

Created at 2021. 02. 26. 17:40
(for App. ver. 1.3)

Table of Contents

1. About.....	2
1.1. Default text.....	2
1.2. Default font.....	3
2. The editor Application.....	4
2.1. Create / extend a translation.....	6
2.1.1. Open existing data.....	6
2.1.2. Merge a new Fw's data into the existing.....	7
2.1.3. Select and edit an entry.....	7
2.1.4. Export the translation to a binary file.....	9
2.1.5. Saving the data.....	10
3. Using the translation in the Fw.....	11
3.1. Restrictions and notes.....	12

1.About

The texts of the MGen-3 HandController's (HC) Firmware are constantly present for each version release but the system is prepared for handling external text definitions along with an external font type. The User is able to change the text of the HC with creating a *string definition binary* file, putting it onto the SD card and enabling the usage of it by a checkbox in the Firmware.

The next sections will describe the steps of creating a custom text and/or font version for a given Firmware release.

As the editor Application enables entering of any texts, the author of MGen-3 takes no responsibility for their content!

1.1. Default text

One *text (definition) entry* is usually a short expression or word that the Fw. displays for an individual element (a parameter / value) or can be a longer sentence that is displayed as an info or message box text. These entries can be redefined externally using the *MGen-3 Text Definition Editor* Application.

By later Fw. releases the user can use an older *string definition binary* file (made for an older Firmware) as the text entries are defined as backward-compatible. They change for a new text only if the meaning of them changes much and it's very probable that the user should to translate / modify it for himself. For such entries the default English text are shown.

The default language of the Firmware is English. In the case of a missing external *text entry* the default English text of this entry will be displayed. Note that if the english characters are missing from the chosen font, only rectangles will be shown for each default text entries.

1.2. Default font

The font to be used would also be defined in the external *string definition binary*. The User must give the font type when exporting the data into this binary.

A preview window is available on the editor Application to see how the text will look like on the HC's display. The User must ensure that a character is available and properly visible on this view before using a font / character for his text definition file.

The system's default font embedded in each Fw. release is the following:

- Font: **Arial**
- Font style: **Regular**
- Size: **10**
- Script: **Western**

Exporting with this font setting should lead to the same look-out as by default. The Firmware is developed using this default font, the default text, item organization and (max.) size etc. are all based on this.

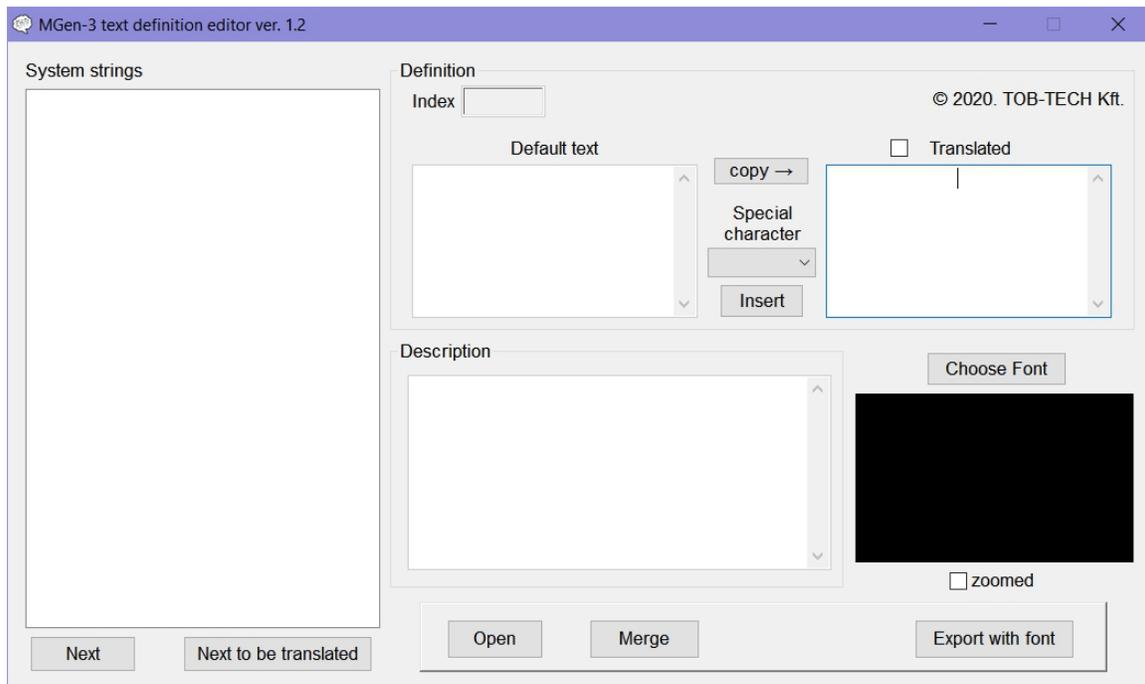
2.The editor Application

It is called *MGen-3 Text Definition Editor*. The App's version that this document is valid for can be found on the first page.

As this feature must be mostly used for translation into other languages I call the items and text a “translation”.

The current release is a 64-bit executable for Windows 7/8/10.

After starting the App, the main window appears with the following sections on it:



- ✧ On the left there is the tree view of the loaded data with all the available definitions listed (“System strings”). As there is no data loaded, it’s empty after startup.
- ✧ On the middle top “Default text” box shows you what text is used by the Firmware for that text entry, if you selected one of the System strings. This is a read-only field.
- ✧ On the right top there is the “Translated” checkbox and text box that is used to define a new text entry (the “translation”) instead of the default one. You must enter any text there with UNICODE characters. The checkbox must be checked to export the given string as the text entry.

By the “Copy ->” button you can copy the default text to the translated one.

The “Special character” and “Insert” button items are used to enter the special characters defined for the Firmware. (More on this later.)

- ✧ On the middle bottom the “Description” field will have some info about the text what it should mean, where or when it is displayed etc. This helps you when making translation for other language.
- ✧ On the right bottom you can see a black preview window, in which the selected text entry will be shown as it would be shown on the display of the HC. You must choose a font first to have this preview available.
- ✧ On the bottom you can find the control buttons: loading, merging data and exporting it along with the selected font.

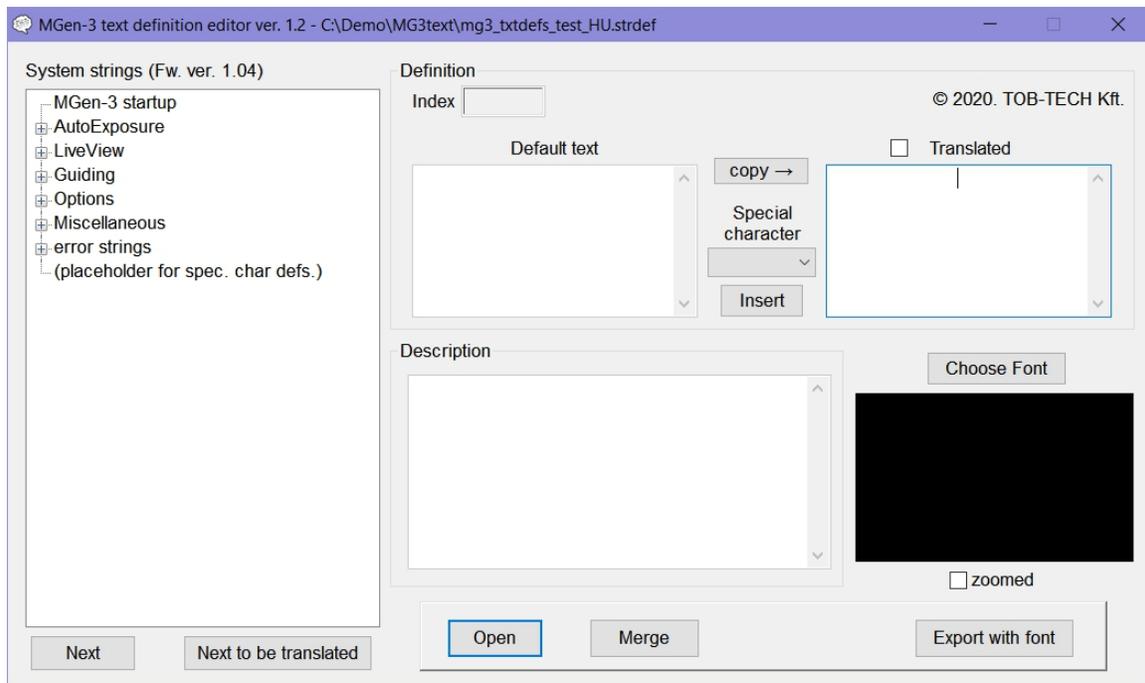
2.1. Create / extend a translation

If you want to start a new translation instead of extending an existing one, do the following:

- You may need to download the text package for a given Firmware version; this package contains (a probably newer version of) this document and the data file called e.g. `mg3_txtdefs_v0104.strdef` for Fw. ver. 1.04.
- Copy the data file with whatever name you want to have as your base translation data file. For example you can copy it to `mg3_txtdefs_test_HU.strdef` for your own Hungarian translation. The actual version number of it will be stored in the file itself when you later extend it with a new Firmware's data.
- Now you can continue just like you were about to extend an existing translation with a new Firmware's data you haven't translated yet.

2.1.1. Open existing data

Press the “Open” button and select your base translation data file you want to edit or extend.



The “System strings” tree now contains the text entries loaded from your data file. These are the same as were after the latest merging. (See later.) The Fw. version number is shown in the brackets that this text data can be used for.

If a tree item is not an entry but some organizing branch, you can’t see a default text for it.

By clicking “Next” the next text entry will be selected automatically.

By clicking “Next to be translated” the next entry will be selected that has the “Translated” option yet unchecked. Using this you can be sure that all entries were processed.

2.1.2. Merge a new Fw.’s data into the existing

You can skip this chapter if you started a new translation, probably with the newest Firmware’s text definition file.

To merge a newer Fw.’s texts with your existing data, click “Merge” after opening your data. You must pick the new Fw.’s `.strdef` file with the dialog box. If the file is valid, merging is done immediately. Your “System strings” tree may have changed (as the new Fw. changes) but the translations are kept that you have already done.

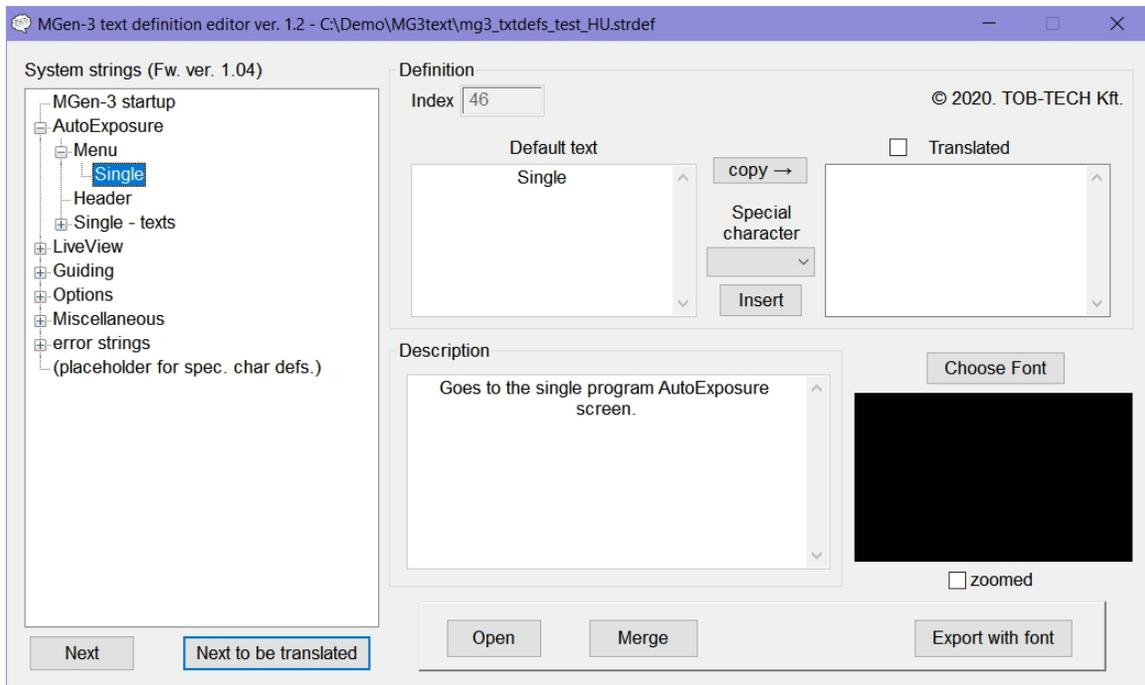
After a merging it’s probably that you need to translate only some new entries or re-translate some former entries that have some new meaning in the new Fw. The “Next to be translated” button will guide you through these.

2.1.3. Select and edit an entry

When you select a text entry from the tree list, fields are filled with its data:

- Default text: the one used by the Firmware;
- Translated: checkbox and text with your existing definition (or empty);
- Description: describes the text when it is displayed and why etc.

For example if you select the first item, which is the “Single” screen’s name as displayed in the AutoExposure’s menu:



You have the default text as “Single”. Now you can enter the translated form of it into the edit field under “Translated”, click on the field and type.



“Translated” checkbox automatically changes to checked whenever the text is altered in this field. If you want to translate a text entry later and don’t want to forget about it, neither lose what you’ve already typed in, simply click the checkbox (uncheck) to remove the text’s “translated” state.

You can continue to the next not-yet-translated text entry by clicking “Next to be translated”. Once these are all done, a message informs you about that.

About multi line entries:

MGen-3 Fw. rarely use multi line texts, by default all text should be treated and will be displayed as single line, even if a new line was inserted in the App’s edit field. In this single line case the new line character entered is ignored and is not replaced with space.

Multi line texts are for example message body texts, where the text is automatically split to lines at white space characters between the words. From ver. 1.10 the Fw.

interprets newline (NL) characters for this multi line entries and NL characters insert a new line regardless to the current line's width. The automatic line splitting works at white space characters between the words as by default.

About inversion:

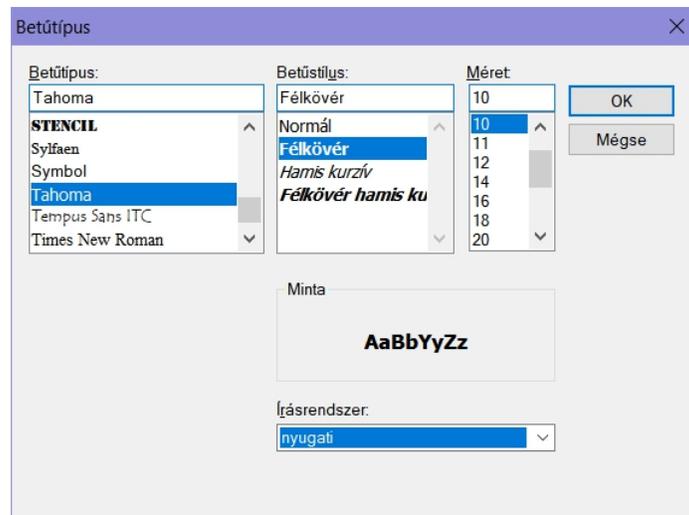
The default character set includes two control characters (specials) that controls the inverted mode: "normal" and "inverted". Once the "inverted" character appears, the following characters will be displayed inverted, the background will be drawn with the text color and the character area is left transparent. This state is held until a "normal" character code appears.

For multi line display, an inverted text area is treated as a single word and so will not be split automatically at all, not even for the presence of a newline character.

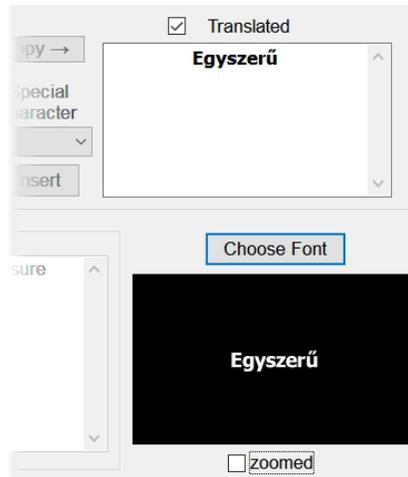
2.1.4. Export the translation to a binary file

As the first step you need to choose a font that the HC should use for displaying your (and probably the default) texts. Click on "Choose Font" button, a font selector dialog box opens. Select your font, pick the options carefully to have all characters available. The size used is always 10 for rendering, this option doesn't make sense here.

For example, choosing "Tahoma" font with bold style and western script:



The text with the selected font can be viewed both in the "Translated" edit field and the HC's preview window. Note that the preview window may show only a rectangle (missing character) for some characters that are not available for the selected font, while your PC may be able to replace that character from another type and show it almost correctly in the "Translated" edit field. You'll always get what the preview window shows, except for the text inversion and multiline features.



Checking “zoomed” will display the text at a 3x scale.

Though you can enter a multi-line text, a text definition can not have line breaks and so the preview won’t show them too. Line breaks are automatic by the Firmware where allowed. If you need long text, simply enter it into the edit field using spaces. It will be auto-broken into lines but only for the display on your PC. The preview won’t do any line breaks.

If you have already selected a font you don’t need to do it again right before the export. Also, selecting a font in advance helps you to check your translation entries if they would be displayed well or not.

Now click the “Export with font” button and give some name to your *string definition binary* file. The extension will be `.m3tx`. This file is the one that the MGen-3 HC can load and use reliably.

This file must be copied into the SD card’s `./MG3/Texts` directory. The Firmware will try to load the first file with this extension that is placed here.

2.1.5. Saving the data

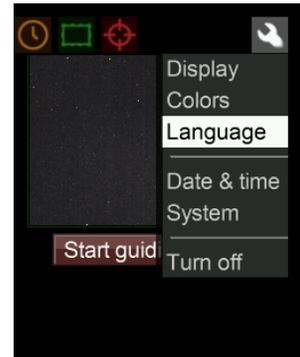
Currently saving is done at the time of closing the main window. You can export binary files from the current data but the `.strdef` file will be updated only at exiting the App.

When the text data were changed and new file is about to be saved, backup files are created. These files have the same name extended with `.bckN`, where N stands for the index of the backup file. (The lowest index of 1 is always the latest backup.) If you did a serious and unwanted change to your data and want to restore it, simply copy a proper backup file to your base data file. (Of course when the App. is not running.) The last 9 backup files are kept only.

3.Using the translation in the Fw.

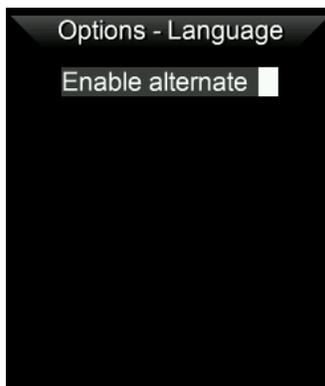
The external *string definition binary* file can be loaded by the Firmware from the SD card. You need to place the file into the SD card's `./MG3/Texts` directory. The Firmware always tries to load the first file with the extension `.m3tx` here.

The option for the external texts (called as “alternate language” in Fw.) can be found at the “Options” menu’s 3rd entry (“Language”) from the top. (When you accidentally load an unreadable font/text, you can find this option step-by-step to check off.)

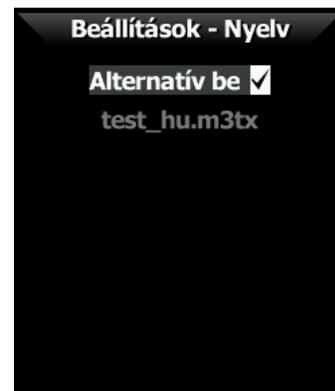


On this screen there is only the switch for enabling or disabling the alternate language.

When this option gets checked, the Firmware tries to load the binary file. If succeeds, the font is altered immediately and the filename of the binary is displayed under the checkbox. If there’s no filename shown, the default system font is being used with the default English texts. (Some error must be with reading the file.)



Tahoma bold, hungarian
→
→
→



3.1. Restrictions and notes

You should know some restrictions and notes on using custom text and fonts.

- Characters' width is varying, there's no guarantee (except for the system font) that the texts will not be cut or "drawn over the limits" if the font is too thick. Thinner fonts would rarely cause such problems.
- The width of the numeric characters (0..9) should be equal. Though it is not a "required" property, the Fw. development may rely on this and you could see some anomalies when displaying/editing numbers.
- There is a memory limit of 64k Bytes for the alternate font and also 64k Bytes for the alternate text definitions. This may introduce some limit for languages with a lot of different and big/complex characters (e.g. Chinese) or less probably when using very long texts.

Note that the size of the binary file is not equal to the required memory. If a binary would need too much memory, the App. is able to generate a valid file and warns you about this but the current Fw. probably won't load it.