MKOB4
User's Manual

4.4.3

Table of Contents

Introduction

During the last half of the 19th Century and the first half of the 20th Century the steady and rapid fire click clacking of the Landline Telegraph sounder was how important news and events were transmitted over vast distances. This simple system transmitted news of great national events and small family ones. It transmitted business deals, birth and death announcements, the outbreak and fortunes of war, and the weather in nearby and far off places. In the old days you learned this system by working with the professionals who already knew how to use it. We are not so lucky these days as many of the old pros are gone.

MorseKOB by Les Kerr, originally written for Windows only, allowed us to learn, and more importantly USE, landline telegraphy!

Now rebuilt from scratch and updated, MKOB v4 runs on most computers big and small. With this program it is possible to once again hear the sounds of the Telegraph sounder as messages come in from distant places. It can be adjusted to make learning to read and send the code easier and, when you are ready, to connect to others over the Internet.

Welcome to the world of Landline Telegraphy and thanks for helping to keep "The sound of Morse Alive".

> 73 (Telegrapher's code for Best Wishes)
> Ciao
>
> — Kevin Jepson, Calgary (CG) Chapter of International Morse Telegraph Club.

Preface

Like all computer programs MKOB has lots of configuration settings and tricks for making the best use of it. This user's manual attempts to cover all of them and gives a good idea of how to set it up.

Don't let the apparent complexity intimidate you. At its heart MKOB is simply a tool for learning and using landline telegraphy on a computer. MKOB runs on Windows, Linux, and Mac, and people are porting it to other systems like phones and embedded devices. It can be used with a real telegraph key and sounder or it can realistically simulate them.

The Quick Start portion will get you up and running fast and when you want to do a bit more, dig into the details in the User's Guide portion.

> Although MKOB's functionality, and many of the user actions, are similar to MorseKOB 2.5, it is not identical, and it is not intended to be. This document describes the functionality and user actions of the MKOB4 application and, typically, does not compare MKOB4 to MorseKOB 2.5. Knowledge of the MorseKOB 2.5 application is not assumed or required.

Related Components

pykob Package

MKOB 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 MKOB, it is advisable to get the latest (associated) release of pykob as well.

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 document that describes MRT and its options.

Telegram

Telegram is an application specifically designed for use in displays. Telegram presents a full screen interface with no exposed controls. Its interface appears as a telegram form that fills the entire screen. Received messages are displayed on the form. The form is cleared, or scrolled off, at the end of each message, or after a configurable period of idle time. The appearance of the form is configurable without changing any code.

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 MKOB and have some fun. Once you've had a chance to try things out, make sure to read the User's Guide portion to get the most out of MKOB.

Main Screen

The main screen of the MKOB app looks like this on Windows.[1]
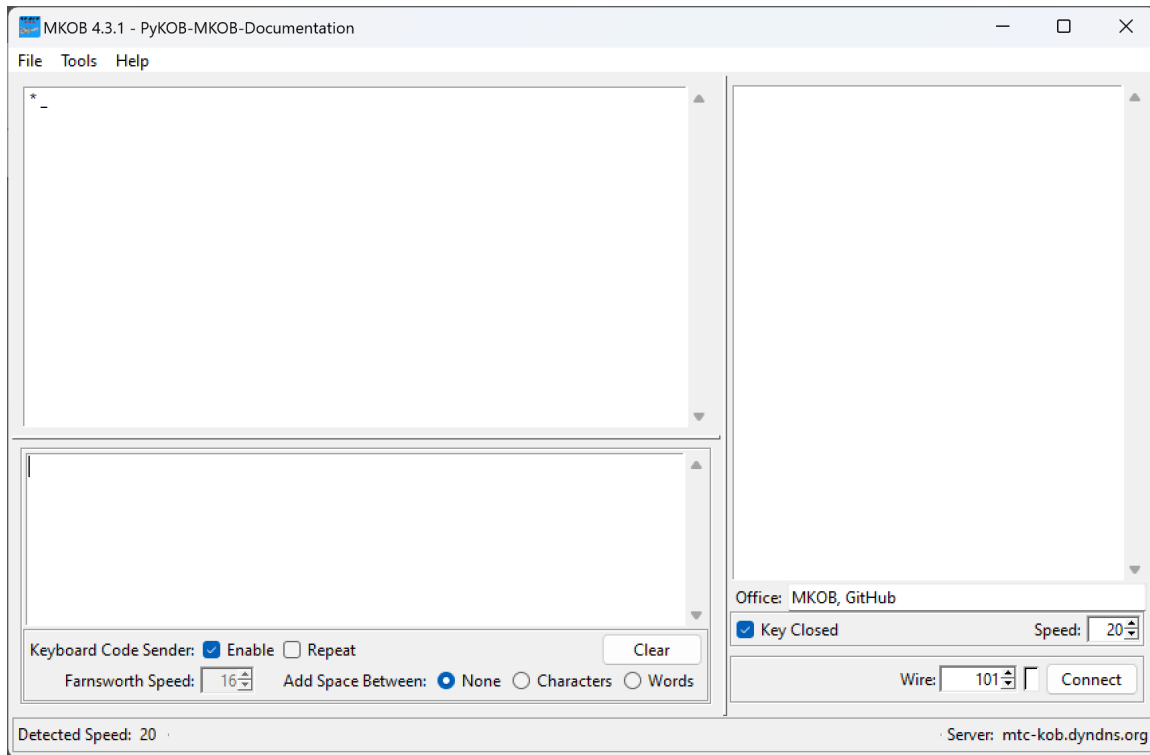
Figure 1. Main MKOB app screen

The main screen contains

Three main window areas:

1. Code Reader (Decoder) area - Displays decoded code received from a wire or generated locally

2. Offices/Stations Connected list - List of offices/stations connected to the current wire

3. Keyboard Sender text area - Text entry/editing area for generating code locally

Two control sections:

1. Wire operation - Key/circuit closed control, speed, wire number, and connect button and indicator

2. Keyboard Sender operation - Sender enable, sender repeat, and Farnsworth speed

< MORE TO COME… >

[1] Screenshots are from the Windows platform. The interface looks similar on Mac and Linux platforms.

User's Guide

Landline, Closed Circuit, Telegraphy

ℹ️ Even if you are very familiar with Landline Telegraphy, it is suggested that you at least read through the section Circuit and Key Closer and MKOB's Virtual Closer to get the best results and experience using MKOB.

This section provides a very brief description of Landline, Closed Circuit, Telegraphy, that is important for the effective, and enjoyable, use of the MKOB application.

One of the primary features of MKOB (and the pykob modules and MRT) is support for interfacing to a physical 'closed'/'loop' telegraph circuit and simulating the operation of a loop telegraph circuit locally and connected to a KOBServer hub via the internet.

This simulates the operation of Landline-Telegraphy and a closed-loop circuit as used by virtually every railroad, Western Union, and many other communication services in the United States, and in fact, North America and beyond, when telegraphy was THE means of communication. In fact, for railroads, that lasted well into the 1960's (and in some areas, even into the 70's).

Describing closed loop, landline, telegraphy is not the focus of this document. There are a number of sources available online or at the library that describe it. However, there is one aspect of closed loop, landline, telegraphy that is critical to understand for successful use of MKOB (and MRT and some of the utilities), and for connecting to a KOBServer and communicating with others – that is the aspect of the circuit being open or closed, and use of the Closer on the telegraph key.

Telegraph Key Closer

Landline telegraphy relies on a connection between each and every station in the network. Again, it is beyond the scope/intention of this document to describe how this was achieved, but you can think of it (in concept anyway) like the lighting wiring in your house. Power goes through the wall switches in the room (sometimes multiple for 3/multi-way switches) and is connected to the lights. When all the switches are closed the lights are on. If any of the switches are open the lights are off.

In the telegraph line the operators in each office/station have a telegraph key that is wired into the circuit. If the key is opened, the power to the circuit is cut and all the sounders in the circuit are de-energized (producing the distinctive 'clack' sound). When all the operator's keys are closed, the power to the circuit is restored and all the sounders are energized (producing the distinctive 'click' sound). To communicate, all the operators at the offices/stations in the circuit/loop line keep their keys closed except for the one that wants to send a message. That operator Opens their key and then starts sending the message by actuating the key (pressing and releasing it).

A telegraph key designed for landline telegraphy includes a part referred to as The Closer. Keys designed for radio often do not include a Closer.

Below are pictures of a landline telegraph straight key and a "bug" type semi-automatic key showing the Closer.

The Closer on a J-38 Telegraph straight key.

Vibroplex 'BUG' keyer showing the Closer

The flow for communicating using landline telegraphy is:

- Use the Closer on the key to keep your key closed when you are not sending (all other operators do the same).

- When you are ready to send, listen for a pause in activity, then open the Closer on your key.

- This (opening your key) will cause all sounders in the circuit to de-energize – producing the distinctive 'clack'.

- Send your message. This will cause all sounders in the circuit to 'follow' your key (energizing and de-energizing as your key is closed and opened).

- When you finish sending your message, close your key. That will energize the circuit causing all the sounders to produce the distinctive 'click'.

- This process repeats, for each operator that sends a message.

Circuit and Key Closer and MKOB's Virtual Closer

(Important for using MKOB)

MKOB tracks the open/closed state of the key attached to the interface and will show its state in the Key Closed checkbox in the interface. In addition to following the state of the physical key, MKOB implements a Virtual Closer.

Typically, the state of the Virtual Closer will be the same as the physical key's Closer. However, if the physical key does not have a closer (for example, when using a key designed for radio), or there isn't a physical key attached, the MKOB's Virtual Closer can be used to control the open/closed state of the circuit – allowing code to be sent to a wire from the key or from the Keyboard Sender.

In addition, the Virtual Closer can actually be used to override the state of the physical key. Allowing code to be sent from the Keyboard Sender (including the Keyer keys) even if the physical key is closed.

Understanding the functionality and use of the Virtual Closer in MKOB is important to the successful (and enjoyable) operation of the application and communicating on wires with others in the best simulation of a true landline telegraph circuit using the world-wide connectivity afforded by the internet. Could Morse and Vail have ever envisioned this?!

Comparison to MorseKOB 2.5

This is the one place where the operation of MKOB will be compared to that of MorseKOB 2.5. If you do not have experience with MorseKOB 2.5 you can skip the rest of this section.

MorseKOB 2.5 doesn't track the state of the physical key's closer in its interface. It treats its keyboard sender, its system (computer) audio sounder, and its Circuit Closer checkbox as a 'separate circuit' that is connected in series with the connected physical key and sounder (loop) and the KOBServer wire (circuit) that others are also connected into.

The operation of the Virtual Closer in MKOB was designed to allow the use of a physical key and sounder, with occasional use of the Keyboard Sender without requiring interaction with the MKOB user interface. The circuit is opened and closed using the physical key's closer and, when needed, text can be entered using the keyboard, then you can go directly back to the physical key to continue sending. When you are done, you close the key's closer, and you are ready to receive code from the other operators connected to the wire.

However, when using a KOB/Loop you need to have the physical key closed in order for the sounder to sound. If you want to send from the Keyboard Sender and hear the code on the sounder, you can leave the physical key closed and uncheck the Key Closed box and then send from the keyboard. Opening and closing the physical key will once again sync up the

Key Closed state in the interface to the state of the key. Similarly, with the key's closer open, you can check the Key Closed box and MKOB will treat the circuit as closed. Again, closing the key will re-sync the interface to the state of the key.

If you don't have a physical key and/or sounder connected, that's fine. Just use the Virtual Closer, the Keyboard Sender, and the System sound.

Overview

The functionality of MKOB will be covered in sections relating to the sections and controls on the screens.

Main Sections and Controls
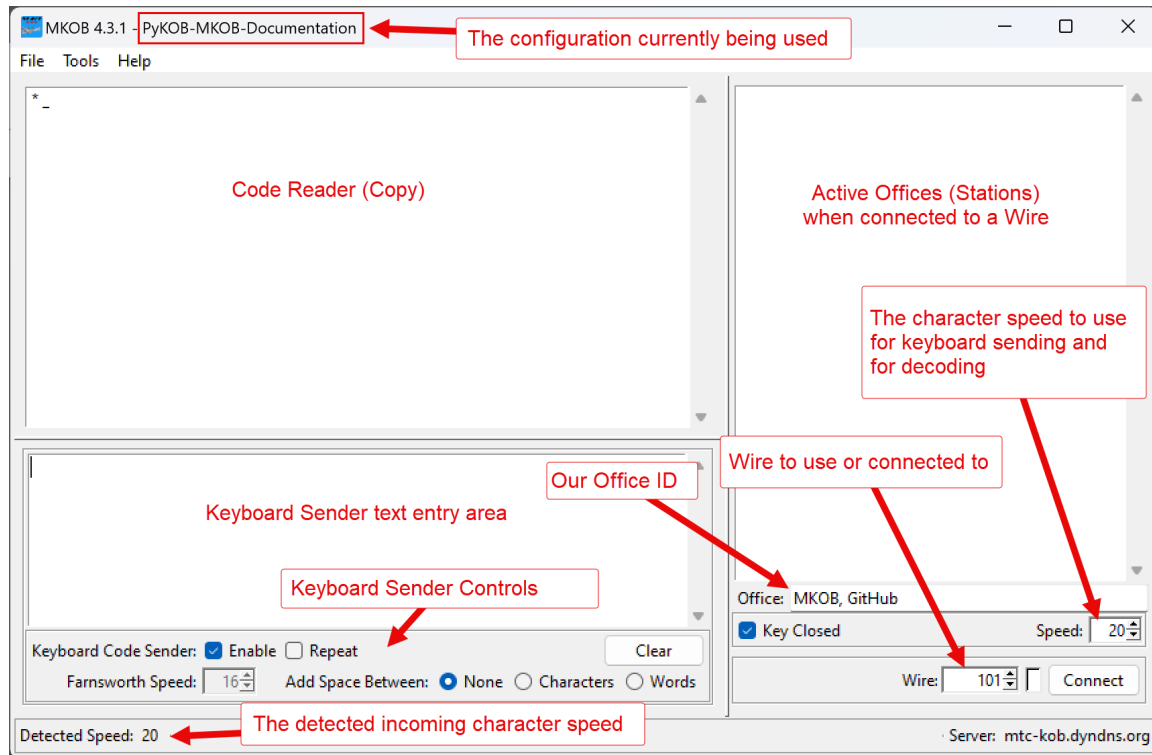
Figure 2. Main MKOB app screen

Code Reader (Copy)

Copied (decoded) code is displayed in this window. Code is copied from the local key and from stations on the wire when connected. The Reader window is also occasionally used to display information from the MKOB application to the user.

Code Format and Information Display

Some information about the received code and some formatting is performed by the Reader as the code is displayed.

When you connect to a wire and code is first received, the Office/Station ID of the sender is printed within angle brackets - *<Office ID>*. The Office/Station ID is also printed when the sender changes. When a paragraph mark ('=') is received the Reader will go to a new line on the next non paragraph mark received. When the sender closes their key an underscore is printed. When you disconnect from a wire (or get disconnected) *'#####'* is printed. If the reader detects that the spacing of a character isn't as expected it displays the character with a small under-tick.

Active Offices (Stations)

Offices that are currently connected to the wire are listed in this window. When not connected, this station will be listed in this window when code is sent from it.

Order and Format of Listed Offices

The order of the offices and the formatting used can be helpful when connected to a chat-type wire with a number of offices connected. The offices are listed as follows:

- The top/first office is the current sender.

- After the current sender (and a line), other offices/stations that you have not received code from will be listed in order of how long they have been connected (as observed by this station), starting with the most recent. These are indented.

- After the indented list, stations that you have received code from are listed from the least recently received from to the most recently received from (the last sender).

This ordering can help if you are the current sender and you are trying to decide who to turn the wire over to next. It is important to realize that the order is based on activity seen since you connected. The wire traffic doesn't contain history information, and therefore it is possible that an office listed near the top and indented, has actually sent some information before you connected to the wire. However, it can still be helpful, especially if you have been connected to the wire for a while.

Keyboard Sender

Text entered in this window, either from the keyboard or loaded from a file, will be coded at the current character/text speed when the Keyboard Code Sender (Enable) checkbox is checked, and the key is open (either the physical key or the virtual key by unchecking the Key Closed checkbox). The Repeat checkbox will cause the sender to repeat when it reaches the end of the text.

You can use the cursor keys or click with the mouse to move the position of coding when the sender is enabled, or the text insert position when the sender is disabled.

Office (Station) Name

Text entry field used to enter/display the name of this office/station. The office/station name is used when you connect to a wire on a KOBServer. The Office/Station name can also be set in the Preferences dialog.

Key Closed (checkbox)

Indicates the state of the physical and/or virtual key, open/closed. It can be used to control the state of the circuit, overriding the physical key, if one is being used.

Speed

This is the character speed used for sending from the Keyboard Sender and 'Keyer' keys (Shift-Left and Shift-Right when the Keyboard Sender is enabled), and for decoding (copying) incoming code. See the section in Preferences about character speed, text speed, and spacing, for information on how to use Farnsworth.

Menu Bar

File Menu

The file menu contains four sections.

Section 1 - Keyboard Sender

Operations that apply to the Keyboard Sender.

New

Clear the Keyboard Sender window. This is retained from earlier versions for compatibility. It does the same as the Clear button in the Keyboard Sender area or the F12 hotkey.

Open…

Allows selecting a text file. The contents of the text file are copied into the Keyboard Sender window. Within the text, two special characters are recognized to allow the (virtual) key to be opened and closed. * '~' Opens the virtual key. * '+' Closes the virtual key.

Section 2 - Recording and Playback

Operations that control session recording and playback.

Session recording and playback provides a great way of practicing your ability to copy hand-keyed code (listening to distinctive 'fists', as it is referred to). Recording a 'Wire 11' (or other 'live') session, and them playing it back, allows you to hear the code as it was sent. Refer to the Record & Playback section for more information.

Record

Start recording the current session. The recording uses an automatically generated file name. The recording can be played using File:Play… or using the Play.py command line utility.

End Recording

End recording of the current session.

Play…

Allows selecting a PyKOB recording file (.pkrec or .json file). The selected recording file will start playing. Refer to the Record & Playback section for more information about controlling

playback.

Section 3 - Application Preferences (Configuration)

Operations pertaining to application preferences (configuration).

Preferences…

Opens the Preferences dialog to allow setting preference/configuration options. Refer to the Preferences section for more information.

Load…

Allows selecting a configuration file that is then loaded and used as the current configuration.

Save

Saves the current configuration. This saves to the configuration that is currently in use (named configuration file or the global configuration).

Save As…

Saves the current configuration to a named configuration file. The file can be an existing configuration file or a new configuration file. This will not save to the global configuration.

Load Global

Load the global configuration and make it the current configuration.

Save Global

Save the current configuration as the global configuration.

Section 4 - Exit

Exit

Exit MKOB.

Tools Menu

Application Tools and Utilities

Show Packets

When connected to a Wire this will display information about the packets of information sent to the wire and received from the wire.

This is intended primarily for program debugging, but the information can be educational as well.

Key Timing Graph…

Open the Key Timing Graph.

The Key Timing Graph gives a visual representation of the key press and release durations of entered or received code.

Refer to the separate Key Timing Graph section for more information.

Help Menu
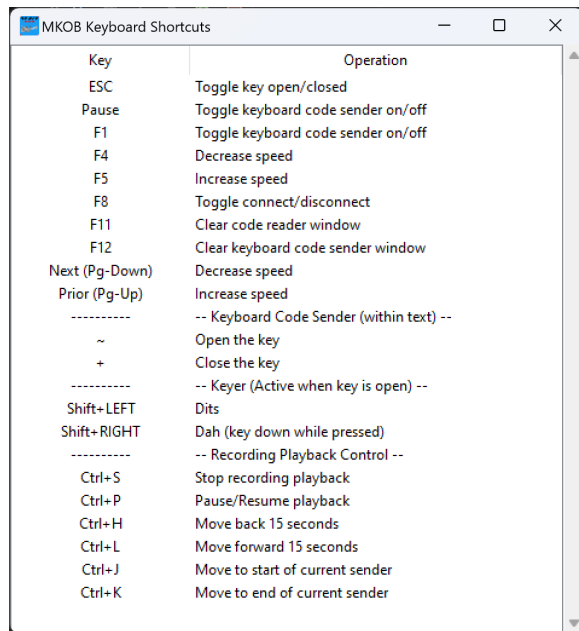
Helpful Information and Actions

Active Wires

Opens a separate window that displays the currently active wires on the KOBServer. The list can be used to select a wire to connect to (single click) and to connect (double click). If already connected, it will switch the wire.

Keyboard Shortcuts

Opens a separate window that displays the keyboard shortcuts available in MKOB.

Key	Operation
ESC	Toggle key open/closed
Pause	Toggle keyboard code sender on/off
F1	Toggle keyboard code sender on/off
F4	Decrease speed
F5	Increase speed
F8	Toggle connect/disconnect
F11	Clear code reader window
F12	Clear keyboard code sender window
Next (Pg-Down)	Decrease speed
Prior (Pg-Up)	Increase speed
----------	-- Keyboard Code Sender (within text) --
~	Open the key
+	Close the key
----------	-- Keyer (Active when key is open) --
Shift+LEFT	Dits
Shift+RIGHT	Dah (key down while pressed)
----------	-- Recording Playback Control --
Ctrl+S	Stop recording playback
Ctrl+P	Pause/Resume playback
Ctrl+H	Move back 15 seconds
Ctrl+L	Move forward 15 seconds
Ctrl+J	Move to start of current sender
Ctrl+K	Move to end of current sender

Figure 3. Keyboard shortcuts

About

Lists version information for MKOB, pykob, and some of the primary modules used.

Preferences

All of the application preferences (also referred to as the configuration) can be set using the Preferences Dialog accessed using the File:Preferences… menu. They can also be listed and set using the **Configure** command line utility. Some of the configuration can also be set using the operational controls in the interface.

The Preferences dialog contains three tabs that help organize the application settings.
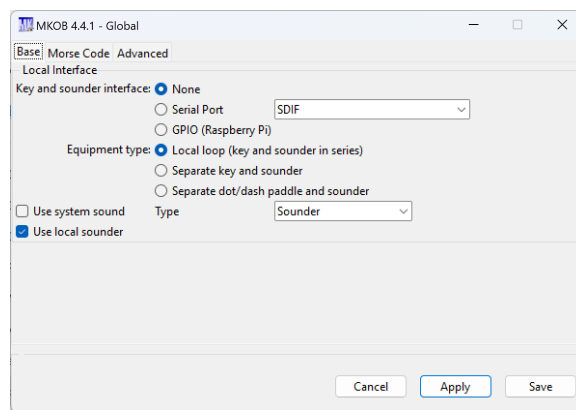
Base

Application hardware configuration.

Figure 4. Preferences base hardware/equipment settings

Local Interface

Specifies the interface to a physical key and sounder.

- Key and sounder interface:
    - None– No interface is to be used
    - Serial Port– A serial interface to either a key, a sounder, or a KOB, is used. The drop-down menu contains a list of serial devices found. An interface value that isn't currently available (or doesn't show up) can be entered manually.

      The special value *SDIF* can be manually entered to have MKOB look for a SilkyDESIGN-Interface when it runs.

    - GPIO (Raspberry Pi) – The GPIO on a Raspberry Pi is used to interface to a key, a sounder, or a KOB.
- Equipment type:
    - Local loop (key and sounder connected in series) – This is often considered 'the traditional configuration', as it is how actual telegraph stations were set up. This requires a special loop interface in order to be able to detect the key as well as drive

the sounder with them connected in series. This is how a KOB (Key on Board) is wired.

- Separate key and sounder – This should be used when the interface connects to the key and sounder individually. It should also be used, if only a key or only a sounder is being used. For example, when using a physical key and the system (computer) sound.

- Separate dot/dash paddle and sounder – This is currently under development. It will be used when a dot/dash (keyer) device is used along with a sounder or tone output. The Keyboard Sender does provide 'keyer' functionality using the Shift-Left and Shift-Right keys.

- Use system sound: This checkbox enables the use of the system sound to simulate a sounder or provide a tone.

    - Type: drop-down selection of

        · Sounder – Provide a simulated sounder ('click' (down), 'clack' (up))

        · Tone – Use a radio-style (CW) tone

- Use local sounder: This checkbox enables the use of the local sounder (when one is connected). Disabling the local sounder can be helpful when one wants to practice without disturbing others. The system sound can be used with headphones while still being able to use the physical key. This even works with most loop interfaces and a KOB.

New in MKOB version 4.4.1, the Key and sounder interface can be set to *None* while selecting/retaining a value for the port (the value in the entry field). This allows disabling the interface while keeping the port value for later use.

Morse Code

Controls the Keyboard Sender sending and Code Reader decoding.
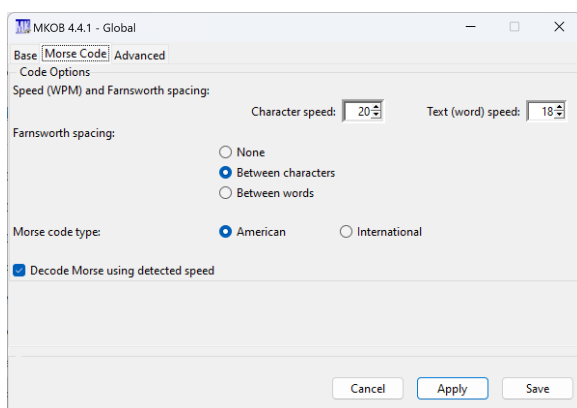

Figure 5. Morse encoding/decoding settings

- **Speed (WPM) and Farnsworth spacing**: The Farnsworth Method is a technique to aid in learning to copy code. Using the technique, characters are sent at a faster speed than the overall speed of the text. This can help you by listening to characters at the speed

you wish to achieve, while making it somewhat easier to copy the full content, as it is being delivered at a lower speed.

- **Character speed**: The speed the characters will be sent and decoded at
- **Text (word) speed**: The speed the overall text will be delivered at The text speed will always be less than or equal to the character speed.

   The speeds and spacing option can also be set in the main application interface.

- **Farnsworth spacing**: Controls where space (additional time) will be added to effectively slow down the overall speed of the delivered text.

  - **None** – No additional space will be added. Selecting this will disable the Text (word) speed selector, as the overall speed will be the character speed.

  - **Between characters** – Space, as needed, will be inserted between each character sent.

  - **Between words** – Space, as needed, will be only be inserted between words. Words themselves, will be delivered at the full character speed.

   The speeds and spacing option can also be set in the main application interface.

- **Morse code type**:

  - **American** – Use American (Landline) Morse when sending from the Keyboard or decoding in the Reader.

  - **International** – Use International (Continental) code when sending from the Keyboard or decoding in the Reader.

- **Decode Morse at detected speed**:

  - When checked, MKOB will use the speed that it is detecting for the incoming code to decode and display the text.

  - When unchecked, MKOB will use the configured speed (the **Character speed**) to decode and display the text.

Advanced

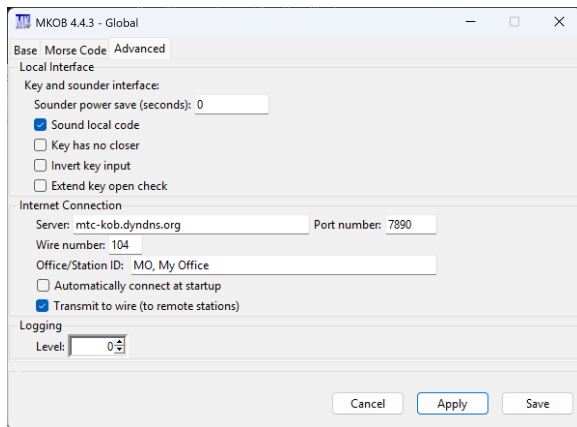Controls other aspects of the application that are less frequently changed.

Figure 6. Advanced (less frequently used) settings

Local Interface

Additional control of the local, physical, interface.

- **Key and sounder interface**:

  - **Sounder power save (seconds)** Since a closed-loop telegraph has the sounder energized whenever it is idle, this feature allows power to be saved by de-energizing the sounder after a period of no activity. When the specified number of seconds elapses with no activity of the sounder, the sounder will be powered off. Any activity, whether received from a connected wire or initiated on the local key or Keyboard Sender, will re-energize the sounder and normal operation will continue. This can be very helpful when the KOB/loop is being powered by a battery.

    A value of zero (0) disables this functionality.

  - **Sound local code**: This enables/disables the sounding of code keyed locally, from either the physical key or from the Keyboard Sender.

  - **Invert key input**: This inverts the state of the key. Some interfaces, most notably the modem interface, present the key state at the opposite level of what is considered 'normal', and this option allows their use.

  - **Extend key open check**: This extends the length of time that the key closer must be open before opening the circuit. When this option is enabled the circuit isn't considered open until the key closer has remained open continuously for almost a second (rather than immediately upon opening it). This option is primarily intended for use with a longer physical wire loop, where noise on the line might otherwise trigger a circuit open state.

Internet Connection

Configuration for connecting to a KOBServer and wire.

- **Server**: The hostname or IP address of the KOBServer

- **Port number**: The port to use for connecting to the KOBServer If the Server or Port (or both) are cleared, the application will use the default KOBServer at mtc-kob.dyndns.org and port 7890.

- **Wire number**: The wire number to connect to on the KOBServer The Wire number can be set in the main application interface.

- **Office/Station ID**: The Office ID used when connecting to the KOBServer Wire For an individual, this is typically a two-letter designator that identifies your office/station, followed by some brief space or comma separated information that may be interesting to others. For example, your initials, your location, etc. When selecting a designator, it is good practice to observe the active wires for a while and do your best to select something not being used by someone else. For a feed of some type, this is a description of the feed, including a designator identifying who is providing it. If you are interested in providing a feed, please contact the MTC to get a Wire allocated, and be added to the wire plan.

- **Automatically connect at startup**: This causes MKOB to connect to the configured wire as soon as it is started, without requiring the Connect button to be pressed.

- **Transmit to wire (to remote stations)**: Transmit keyed code from either the physical key or the Keyboard Sender to the KOBServer wire (when connected). Sending to the wire is typically what is desired, but this option can be used as a safeguard to keep you from accidentally sending code when you are practicing.

Logging

Configuration for controlling the amount of logging sent to the console.

- **Level**: The desired level of logging information

Information is logged at four (4) main levels:

Error – This is information that represents a serious error in the application operation. This generally means that some aspect of the application is not functioning and might even result in the application terminating. This is Level -3 (minus 3)

Warn – This is information about something that might cause, or be causing, the application to not operate as intended/desired. This is Level -2 (minus 2)

Info – This is information about the operation, configuration, or other aspect of the application or environment that might be interesting or helpful. This is Level -1 (minus 1)

Debug – Debugging information is actually provided at a number of levels and is intended to help in troubleshooting problems in the operation of the application. This is Level 1+ (1 and greater)

The **Level** value for this configuration setting is used to control what is logged:

- **0** – Log Error, Warn, and Info level information. Do not log any Debug information.

- **-1** – Log Error and Warn level information. Do not log Info or Debug information.

- **-2** – Log Error level information. Do not log Warn, Info or Debug information.

- **-3** – Do not log any information (completely silent).

- **1** (and greater) – Log Error, Warn, and Info level information, as well as Debug information designated as Level-X, where X is equal to or less than the value specified. The higher the number, the more debugging information that will be logged.

Cancel, Apply, Save

The buttons at the bottom all close the Preferences dialog. They allow you to:

- **Cancel** – Do not make any changes to the active configuration (even if changes have been made within the dialog).

- **Apply** – Apply changes that have been made to the active configuration, but do not save them. This is often used when making temporary changes, or when you want to save a new configuration based on the active/current configuration. To do that, refer to the additional information below.

- **Save** – Apply changes that have been made to the active configuration and save the changes out to the current configuration (a named configuration file or the global configuration).

Creating a New Configuration or Updating an Existing Configuration

Using the Preferences dialog and the File menu, new named configurations can be created or an existing configuration can be updated.

The steps are:

1. Start with an existing configuration. Do this by loading a configuration file or the global configuration using the File menu.

2. Use the Preferences dialog to make any changes desired.

3. Use Apply (not Save) to close the dialog.

4. Use the File:Save As… to save a named configuration file, or…

5. Use the File:Save Global to save to the global configuration.

The **Configure** command line utility can also read from one configuration, make changes, and save to a different configuration. Use the `--help` command line option of the Configure utility to see information about its operation.

Starting MKOB with a Specific Configuration (and multiple instances)

MKOB provides command line options that allow starting with a specific configuration. The option can be used to create desktop shortcuts or scripts that run MKOB in a specific way.

Multiple instances of MKOB can be running at the same time using different configurations to connect to different servers, different wires, or use different connected equipment.

This is done with a single installation of MKOB (pykob). There is no need to install MKOB (pykob) multiple times.

The command line option of interest is: * **--config config-file** The configfile is the path to a configuration file to be used.

Without this command line option, MKOB uses the last configuration loaded or saved.

If you intend to run multiple instances, and you are using a physical key/sounder. You will need to make sure that you specify different Serial Port values for each instance (if you have multiple interfaces connected) or specify None for the additional instances. If you don't do this, MKOB will still function, but you will receive an error dialog when the additional instances start, letting you know that the interface is busy/not available.

Typical Configurations

Here are recommendations for Preference/Configuration settings for some typical combinations of equipment.

Of course, these are not all the possibilities, but these are some of the most common.

Physical KOB (Key on Board) device connected using a 'loop' interface:

1. Select the serial port used to connect the interface.

2. Select 'Local loop' as the equipment type.

3. Check 'Use local sounder'.

4. Uncheck 'Use system sound'.

Physical Key & Sounder connected through the interface, but not connected in series:

1. Select the serial port used to connect the interface.

2. Select 'Separate key and sounder' as the equipment type.

3. Check 'Use local sounder'.

4. Uncheck 'Use system sound'.

Physical Key connected to the DSR line of a serial port (no sounder):

1. Select the serial port used to connect the interface.

2. Select 'Separate key and sounder' as the equipment type.

3. Uncheck 'Use local sounder'.

4. Check 'Use system sound' to provide the synthesized sounder.

5. (if using International Code, optionally select 'Tone' rather than 'Sounder')

Physical Sounder connected through an interface to the RTS line of a serial port (no key):

1. Select the serial port used to connect the interface.

2. Select 'Separate key and sounder' as the equipment type.

3. Check 'Use local sounder'.

4. Uncheck 'Use system sound'.

No Physical Equipment:

1. Select 'None' for 'Key and sounder interface'.

2. Uncheck 'Use local sounder'.

3. Check 'Use system sound' to provide the synthesized sounder.

4. (if using International Code, optionally select 'Tone' rather than 'Sounder')

Physical KOB connected using a 'loop' interface – for 'private' practice:

> If using the Les, Chip, or SilkyDESIGN Loop interfaces [1] and a KOB, MKOB allows you to disable the sounder, use the system sound (allowing you to use headphones), and still be able to key using the physical key on the KOB! This allows you to practice your physical keying without disturbing others in your house, even though your equipment is a Loop and KOB!!!

1. Select the serial port used to connect the interface.

2. Select 'Local loop' as the equipment type.

3. Uncheck 'Use local sounder'.

4. Check 'Use system sound'.

Plug in your headphones/earbuds and start keying. No excuse now!

---

[1] The Les (circuit), Chip interface, and SilkyDESIGN interface have been use to demonstrate this functionality. It is possible that other KOB/Loop interfaces will also work in this manner. If you have a different interface, and it works in this way, let us know and it will be noted.

Key Timing Graph

The Key Timing graph can be used for: * Practice and improvement of your keying. * Tuning an auto/bug key. * Adjusting the Speed setting for better decoding of received code. * more?
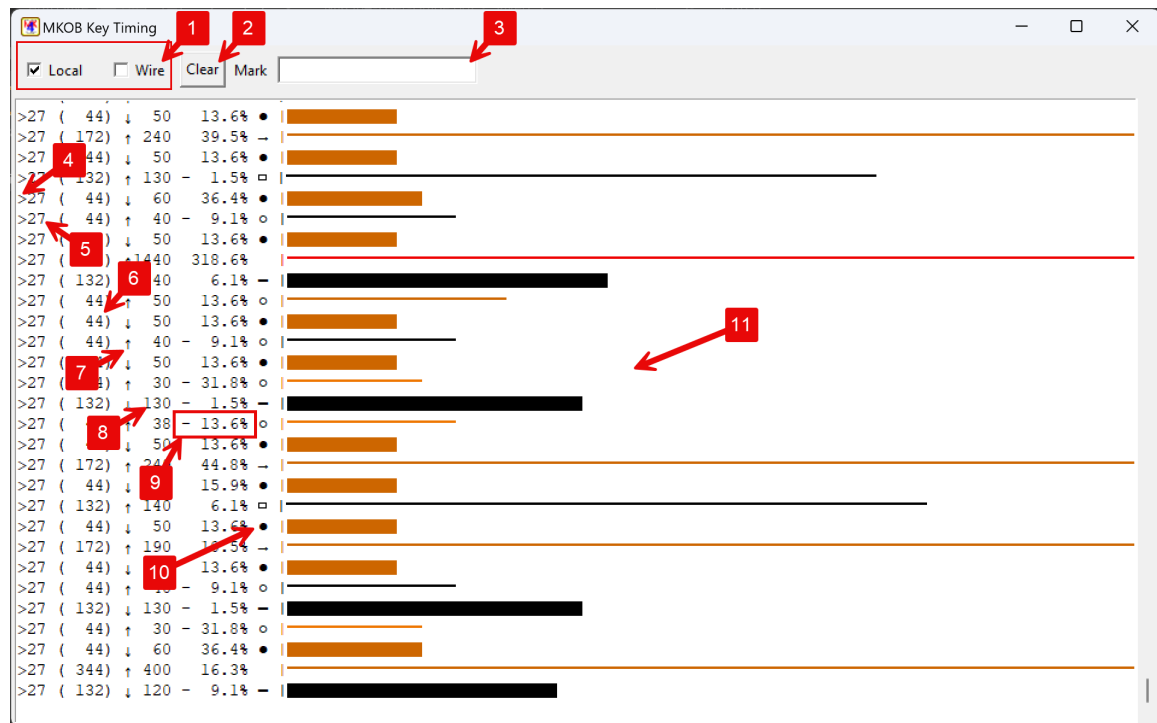

Figure 7. Key timing graph window

Explanation of the graph:

1. Select whether key timing from 'Local' and/or 'Wire' sources are graphed.

2. Clear the contents of the graph.

3. Enter text to mark the graph. A line of text is inserted into the graph by typing in the field and pressing the Enter/Return key.

4. Indicates the source of the key action:

    1. '>' Local

    2. '<' Wire

5. Speed (WPM) setting being used for comparisons/calculations.

6. Expected duration in milliseconds (1000 = 1 second)

7. Key action:

    1. ¬ Key down

    2. ¬ Key up

8. Measured duration in milliseconds (1000 = 1 second)

9. Error percentage compared to expected. A negative value indicates that the measured

duration was shorter than expected, a positive value indicates that it is longer.

10. Indication of what element this duration is closest to, given the Speed (WPM) setting:

    1. '¬' Dit (dot)

    2. '¬' Dah (dash)

    3. 'L' Long 'L' dash

    4. '0' Extra-long '0' dash

    5. '¬' Element space (the space between character elements)

    6. '¬' Intra-character space (space within the Morse characters: C, O, R, Y, Z, &)

    7. ' ' Space between characters

    8. '¬' Space between words

11. Graphical representation of the duration of the key action:

    1. Length is based on the key action duration.

    2. Key down actions use a wide bar.

    3. Key up actions use a thin line.

    4. The color is based on the duration error:

    5. Black: Less than ±10% ii. Lighter brown: -10% or more (negative error – too short) iii. Darker brown: +10% or more (positive error – too long) iv. Red: The value was too large to fit within the graph maximum

Local sources include a connected key, text entered in the Keyboard Sender window, and recordings being played using the File:Play… function. Wire source is code received from a connected wire.

It is important to understand that the graph is not decoding Morse. It is selecting the expected element based on the duration of the key action, given the configured code speed (WPM). Therefore, it is possible to get a sequence of elements that do not represent a valid Morse character. For example, you might get a sequence of: ¬¬¬¬ (dit, intra-char-space, dit, intra-char-space, dit - which isn't a valid character) if you were attempting to key an 'S' and your element spacing was slightly too long. You can monitor the Reader window in the main window to see what the Morse Reader/Decoder recognized your code as.

Recording and Playing Back Sessions

Record

You can start recording a session using the File:Record menu entry. That will start recording the current session using an automatically generated file name (based on the current date and time). The name of the file will be presented in a dialog box and displayed in the MKOB status bar. The recording will be stored in the folder containing MKOB.

Once the recording is started, all code activity (local and from the connected wire) will be recorded.

The recording can be stopped using the File:End Recording menu entry or when MKOB is closed.

Play Recording

A recording is loaded to play using the File:Play… menu entry. The menu entry opens a file selection dialog that is used to select a .pkrec (or a recording .json file). When a recording file is selected, MKOB will start reproducing the session. That includes listing the Offices/Stations that were involved in the session, reproducing the code (as it was sent), and cycling the Offices window through the active senders.

(Currently, only offices/stations that sent code are included in the recording. Including all offices, and displaying them as they join and drop, is planned as an enhancement.)

Playback

MKOB plays the recording at actual speed. The Play utility (part of the PyKOB package) allows adjusting the playback speed as a +/- percentage of actual speed.

MKOB compresses long periods of 'dead'/idle space (silence) down to five (5) seconds of re-produced idle time. The Play utility allows adjusting the period of idle time that is reproduced (allowing for exact reproduction of the recorded session).

Recording playback is treated like local keying, except that it will not send to a wire. If MKOB is currently connected to a wire, it will disconnect before playing a recording. Connecting is disabled while a recording is playing. This is done to avoid the recording being misconstrued by others as a live session. To provide a recording on a live wire, you can use the **Feed** utility (part of the PyKOB package). That allows you to provide a feed sourced from a recording, but it identifies the feed as the source rather than the individual offices that contributed to the recording. This is how Wire-134 is sourced from the recording of Instructograph Tape 5.

Playback Control

During playback, keyboard keys can be used to pause, move, and stop playback.

Stop Playback

While playing back – `Ctrl` + `S` will stop playback.

Pause/Resume Playback

`Ctrl` + `P` will pause current playback, and resume paused playback.

Move Back 15 Seconds

`Ctrl` + `H` will move back 15 seconds.

> ℹ️ MKOB does not put any indication of this movement into the Reader (Code Decoder) window.

Move Forward 15 Seconds

`Ctrl` + `L` will move forward 15 seconds.

> ℹ️ MKOB does not put any indication of this movement into the Reader (Code Decoder) window.

Move to Start of Current Sender

`Ctrl` + `J` will move the playback back to the beginning of the code sent by the current sender.

> ℹ️ MKOB does not put any indication of this movement into the Reader (Code Decoder) window.

Move to the End of the Current Sender

`Ctrl` + `K` will move the playback forward to the end of the code sent by the current sender.

> ℹ️ MKOB does not put any indication of this movement into the Reader (Code Decoder) window.

Moving to the Previous Sender

There isn't a control to move to the previous sender. However, the combination of: * `Ctrl` + `J`

– Move to the Start of the Current Sender * `Ctrl` + `H` – Move back 15 Seconds * `Ctrl` + `J` – Move to the Start of the Current Sender Will typically do it. This will move to the start of the current sender, then move back 15 seconds (which should be back into the previous sender), then move to the start of that sender. (File an enhancement issue if that doesn't work for you ¬)

Index

tone, 24

    International Code, 31-32

    system sound, 24

U

under-tick

    Code Reader, 16

Updating an Existing Configuration, 29

V

version 4.4.1, 24

W

Warn -2

    Level, 27

Colophon

This document reflects the features of MKOB Version 4.4.3 Edit date: 12-10-2024

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 MKOB 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 that became MKOB.