sBITX Toolbox Guide   v2.1

Table of Contents

Introduction

Welcome to the sBITX Toolbox Guide! This guide is here to introduce you to a set of applications created by JJ (W9JES), designed to improve your experience using the sBitx transceiver.

This guide is designed to help you from installation, using, updating, and removing the Toolbox. Each section is broken out into categories to quickly find what you need. You are encouraged to read the entire guide from the start if you are unfamiliar with the sBitx platform, Linux, and my applications.

Background

When I first used my sBitx, I noticed it was missing some features that I was used to on my other transceivers, such as direct frequency entry, memory bank access, and remote control options for mode, bandwidth, AGC, etc.

The sBitx, combining a single board computer (SBC) and a transceiver in one unit, is straightforward but also complex. I quickly learned that the sBitx software doesn't fully support standard third-party, remote rig control applications. I searched for an existing app to add these features but found none available for the sBitx.

This left me with limited options, such as developing web control interface or using telnet commands. As a result, I took on the project of developing my own application, **sBITX Manager**. I was pleased with how it turned out, and that inspired me to develop more apps and share them with others.

Prerequisites

- An internet connection configured and active on your sBitx transceiver
- USB keyboard, virtual keyboard, SSH, or VNC for initial installation
- USB keyboard, virtual keyboard, or VNC required for entering text in some applications.
- sBitx software version 2.1 or higher
- Python version 3 (if you are using a custom image or OS). The script will check and install Python if it is not present.
- Minimum OS: Raspberry Pi (Raspbian) version 10 "Buster"
  - Run **cat /etc/os-release** from terminal for verification

# Installation

1. Open the terminal application on your sBitx

2.
3. Type: **git clone https://github.com/drexjj/sBITX-toolbox.git**
4. Type: **cd sBITX-toolbox**
5. Type: **chmod +x *.sh**
6. Type: **./installer.sh**

7. Your screen should look like this when complete.

```
pi@sbitx:~ $ git clone https://github.com/drexjj/sBITX-toolbox.git
Cloning into 'sBITX-toolbox'...
remote: Enumerating objects: 154, done.
remote: Counting objects: 100% (141/141), done.
remote: Compressing objects: 100% (114/114), done.
remote: Total 154 (delta 53), reused 78 (delta 21), pack-reused 13
Receiving objects: 100% (154/154), 117.28 KiB | 1.45 MiB/s, done.
Resolving deltas: 100% (53/53), done.
pi@sbitx:~ $ cd sBITX-toolbox
pi@sbitx:~/sBITX-toolbox $ chmod +x *.sh
pi@sbitx:~/sBITX-toolbox $ ./installer.sh
sBITX-Launcher setup completed successfully! The applications have been added to
 the Pi Menu
pi@sbitx:~/sBITX-toolbox $ █
```

8. Wait for the installation to complete then close the terminal.
9. Verify the icons have been added to the desktop and Pi menu.
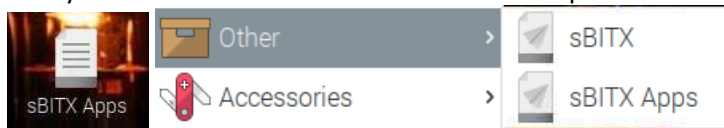
sBITX Toolbox
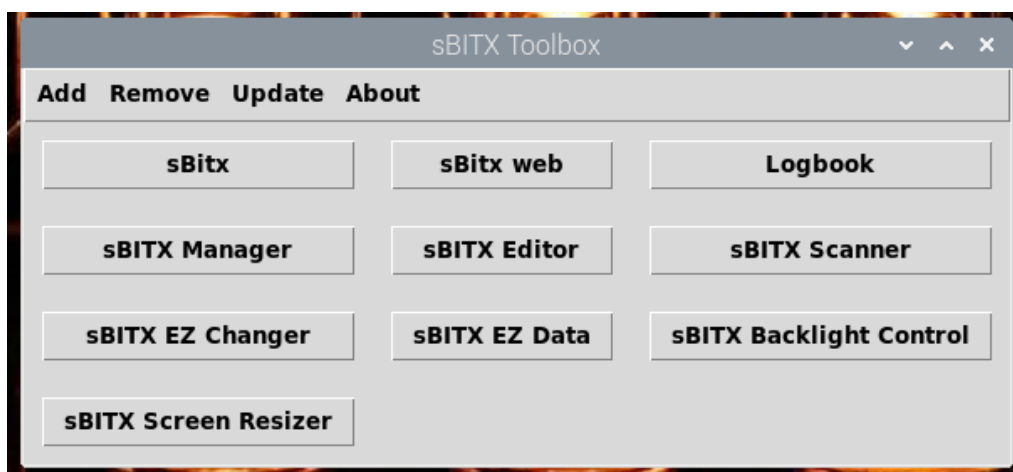
Using the sBITX Toolbox is fairly straightforward. The top menu contains commands and options to **add** or **remove** application buttons to the main area. Add or remove apps by selecting them from the dropdown menu. You can change the order with the add/remove buttons. Press a button to open the app of your choice.

You can also add custom buttons such as URLs, executable binaries or scripts. Click **Custom Application**

The format for a URL is **https://mywebsite.zzz** or **http://mywebsite.zzz/mypage.** For example, to add your QRZ profile, type **https://qrz.com/db/CALLSIGN**

To add an executable binary or script application like WSJTX, you can either try one of these options.

1. Click Run from the Pi menu and type the executable name to test. If it works, then add this same format in the **Add Custom Application** box as shown.
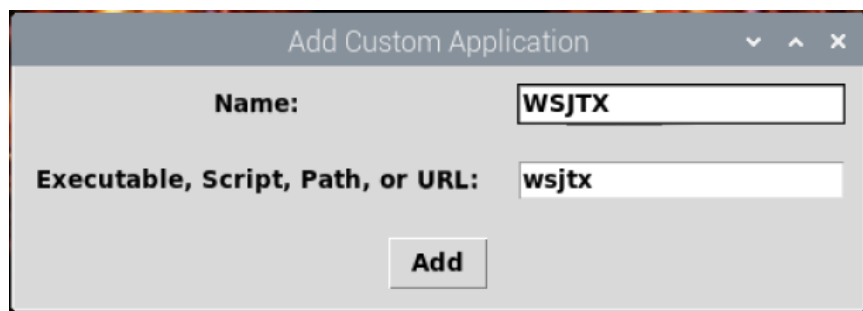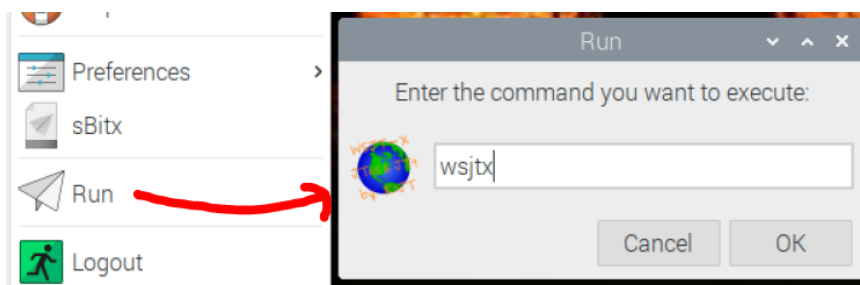
2. If you know the path of the executable or script, then you can enter it as shown.

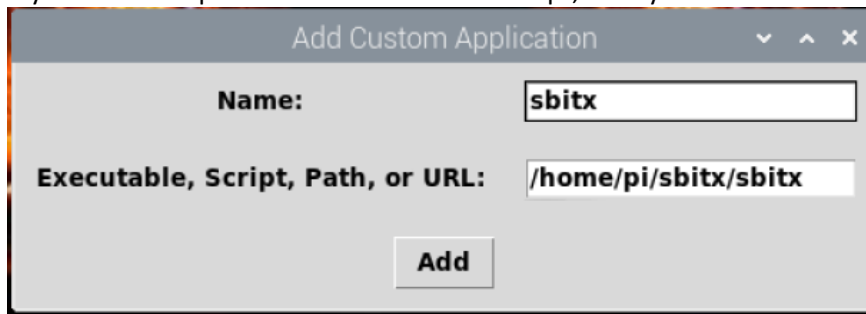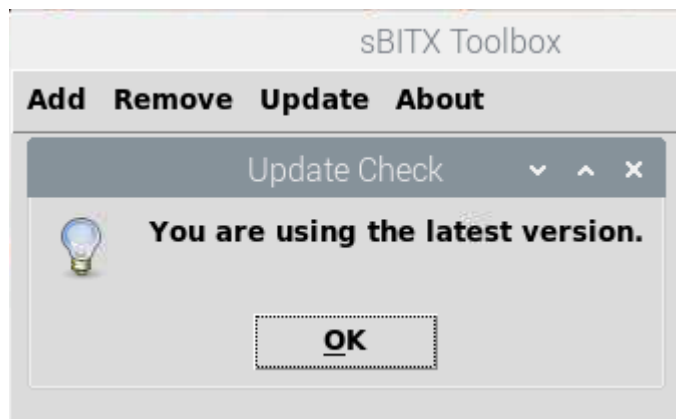Periodic updates to the sBITX Toolbox applications and features may be added over time. Press the **Update** button to check for updates and install them. It is best to visit https://github.com/drexjj/sBITX-toolbox/blob/main/release_notes.txt to see what has changed before updating your apps.

You can check the currently installed version from the **About** menu.

Applications

sBitx
A shortcut to open the sbitx application.

sBitx web
A shortcut to open the web version of sbitx.

Logbook
A shortcut to open the web-based, sBitx logbook.

sBitx Manager
An app that performs auxiliary, "rig control" of your sBitx.

sBitx Editor
A frequency editor application for sBITX Manager.

sBITX Scanner
An app that adds frequency scanning functionality to your sBitx

sBITX EZ Changer
An application to safely and effortlessly backup your entire sBitx software, restore it back to the original state, and upgrade/downgrade sBitx software while retaining your hardware settings.

sBITX EZ Data
An application for saving/restoring the hardware and software configuration data via USB drive.

sBITX Backlight Control
An application to conveniently control the screen brightness of the built-in display on your sBITX.

sBITX Screen Resizer
An application that allows you to easily change the screen size of your sBitx desktop equipped with the 7 inch Raspberry Pi display.

sBITX Time Sync
An application that syncs time with network time server, displays time zone offset, and shows current time for use with digital apps like WSJTX.
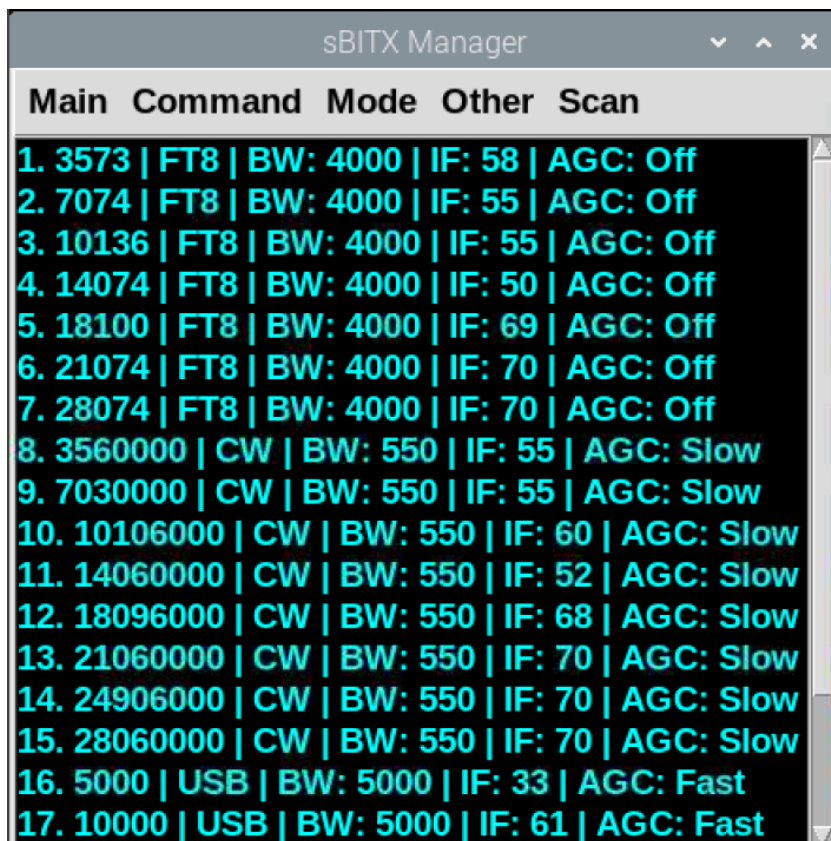
sBITX Macro Manager
An application deigned to edit, create, and manage macros on the sBITX transceiver.

sBitx Log Exporter
An application to select and download logs from the sBitx database and export them to an ADIF format to be used with other logging applications.

sBITX Manager Usage

Using sBITX Manager is much like using other rig control software even with the limitations of the sBitx transceiver. The key difference is that this application uses the telnet protocol to establish connection and send or receive information.

Features:

- Provides a user-friendly interface for interacting with the sBITX transceiver.
- Memory management allows for adding, editing, and removal of frequencies.
- Scanning function that scans through the list of stored frequencies at a customized interval.
- Send commands specifying details such as Frequency, VFO, Step, Mode, Bandwidth, IF, AGC, Audio, and more.
- Decodes messages in FT8 and CW modes.
- Visual adjustments such as text size and dark mode for enhanced readability.

The interface is an attempt at a "ham's don't read manuals" layout and closely resembles the user interface text of the sBITX. The menu bar contains most of the commands that require your input. You can select commands from the dropdown menu and either select or type a command. The sBitx will respond to the command you choose and set the desired value. The main window consists of the frequency memory bank where you can press to set the sBitx frequency.

The memory bank for this app is stored in a file called **sbmanager.json** in the /home/pi/sBITX-toolbox/data directory. A prepopulated frequency list is provided or you can start your own list by simply deleting the file and reopening sBITX Manager, then add a frequency to the memory bank.
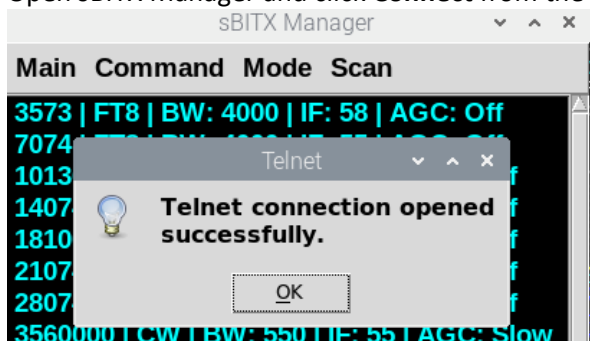
A word of caution: The sBitx transceiver can only allow one active telnet session at a time. That means you cannot use sBITX Manager and sBITX Scanner simultaneously.

Steps:

1. Ensure that the sBITX transceiver application is open and functional

2. Open sBITX Manager and click **Connect** from the **main** menu. You should see a popup message

3. If you receive an error and/or the commands and stored frequencies don't work, then close the sBitx app, close sBITX Manager. Start at step 1 and try again.
4. Enter the desired command or add/edit a frequency in the memory bank.
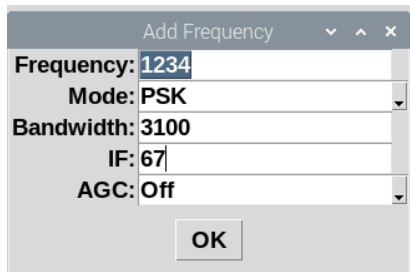
5. The scan function selects the first memory in the bank and scans "downward" in the memory bank list. Set the scan wait time to pause between memory bank changes.
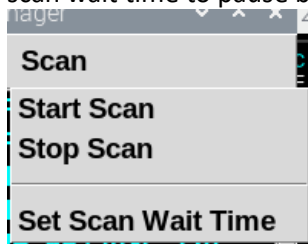
6. Editing or removing frequencies is easy. Select the frequency from the memory bank and go to the **Command menu → Frequency → edit/remove**. Make your changes and press OK .
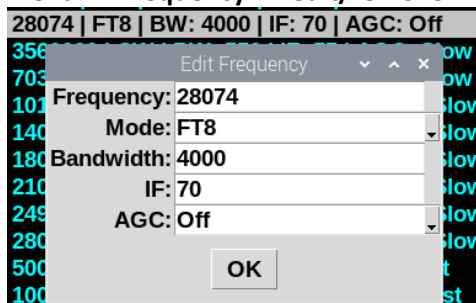
7. When you are done using the app, select **Disconnect** or **exit**. The app will close, however, the telnet session may still be "active" in the sBitx app (there is a bug on the sbitx side). I recommend you restart the sbitx app when finished with a "telnet" session.

A standalone version of this app is available for Windows 10/11 as well. https://github.com/drexjj/sBITX-toolbox/releases You can use this app on a computer that is connected to the same network as ypur sBitx to perform remote control functions. It is a great companion with the web version sBitx too.

sBITX Editor Usage
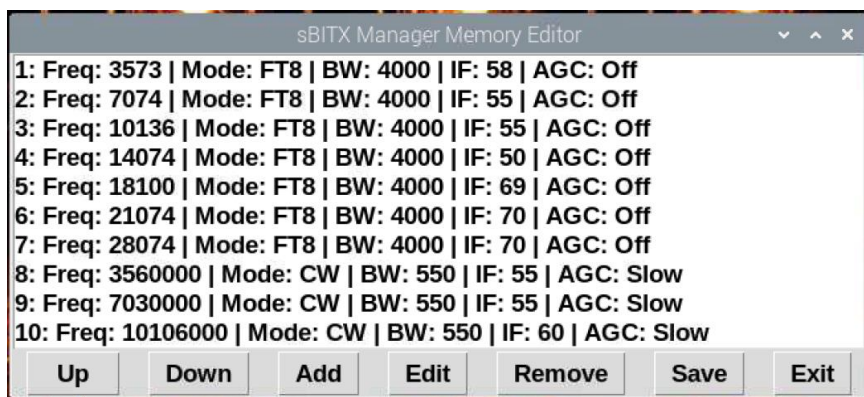
sBITX Editor is a companion to sBITX Manager and allows you to change the order of, add, edit, or remove frequencies in the memory bank. Select a frequency to change the order or edit the settings. Click **Save Order** when finished and restart sBITX Manager to see your changes.

sBITX Scanner Usage
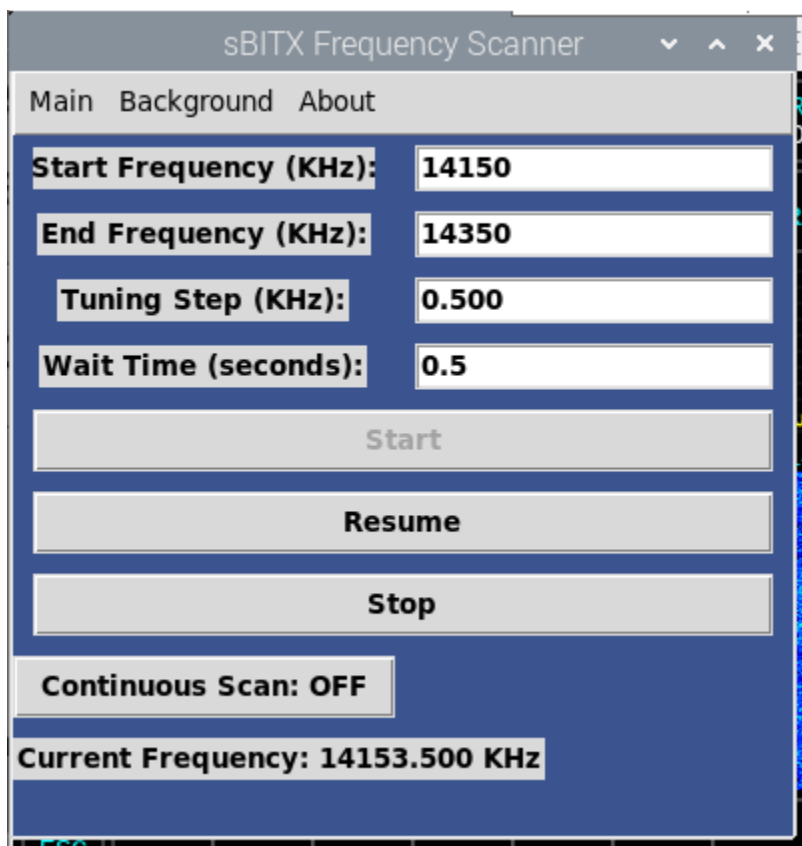
sBITX Scanner turns your sBitx into an automatic scanner to find open or active frequencies.

Features:

- Choose the frequency range, tuning step, and wait time
- Bidirectional scanning capabilities
- Continuous loop scanning
- Current frequency display in the app

A word of caution: The sBitx transceiver can only allow one active telnet session at a time. That means you cannot use sBITX Manager and sBITX Scanner simultaneously.

Steps:

1. Ensure that the sBITX application is open and functional
2. Open sBITX Scanner and click **Connect** from the **main** menu. You should see a popup message

3. If you receive an error and/or the commands and stored frequencies don't work, then close the sBitx transceiver app, close sBITX Scanner. Start at step 1 and try again.
4. Select a start and end frequency then select the tuning step and wait time between frequency changes.
5. Press start and watch the current frequency increment. You will hear the band activity through the speaker. If you find a frequency that you want to stop at, press pause or stop.
6. When you are done using the app, select **Disconnect** or **exit**. The app will close, however, the telnet session may still be "active" in the sbitx app (there is a bug on the sbitx side). I recommend you restart the sbitx app when finished with a "telnet" session.

A standalone version of this app is available for Windows 10/11 as well. https://github.com/drexjj/sBITX-toolbox/releases You can use this app on a computer that is connected to the same network as ypur sBitx to perform remote control functions. It is a great companion with the web version sBitx too.

sBITX EZ Changer Usage

sBITX EZ Changer allows you to safely and effortlessly backup your entire sBitx application, restore it back to the original state, and upgrade/downgrade versions while retaining your hardware settings.

You can install new versions of the sBitx software or upgrade/downgrade using the simple buttons. Your hardware calibration will not be affected so you don't have to worry about recalibrating your transceiver. Additional buttons will be added when new versions are published on https://github.com/afarhan/sbitx

Backups are stored on your sBitx in the **/home/pi/sbitx_backup** folder and the hidden, **/home/pi/.sbitx_backup** folder.

Note: After pressing a version button, please wait until the popup is displayed. it may take some time to complete depending on your internet connection and the script actions to reconfigure the sBitx folder. The software versions are based on the commit history of Farhan's repository.

Steps:

1. Ensure that the sBitx app is closed
2. Open sBITX EZ Changer and select **Backup**.
3. Select a version of sBitx you wish to install.
4. Select **Restore** to bring back your original configuration and files.
5. Open sBitx (If you are upgrading from v3.0 or below, then the sBitx app will need to take about a minute to create the "wisdom" files needed to calibrate the FFT. Please be patient!

sBITX EZ Data Utility Usage

This app allows you to save, restore, and export backups of your calibrated hardware files located in **/home/pi/sbitx/data**.

Steps:

1. Insert a USB drive into an open slot of your sBitx, then open **sBITX EZdata**.
2. Click **Refresh** and select the appropriate drive from the list.
3. Click "**Backup Files To The Selected USB Drive**".
4. A new folder will be created on the USB drive called **sBitx_Data_backup**.
5. Inside that folder will be your backup with the date and time of the backup.
6. You can select a date to restore the data folder back to the sBitx transceiver.

sBITX Backlight Control Usage
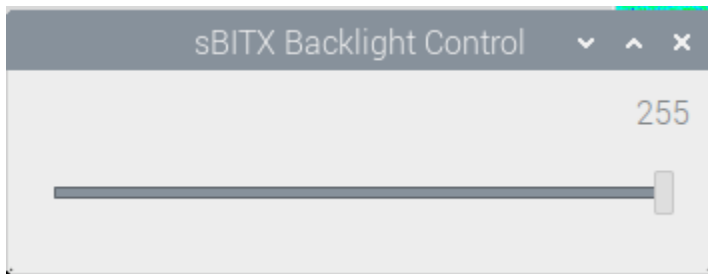
A convenient way to control the backlight level (brightness) of the sBitx display for eye comfort and reduced power draw.

A touchscreen friendly slider control allows you to change the brightness.

The backlight level setting is stored on the system and will remain enabled after a reboot.

The minimum brightness level is set to 15 which is barely readable in low light settings to consume less power and extend life of the display.

Simply drag the slider control to your preferences.

sBITX Screen Resizer Usage
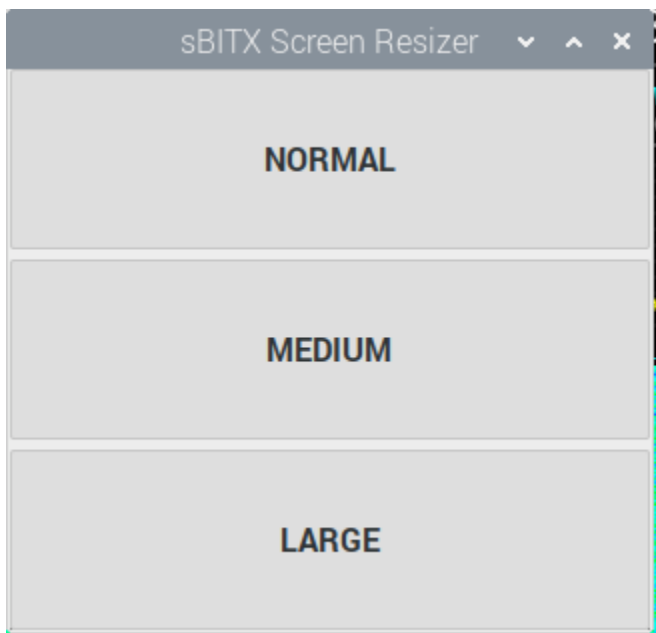
An application that allows you to easily change the screen size of your sBitx desktop equipped with the 7 inch Raspberry Pi display.

The app has three settings which change the resolution as follows:

- Normal - The default, sBitx resolution of 800 X 400
- Medium - 1024 x 768 resolution for running apps like WSJT-X with screen panning enabled.
- Large - 1820 x 1024 resolution for running multiple apps via VNC with screen panning enabled.

The transceiver will default back to the normal screen size on reboot.

sBITX Time Sync Usage

sBITX Time Sync is a graphical tool that can be used to verify and update your local time calibration for use with digital modes which rely on an accurate time base. This app works appropriately when there is a routable internet connection.
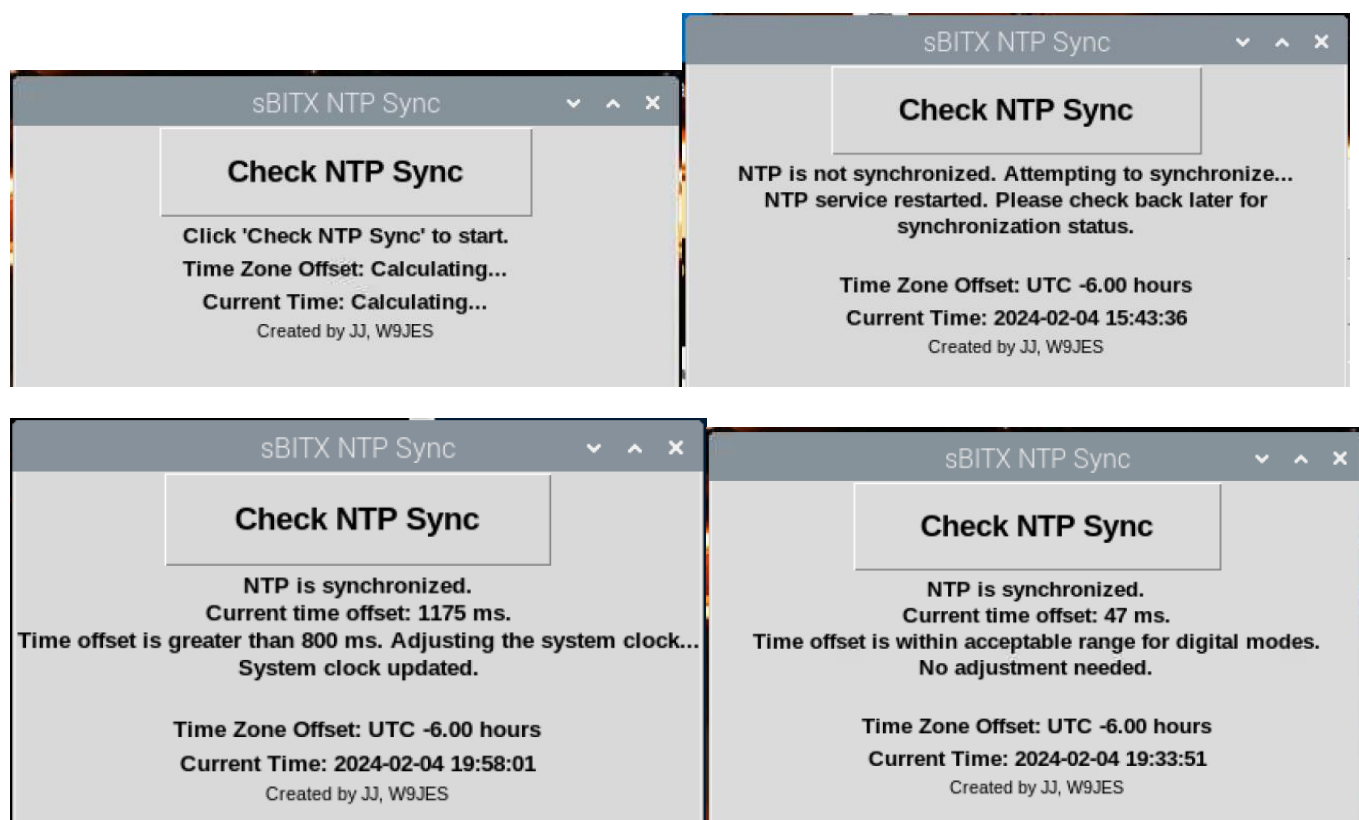
It is highly recommended to connect the sBITX to your home WIFI or smartphone hotspot to maintain an accurate time source when digital modes are in use.

Simply press the **Check NTP Sync** button to check the time variance and correct it. If you receive a "not synchronized message repeatedly, then check your internet connection or restart the sBitx transceiver and try again.

This app will perform a time sync of the local time and server time are different by 800 milliseconds or more.

sBITX Macro Manager
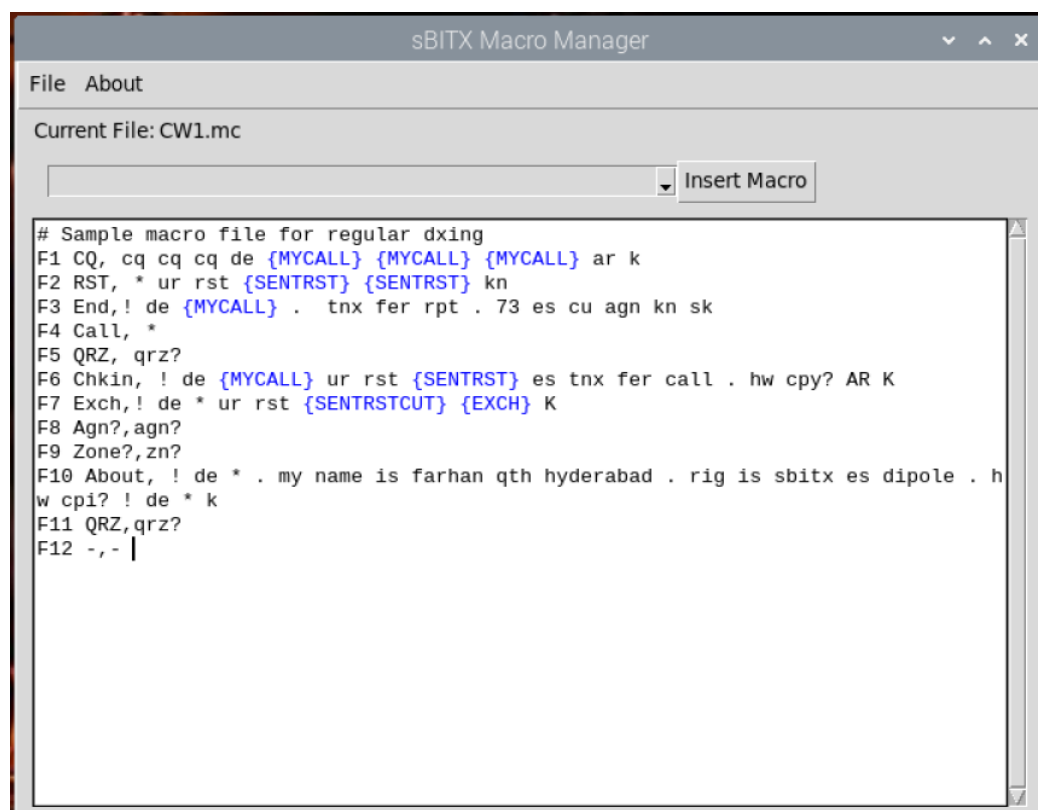
Creating a New Macro File:
1. Click on "**File**" in the menu bar and select "**New**". This will open a template in the text area where you can start adding your macros.
2. Use the provided template and modify its contents.

Opening an Existing Macro File:
1. Click on "**File**" in the menu bar and select "**Open**".
2. The default location where sBitx macro files are stored is displayed, select the desired file, and click "**Open**". The contents of the file will be loaded into the text area for editing.

Saving Macro Files:
- **Save:** To save your changes to the currently opened file, click "**File**" → "**Save**".
- **Save As:** To save your macro file with a new name or location, click "File"→"Save As...", enter the file name, and click "**Save**".

Inserting Macros:
1. In the macro frame, use the dropdown menu to select the macro you want to insert into your file. The dropdown shows both the macro code and a brief description for each.
2. Click the "**Insert Macro**" button. The selected macro will be inserted into the text area at the current cursor position.

Editing Macros:
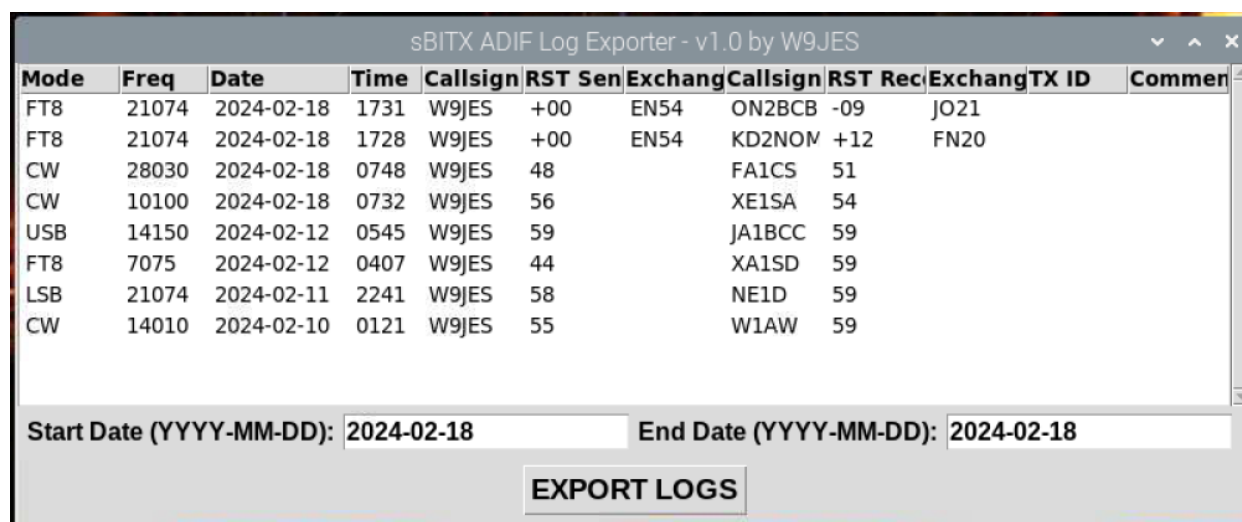- Type directly into the text area to add or modify the text and macros in your file.

Highlighting Macros:
- As you type or after inserting macros, the application will automatically highlight valid macro codes in blue for easy identification.

Tips for Effective Use
- Familiarize yourself with the available macros and their purposes to make the most out of your macro files.
- Test the macros on low power into a dummy load or on an unused frequency to ensure desired functionality.
- You can load the macro into your sBitx by using sBITX Manager or by typing \macro "macro name" in the text field on the sBitx application.
- Use the keyboard from sBITX Toolbox or connect and external keyboard to enter text.

sBITX Log Exporter

Mode	Freq	Date	Time	Callsign	RST Sen	Exchang	Callsign	RST Rec	Exchang	TX ID	Commen
FT8	21074	2024-02-18	1731	W9JES	+00	EN54	ON2BCB	-09	JO21		
FT8	21074	2024-02-18	1728	W9JES	+00	EN54	KD2NOM	+12	FN20		
CW	28030	2024-02-18	0748	W9JES	48		FA1CS	51			
CW	10100	2024-02-18	0732	W9JES	56		XE1SA	54			
USB	14150	2024-02-12	0545	W9JES	59		JA1BCC	59			
FT8	7075	2024-02-12	0407	W9JES	44		XA1SD	59			
LSB	21074	2024-02-11	2241	W9JES	58		NE1D	59			
CW	14010	2024-02-10	0121	W9JES	55		W1AW	59			

Start Date (YYYY-MM-DD): 2024-02-18   End Date (YYYY-MM-DD): 2024-02-18

EXPORT LOGS

sBITX Log Exporter is an application that allows you to select and export logs from the built-in sBitx log database to use them in other logging applications. The file format is standard ADI and can be imported into various online and software based loggers.

You will need to use the onscreen keyboard from the toolbox or an external keyboard to enter the date.

Select the date range you would like to export to the file and click **EXPORT LOGS**. A popup will display the filename and the number of entries added to the file.

The logs are stored in **/home/pi/sBITX-toolbox/data** and written with the date and time of export. You can use the web browser, USB storage drive, SSH, or other means to transfer the adi file to your logger.
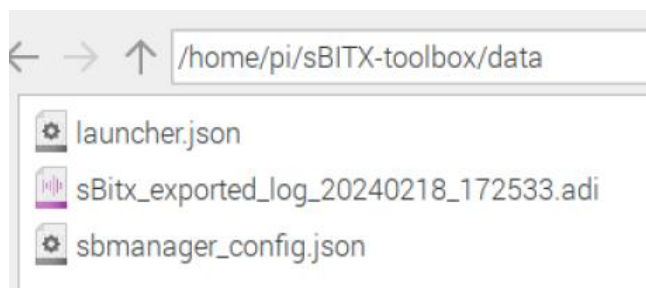
Optional Functionality Improvements

Many versions of sBitx come with a static entry in the hidden profile to set the time zone to Arizona time. This will affect the system clock and make it confusing for the operator.

It was confirmed by A. Farhan that RTTY and PSK31 modes are not functional. These modes can be removed from the sBitX app without loss of functionality.

I've created a script to choose these optional fixes. Each option is preceded by a Y or N. The sbitx_gtk.c file will be backed up before making changes.

Directions:

1. Close the sBITX application if it is open

2. Open the **terminal** application on your sBitx
3. Type **/home/pi/sBITX-toolbox/scripts/sbitx_fix.sh** select options and wait for the script to complete
4. Restart the sBITX transceiver

I repaired a file that is part of the sBitx software package called **remote.c**. This file handles the telnet connection interface to the sBitx application. As mentioned above, there is a bug in the code that causes an overflow or instability in this module which makes reopening the telnet connections impossible. My version has been repaired so you can use it with **sBITX Manager** and **sBITX Scanner** without restarting the sBitx app on the transceiver.

Detailed information on these changes is available in /home/pi/sBITX-toolbox-local/scripts/sbitx/telnet_fix_readme.txt I also commented the code changes to ensure transparency if you choose to examine the file in a text editor.

If you want to use this module, then follow these steps:

1. Close the sBITX application if it is open

2. Open the **terminal** application on your sBitx
3. Type **/home/pi/sBITX-toolbox/scripts/telnet_fix.py** and wait for the script to complete
4. Restart the sBITX application

Advanced Troubleshooting

If you encounter errors performing the update or sBITX Toolbox becomes unstable, then try these steps.

Option A (Force an update):

1. Open the **terminal** application on your sBitx
2. Type: **mousepad /home/pi/sBITX-toolbox/VERSION.txt** and press enter.
3. Change the version to 1.0 then press **File→Save** and close the program.
4. Close the Terminal.
5. Open **sBITX Launcher** and try the update again.

Option B (Force a reinstall of the local application):

1. Open the **terminal** application on your sBitx
2. Type: **/home/pi/sBITX-toolbox/installer.sh** and wait for the script to complete
3. Close **Terminal**.
4. Open **sBITX Launcher** and try the apps or run an update.

Option C (Reinstall the sBITX Toolbox application):

Word of Caution: This option will remove your launcher configuration and custom frequencies in sBITX Manager!
Perform a backup of the files **launcher.json** and **sbmanager_config.json (optional)** before using this option.

1. Open the **terminal** application on your sBitx
2. Type: **/home/pi/sBITX-toolbox/uninstaller.sh** and wait for the script to complete
3. Type: **cd /home/pi**
4. Type: **git clone https://github.com/drexjj/sBITX-toolbox.git**
5. Type: **cd sBITX-toolbox**
6. Type: **chmod +x *.sh**
7. Type: **./installer.sh** and wait for the script to complete
8. (Optional) Restore the launcher.json and sbmanager_config.json files
9. Run **sBITX Launcher**

Option D (Remove any version of sBITX Toolbox manually)

1. Open the **terminal** application on your sBitx
2. Type: **cd $HOME**
3. Type: **sudo rm -rf /home/pi/sBITX-toolbox**
4. Type: **sudo rm /usr/share/applications/sBITX_launcher.desktop**
5. Type: **sudo rm /home/pi/Desktop/sBITX_launcher.desktop**
6. For version 2.0 and above only – continue to step 7 and 8
7. Type: **sudo rm /usr/share/applications/sb_launcher.desktop**
8. Type: **sudo rm /home/pi/Desktop/sb_launcher.desktop**