

Telegram **User's Manual**

1.2.0

Table of Contents

Introduction	3
Preface	4
Related Components	5
pykob Package	5
MKOB (Morse Key on Board)	5
MRT (Morse Receive & Transmit - Mr T)	5
Utilities	5
Quick Start	6
User's Guide	7
Keyboard Commands	8
Indicators	9
Exiting	10
Command-Line Options	11
Telegram Configuration File (tg_config.tgc)	13
JSON Format	14
Entries	14
font	15
font_bold	15
font_italic	15
font_size	15
text_color	15
page_color	15
page_width	16
masthead_file	16
masthead_text	16
masthead_font	16
masthead_font_size	16
masthead_text_color	16
welcome_msg	17
top_margin	17
bottom_margin	17
side_margin	17
wrap_columns	17
page_clear_idle_time	17
page_advance_seconds	18
page_new_on_key_open	18
list_sender	18
Index	19
Colophon	21
Dedication	22

Introduction

Telegram is specifically designed for Telegraph displays (museums and the like). It runs full-screen and presents a (mock) telegram form. Messages received from the wire are sounded and decoded and displayed in the body of the form. At the end of the message a new form is scrolled into place. Or after a configurable amount of idle time, a new, blank, form is shown, ready to display the next message. In addition to receiving messages from the wire, messages can also be entered on a local key or with the keyboard. These are sounded, displayed, and sent out on the wire.

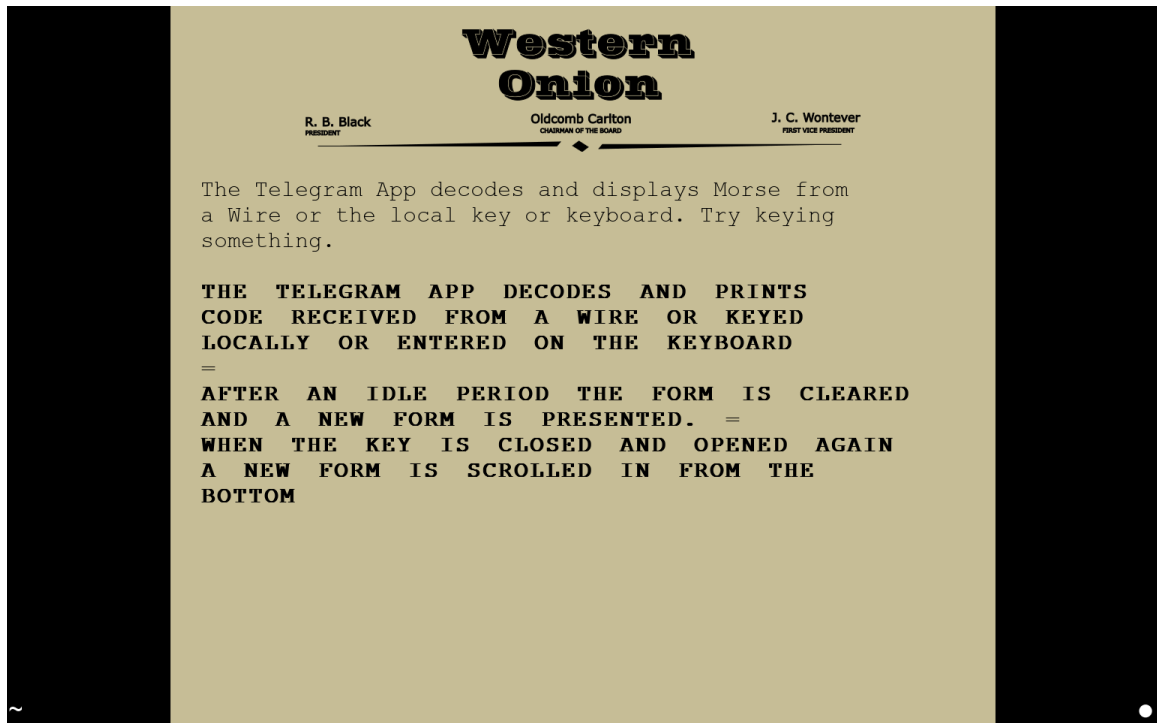


Figure 1. Telegram app screen

Preface

Telegram is a member of the MKOB Suite and uses the same configuration for the Morse, hardware, and other options, either Global or named, as MKOB, MRT, and the utilities do.

This user's manual covers the key and unique features of Telegram, while referring to other documentation for details on common functionality like creating and modifying configurations, selecting interfaces for the key and sounder, etc.

Related Components

pykob Package

Telegram uses the **pykob** package of modules. The pykob package consists of a number of modules that provide different aspects of handling code, the physical key and sounder, internet connection to a KOBServer instance, system audio to simulate a sounder or provide tone, configuration and logging capabilities, and more. When updating Telegram, it is advisable to get the latest (associated) release of pykob as well.

MKOB (Morse Key on Board)

MKOB is the full, graphical, user application. For most interactive use, MKOB is probably the application to use.

There is a full, illustrated, User's Guide that describes MKOB and its operation. Please refer to the **MKOB User's Guide** for descriptions of the core functionality, creating and using configurations, interfacing a key and sounder, using the *Virtual Closer*, and more.

MRT (Morse Receive & Transmit - Mr T)

MRT is a command-line/terminal application that is similar to MKOB in its ability to receive and send code. However, it is completely text based, allowing it to be run via a remote terminal connection to a system, or even automatically in the background (headless). It uses the same pykob configurations as MKOB (and the utility applications). In addition, it has a few options that allow it to support displays and similar applications.

There is a separate User's Guide for MRT.

Utilities

The pykob package also includes a number of utility applications. The README file included in the package contains a short description of each of the utility applications.

Quick Start

Get started with Telegram by using **MKOB** or **Configure** to set up your Morse, hardware, etc. configuration the same as for MKOB, MRT, and the utilities. Copy the example `tg_config.tgc` Telegram configuration file from the Telegram Documentation folder to the directory that you are running Telegram from. Copy the `WesternOnion-Masthead.svg` file from the Telegram Documentation folder to the directory that you are running Telegram from. These two files configure the look and operation of the telegram form and provide the form masthead content.

Now launch Telegram.

Telegram accepts a number of command-line options. Use the `--help` option to get a description of the available options.

When run, Telegram will display incoming code as text. It accepts text typed on the keyboard and converts it to code, which is sent to the wire (if connected). It also listens to an attached key and sends the code to the wire, as well as decoding and displaying it. The Telegram configuration file (**`tg_config.tgc`**) is described in a following section.

User's Guide

Telegram automatically connects to the wire when it is started. The F8 key can be used to disconnect/connect during operation. However, there is no option to change wires. To change wires, Telegram needs to be restarted with the new wire number.

The exception to automatically connecting is if wire number 0 is specified. In that case, Telegram does not connect to a wire, and can be used for local (only) displays.

Telegram monitors the wire and the local key. If the local key is opened, keyed code will be sent to the wire. While the local key is closed, Telegram will sound, as well as decode and display, code received from the wire.

Keyboard Commands

Telegram automatically connects to the wire, either from the configuration or specified on the command line. There are four keyboard commands:

1. '~' - Open the key.
2. '+' - Close the key.
3. ESC - Toggle the key open/closed state
4. F8 - Toggle the wire connection (connect/disconnect)
5. F11 - Clear the form without showing a new Masthead
6. Ctrl-A - Advance in a new form (scroll current form up)
7. Ctrl-X - Clear the form (instantly show a new form)
8. Ctrl-Q - Quit/Exit
9. Ctrl-C - Quit/Exit

If the key is open (either physically or by using '~'), other text typed will be sounded, displayed, and encoded and sent to the wire if connected.

(Simple, right!)

Indicators

Telegram displays two indicators in the bottom corners of the screen.

Bottom Left:

- '+' - The key is closed
- '~' - The key is open

Bottom Right

- '⬤' (filled circle) for Connected to Wire
- '◯' (open circle) for Disconnected from Wire

Exiting

Telegram runs continuously once started. To stop, enter either Ctrl-Q (^Q) or Ctrl-C (^C) on the keyboard.

Command-Line Options

Telegram is controlled by command-line options in addition to the MKOB/PyKOB Configuration being used (a command-line option allows specifying which configuration to use, rather than using the global configuration).

The two options/parameters specific to Telegram are:

- `--tgcfg path_to_telegram_cfg_file` - Allows specifying the path to a Telegram Configuration file with a different name or in a different location than 'tg_config.tgc' in the current directory.
- `wire` (as the last parameter) The wire number to connect to or '0' to not connect to a wire. If this parameter isn't included, the wire from the configuration is used.

The rest of the options are the standard options used to override values from the (main) Configuration.

It is suggested that the `--help` command-line option be used to get the up-to-date description of the options available. The following list is provided for convenience.

```
pygame-ce 2.5.2 (SDL 2.30.8, Python 3.11.4)
usage: Telegram.py [-h] [--tgcfg tg-cfg-file] [-a sound] [-A sounder]
                  [-g gpio] [-P serial] [-p portname] [-S station] [-c
wpm]
                  [-t wpm] [--config config-file]
                  [--logging-level logging-level]
                  [wire]
```

Telegram - Display a telegram form with local and received messages.
Telegram
specific configuration is in the 'tg_config.tgc' file.

positional arguments:

wire	Wire to connect to. If specified, this is used
rather	than the configured wire. Use 0 to not connect to
a	wire (local only).

options:

-h, --help	show this help message and exit
--tgcfg tg-cfg-file	Telegram specific configuration file to use. By default Telegram looks for the file
'tg_config.tgc' in	the current directory. This option specifies the
path	

```

to a '.tgc' (json) file to use.
-a sound, --sound sound
    'ON' or 'OFF' to indicate whether computer audio
    should be used to sound code.
-A sounder, --sounder sounder
    'ON' or 'OFF' to indicate whether to use sounder
if
    'gpio' or 'port' is configured.
-g gpio, --gpio gpio
    'ON' or 'OFF' to indicate whether GPIO (Raspberry
Pi)
    key/sounder interface should be used. GPIO takes
    priority over the serial interface if both are
    specified.
-P serial, --serial serial
    'ON' or 'OFF' to indicate whether a Serial
key/sounder
    interface should be used. GPIO takes priority over
the
    Serial interface if both are specified.
-p portname, --port portname
    The name/ID of the serial port to use, or the
special
    value 'SDIF' to try to find a SilkyDESIGN-
Interface,
    or 'NONE'.
-S station, --station station
    The Station ID to use (or 'NONE').
-c wpm, --charspeed wpm
    The minimum character speed to use in words per
    minute.
-t wpm, --textspeed wpm
    The morse text speed in words per minute. Used for
    Farnsworth timing. Spacing must not be 'NONE' to
    enable Farnsworth.
--config config-file
    Configuration file to use. The special value
'GLOBAL'
    will use the global (un-named) configuration. The
    special value 'NEW' will use a new (defaults)
    configuration.
--logging-level logging-level
    Logging level. A value of '0' disables DEBUG
output,
    '-1' disables INFO, '-2' disables WARN, '-3'
disables
    ERROR. Higher values above '0' enable more DEBUG
    output.

```

Telegram Configuration File

(tg_config.tgc)

While the configuration that you create and set up using **MKOB** or **Configure**, that is used by **Telegram**, controls the Morse encoding and decoding, what hardware you have and how it's connected, your Office/Station name, and other aspects of the operations, a separate configuration file, specific to **Telegram** controls the look and operation of the telegram form.

The Telegram Configuration controls:

- Form (page) color
- Text font
- Text font size
- Text color
- Text bold (true/false)
- Text italic (true/false)
- Side (left/right) margin
- Bottom margin
- Top margin
- Text wrap column (typewriter bell column)
- Page (form) spacing (blank space between forms)
- Masthead (top banner/graphic) file
- Masthead text (used if the masthead file isn't specified/found)
- Masthead font (used for the masthead text)
- Masthead font size
- Masthead text color
- Welcome message (shown on completely new form)
- Display sender office/station ID (true/false)
- Page (form) advance on key opened
- Page (form) clear idle time
- Page (form) page scroll speed

The example `tg_config.tgc` file is in the Telegram Documentation folder

JSON Format

The Telegram Configuration file is a JSON format file, as used by JavaScript, Python, and many other programming languages. Familiarity with JSON can be helpful, but is not required to be able to create or modify the configuration to meet your needs. It is suggested that you start with the provided file, as that will give you everything you need, and you should be able to simply modify some of the values.

Entries

The file is JSON Format Map/Dictionary. Below is an example file. The following sections describe each of the value options. Although the example is provided here, it is recommended that you use the file provided as a starting point, as it is used for testing.

```
{
  "font":"courier",
  "font_bold":true,
  "font_italic":false,
  "font_size":30,
  "text_color":"black",
  "page_color":"(198,189,150)",
  "page_width":1000,
  "masthead_file":"WesternOnion-Masthead.svg",
  "masthead_font":"Vineta,Century725,Times New Roman,Times",
  "masthead_font_size":40,
  "masthead_text":"Western Onion Telegraph Co",
  "masthead_text_color":"(0,0,10)",
  "welcome_msg":"The Telegram App decodes and displays Morse from a Wire
or the local key or keyboard. Try keying something.",
  "top_margin":30,
  "side_margin":40,
  "form_spacing":10,
  "list_sender":false,
  "page_clear_idle_time":15.5,
  "page_advance_seconds":2.2,
  "page_new_on_key_open":false,
  "wrap_columns":8
}
```

If an entry doesn't exist in the Telegram config file, the default value will be used.

Telegram Config Entry Descriptions:

font

Specifies the font to be used for the messages received or entered locally. This is a font name as found on the system. Many text editors and graphics applications provide a mechanism to see the various fonts available on the system. This entry can also be a comma separated list of font names. An example of that is used in the `masthead_font` entry in the file. When a list is used, the list is scanned from left to right and the first font found on the system will be used.

Default: courier

font_bold

Specifies whether the text used for messages should be bold. The value must be `true` or `false`.

Default: false

font_italic

Specifies whether the text used for messages should be italic. The value must be `true` or `false`.

It is valid to specify both bold and italic as `true`.

Default: false

font_size

Specifies the size of the text used for messages. The size is the font *pixel* height.

Default: 20

text_color

Specifies the color of the text used for messages. This can either be a standard system color name, or a RGB (Red,Green,Blue) value. For example, the values "black" and "(0,0,0)" result in the same black text. For anything but the basic colors: BLACK, WHITE, RED, GREEN, BLUE it is suggested that a RGB (Red,Green,Blue) value be used.

Default: black

page_color

Specifies the color of the page (form) background. See **text_color** for a description of how to specify a color value.

Default: (198,189,150)

page_width

Specifies the width of the page (form). The value is an integer. This will be capped at the width of the screen. For a nicer looking display, it is best to have this be narrower than the screen, so that there is a black band on the left and right.

Default: 980

masthead_file

Specifies a graphic file to be used for the Masthead of the form. Testing is done with SVG and PNG format files. If SVG is used, any text must be converted to paths in order to be displayed. If PNG is used, you will probably want to use a transparent background. Other graphics formats may work, but are not tested.

Default: <none>

masthead_text

Specifies the Masthead text (content) to use if a Masthead file isn't specified, or the file can't be found or loaded.

Default: <none>

masthead_font

If a Masthead file isn't specified, or the file can't be found or loaded, it is possible to specify Masthead text to be used instead. This property specifies the font to be used. See font above for valid values.

Default: <uses the text font>

masthead_font_size

Specifies the font size to use for the Masthead text. Not used if a Masthead file is used. See font_size above for valid values.

Default: <uses the text font size>

masthead_text_color

Specifies the color of the Masthead text. See **text_color** for a description of how to specify a color value.

Default: black

welcome_msg

A message that is shown below the masthead when Telegram is started, the idle time is reached, or Control-X is entered.

Default: <none>

top_margin

Specifies the space, in pixels, between the top of the page (form) and the Masthead.

Default: 38

bottom_margin

Specifies the space, in pixels, to maintain between the text and the bottom of the page (form).

Default: 42

side_margin

The space, in pixels, between the left and right sides of the page (form) and the text.

Default: 28

wrap_columns

Specifies the number of columns from the right margin that text will be wrapped on a space (if possible). If there aren't any spaces, the text will wrap/break at the right margin. The column width is based on the width of a space character of the text font specified.

Default: 8

page_clear_idle_time

Specifies the idle time, in seconds, before the page (form) is cleared (a new form, including Masthead, is displayed). A fractional value is allowed. Idle time is time when no code is received from the wire or entered on the local key or keyboard.

Default: 18.0

page_advance_seconds

Specifies the amount of time, in seconds, for complete page (form) to be scrolled off and a new page to be displayed. A fractional value is allowed. A value of 0 will instantly replace the current page with a fresh one (full form without scrolling). The special value of -1 indicates that the form is continuous, and is never replaced by a new form. If -1 is specified, the form can be cleared using the F11 key, or cleared and a masthead displayed using the Ctrl-X key.

Default: 2.8

page_new_on_key_open

If true, a new form is scrolled in when the key is opened (from being closed). If false, opening the key will keep the current form and start displaying the locally keyed code. In this mode a new form can be scrolled in using the Ctrl-A key if desired.

Default: false

list_sender

Specifies that the current sender's office/station ID be displayed when Telegram is started or the sender changes. The value must be `true` or `false`.

Default: false

Index

@

'+' - Close the key, 8
 '~' - Open the key, 8
 --help, 11

B

basic colors, 15
 bold, 15
 bold and italic, 15
 bottom_margin, 17

C

Configure, 6, 13
 Connected to Wire, 9
 Ctrl-A, 18
 Ctrl-A - Advance in a new form (scroll current form up, 8
 Ctrl-C - Quit/Exit, 8
 Ctrl-Q - Quit/Exit, 8
 Ctrl-X, 18
 Ctrl-X - Clear the form (instantly show a new form, 8

D

Disconnected from Wire, 9

E

ESC - Toggle the key open/closed state, 8

F

F11, 18
 F11 - Clear the form without showing a new Masthead, 8
 F8 - Toggle the wire connection (connect/disconnect, 8
 font, 15-16
 text, 16
 font_bold, 15
 font_italic, 15
 font_size, 15-16

G

global configuration, 11

I

Indicators, 9
 indicators, 9
 italic, 15

L

list_sender, 18

M

Masthead, 16-17
 banner, 16
 Masthead file, 16
 Masthead text, 16
 masthead_file, 16
 masthead_font, 16
 masthead_font_size, 16
 masthead_text, 16
 masthead_text_color, 16
 MKOB, 4-6, 13
 Morse Key on Board, 5
 MKOB Suite, 4
 MKOB User's Guide, 5
 MKOB/PyKOB Configuration, 11
 MRT, 4-5
 Morse Receive & Transmit, 5

P

package
 pykob, 5
 page_advance_seconds, 18
 page_clear_idle_time, 17
 page_color, 15
 page_new_on_key_open, 18
 page_width, 16
 pixel, 15
 pixel height, 15
 pykob, 5

R

RGB (Red,Green,Blue, 15

S

side_margin, 17

T

Telegram, 3-11, 13

Telegram Config Entry Descriptions, 14

Telegram Configuration, 14

Telegram configuration file, 6

Telegram Documentation folder, 13

text_color, 15-16

`tg_config.tgc`, 13

tg_config.tgc, 6

top_margin, 17

W

welcome_msg, 17

WesternOnion-Masthead.svg, 6

wire

 parameter, 11

wrap_columns, 17

Colophon

This document reflects the features of Telegram Version 1.2.0 Edit date: 12-22-2024

MIT License

Copyright (c) 2020-24 PyKOB - MorseKOB in Python

This document is authored using **asciidoc**, and the pdf version rendered through **asciidoctor-pdf**, to create a manual that is more readable and more pleasing to the eye.

Dedication

The team would like to thank everybody who through comments, criticism and suggestions has helped to make the Telegram application and this document better and more usable.

Special thanks go out to Les Kerr, who's vision and effort created the original MorseKOB application, and who initiated the Python version.