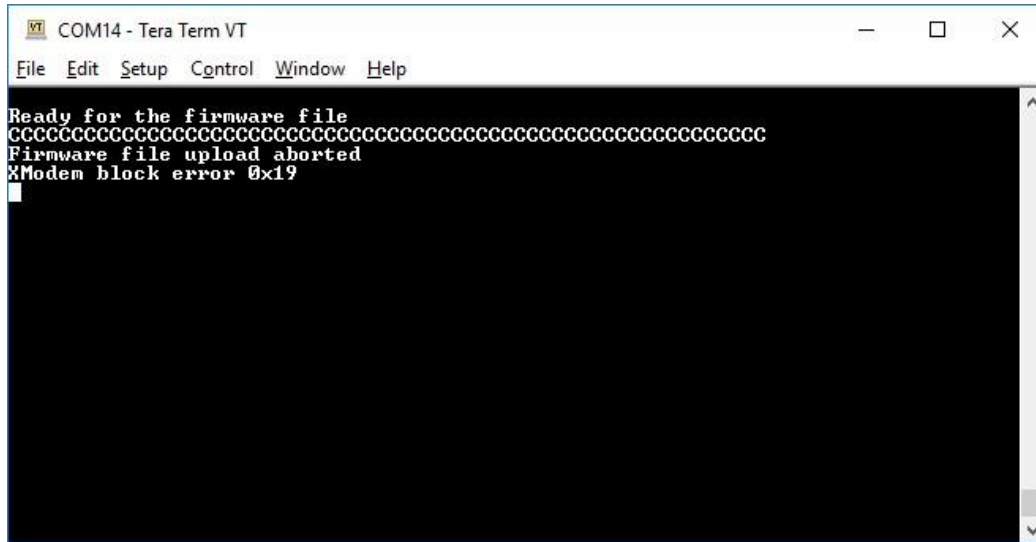


Figure 6-7: Firmware file upload aborted

7. After the successful Firmware Upgrade, the following message would be displayed in the serial console.

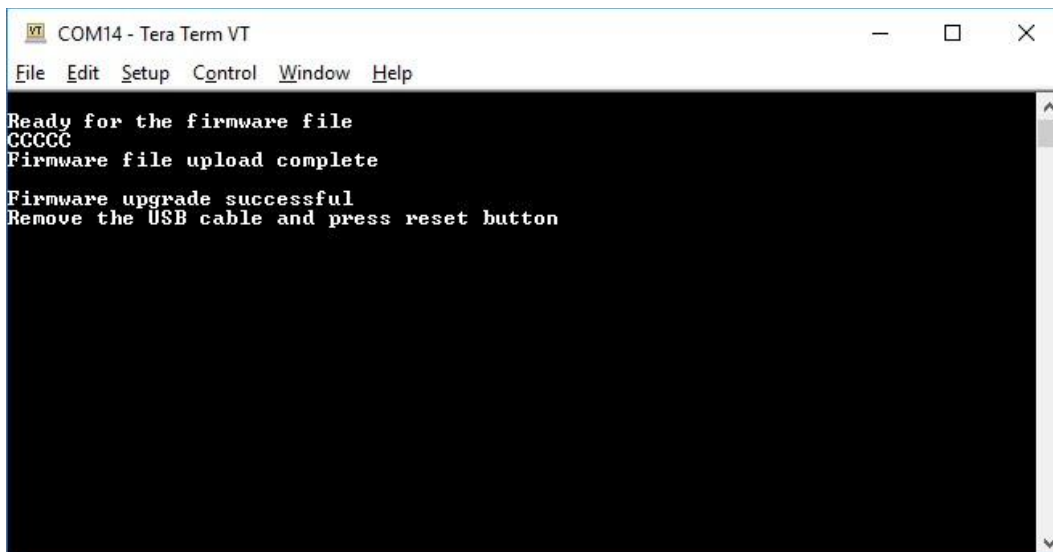


Figure 6-8: Firmware upgrade successful

8. Then as per the messages displayed in serial console, remove the USB cable and press “reset” button on the DS and let the device power up.

6.4 Touchscreen Firmware Upgrade

It is possible to upgrade touchscreen controller firmware. To upgrade touchscreen firmware, it is necessary:

1. Connect USB cable to Tablet and then to Laptop/PC.

2. Press Reset button at the bottom side of Custom Display Tablet (see Figure 1-2).
3. Wait till Custom Display Tablet will be seen as USB flash drive in OS of Laptop/PC.
4. Upload touchscreen controller firmware file with ts_fw.ekt name (this name is mandatory) to USB flash drive associated with Custom Display Tablet.
5. Disconnect USB cable from the Tablet.
6. Power up the Tablet.
7. Press Reset button at the bottom side of Custom Display Tablet (see Figure 1-2).
8. Wait till the Table will load and check that version of touchscreen firmware is corresponded to version presented with uploaded firmware file on Initial screen (if string with touchscreen firmware is present).

7 Compliance Statements

Federal Communications Commission:

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. This device must accept any interference received, including interference that may cause undesired operation.

To comply with FCC exposure limits for general population / uncontrolled exposure, this device should be installed at a distance of 20 cm from all persons and must not be co-located or operating in conjunction with any other transmitter.

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment. This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Innovation, Science and Economic Development Canada:

This device contains licence-exempt transmitter(s)/receiver(s) that comply with Innovation, Science and Economic Development Canada's licence-exempt RSS(s). Operation is subject to the following two conditions:

- (1) This device may not cause interference.

- (2) This device must accept any interference, including interference that may cause undesired operation of the device.

This device should be installed and operated with minimum distance 0.2 m from human body.

L'émetteur/récepteur exempt de licence contenu dans le présent appareil est conforme aux CNR d'Innovation, Sciences et Développement économique Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes:

- a. L'appareil ne doit pas produire de brouillage.
- b. L'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Cet appareil doit être installé et utilisé à une distance minimale de 0.2 m du corps humain.

California Proposition 65:

WARNING: This product can expose you to chemicals including lead, nickel & carbon black, which are known to the State of California to cause cancer, birth defects or other reproductive harm. For more information, go to www.p65warnings.ca.gov.

8 Bibliography

- [1] "Custom Display Tablet Technical Reference Manual," 2021.
- [2] LoRa Alliance, "LoRaWAN 1.1 Regional Parameters," ver. 1.1, rev. B, Jan 2018.
- [3] "Tektelic Network Server User Guide," 2019.
- [4] LoRa Alliance, "LoRaWAN 1.02 Regional Parameters," Revision B, Feb 2017.
- [5] TEKTELIC Communications Inc., "LoRa IoT Smart Room Sensor Technical Reference Manual (TRM)," ver. 1.0.
- [6] LoRa Alliance, "LoRaWAN Specification," ver. 1.0.2, rev. B, Jul 2016.