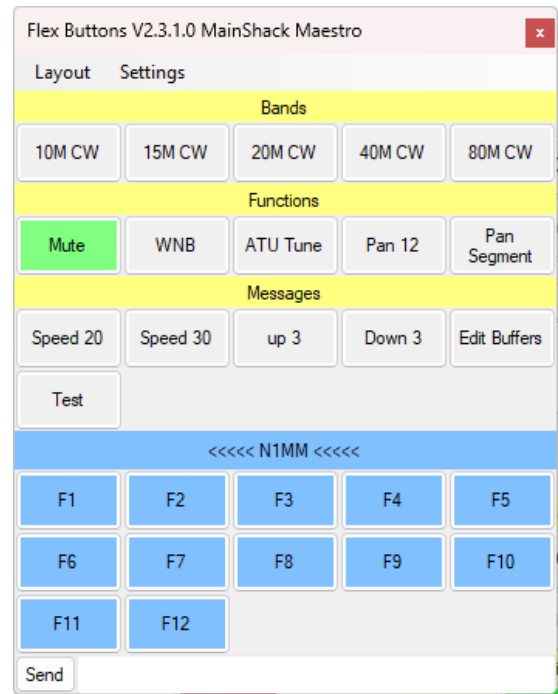


Flex Buttons Utility (V2.4)

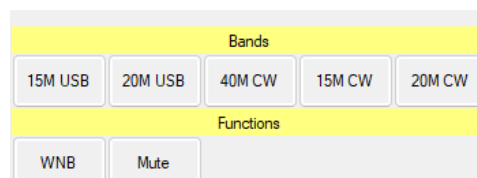
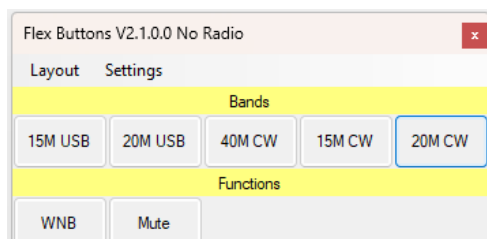
Keith Regli K7KAR
keith@kregli.com

Simplicity of operation and screen real estate are things I care about when I'm operating. This little utility simplifies my screen layout and reduces the number of clicks I need to accomplish some common functions. I no longer have PGXL and TGXL utilities running all the time and don't need to run SmartSDR for mouse access to functions. The pictures below show different layouts that I set up to demonstrate possibilities. You can have as many or as few of these buttons as you like and you can arrange them as you like. **Note:** Start your radios, PGXL and TGXL (and N1MM) *before* starting FlexButtons.



The two on the left have the same functions, but show how gaps can make things easier visually. The one on the right shows more of a vertical orientation, N1MM buttons and adds a CW messages box. Others set up all bands/all modes or buttons for each of the nets they frequent. Still others set up buttons to replace profiles. It all depends on how you use your Flex radio.

Right clicking on the menu bar shrinks the FlexButtons window to just the buttons and a 10 pixel top border. Right clicking on that top border returns to the normal window. Note: to make changes or close the window it must be normal size.



ONE CLICK!

You can combine two or more functions on one button, such as: on slice A go to 14.329Mhz, USB with a 2700 Hz filter, the panadapter zoomed to 10 kHz, set POWER to 25 W, BYPASS the ATU, set the TGXL to TUNE and the PGXL to OPERATE. And now you're ready for the Flex Net.

Due to user interest FlexButtons has grown from just a few functions to buttons for nearly anything you'd want to adjust (and many things you probably have no desire to adjust). The picture below is my attempt to add just one variation of each button type. The down side of this is that testing all possible combinations of things is impossible for me. While I try to test as most features in several configurations, I'm bound to leave in a few bugs that you will encounter – sorry.



With version 2 the actions are organized into loosely defined groups to keep the selections more manageable. I haven't implemented every possible setting, but we may get there :) There were a couple of settings that I wanted to add, but couldn't get them to work. Below is the selection tree:

AF – audio level actions associated with a slice

- AFGain: Set the AF Gain level (0-100) or set an adjustment (-10,+10)
- APF: Audio Peaking Filter on/off/toggle or set a level.
- AudioPan: Set the audio L/R balance (0-100)
- Mute: Set the mute state on/off/toggle
- Mode: Set the operating mode
- Play: Play the record buffer.
- Record: Start recording audio for this slice
- RxDax: Set the DAX channel
- RxFilter: Set the filter to a width in Hz
- RxFilterBounds: Set the upper and lower bounds for filter in Hz. Use this with care.

Cosmetic – non-action layout features

- Blank: Insert a blank space for a button
- Gap: Insert a blank row for a title
- Layout: Switch to a different layout

CW – actions associated with CW operational

- Breakin: Set on/off/toggle for breakin.
- CWDelay: CW Delay 0-2000ms
- SendBuffer: Send a pre-programmed buffer CWX buffer .
- EditBuffers: Edit the CWX buffers.
- SendText: Send a locally pre-programmed buffer. (Note: right click on any button cancels send)
- SendRtty: Send a RTTY signal through DAX with the given text.
- SidePan: Adjust the sidetone L/R balance (0-100).
- Sidetone: Set the sidetone on/off/toggle.
- SideVol: Set the volume of the sidetone (0-100).
- Speed: Set the CW speed for the internal keyer (5-100 wpm).
- SpeedDown: Decrease the CW speed (2-10 wpm).
- SpeedUp: Increase the CW speed (2-10 wpm).
- N1MM: Add an N1MM function key.

External – actions associated with things outside the radio

- ANT: set the RX, TX or RX/TX antenna
- PGXL: Switch the active PGXL to operate or standby
- RUN: Run a program with arguments
- SCAN: Set timer and buttons to cycle through
- TGXL; Switch the active TGXL to operate or bypass

Frequency – actions that set the frequency of the radio

- BandMode: Set a frequency (MHz) and mode
- BandModeFilter: Set a frequency (MHz) mode and filter width (kHz).
- Frequency: Set the frequency in MHz

Panadapter – actions associated with the panadapter displayed

- Average: Set the level of averaging to smooth the panadapter display.
- Bandwidth: Set the panadapter zoom to Band, Segment or a specific width in kHz
- BlackAuto: Set on/off/toggle or threshold for removing weak signals from the display.
- DAXIQ: Set the DAXIQ channel for this panadapter (1-8).
- FallColor: Set the color scheme for the waterfall display (Basic, Add Purple, Dark, Grayscale, Deuteranopia, Trianopia).
- FallGain: Set intensity of the colors in the waterfall display (0-100).
- FPS: Set the frequency that the display is updated (0-30).
- Rate: Set the rate at which the waterfall falls (0-100).
- Weight: Set on/off/toggle for the waterfall to use the weighted average to emphasize certain signals.

RF-Slice – RF related actions that are attached to a slice

- AGCMode: Automatic Gain Control speed may be set to Fast, Medium, Slow, None, Off
- AGCThreshold: Set the maximum gain available to the AGC function (0-100)
- ANF: Automatic Notch Filter on/off/toggle or set a level{0-100}.
- DialStep: Set the dial step rate in Hz (1-1000).
- DIV: Diversity on/off/toggle
- RFGain: -8, 0, 8,16,24,32dB
- RIT: Set the offset in Hz for the RIT (0-10 Hz).
- NB: Noise Blanker on/off/toggle or set a level {0-100}.
- NR: Noise Reduction on/off/toggle or set a level{0-100}.
- WNB: Wideband Noise Blanker on/off/toggle or set a level{0-100}.
- XIT: Set the offset in Hz for the XIT (0-10 Hz).

RF-No Slice – RF related actions that are independent of a slice

- ATU: Invoke Tune or Bypass
- ClearSpots: Clear all spots
- FDX: Full duplex on/off/toggle
- MemoryRecall: Select a memory profile to use
- MOX: Turn MOX on/off/toggle
- Power: Set the transmit power level (0-100).
- TNF: All Tracking Notch Filters on/off/toggle.
- Tune: Radio TUNE on/off/toggle
- TunePower: Set the tune power level for this band (0-100)
- TXFIL: Set the bounds for the TX Filter
- VOX: VOX on/off/toggle or set a threshold.

Slice Operations – actions that manipulate whole slices

- COPY slice X to slice Y
- SLICE: Set the active transmit slice
- SWAP slice X with slice Y
- ToggleSplit add slice B up 1kHz or remove slice B
- CloseSlice Closes the selected slice

The functions that turn things on and off generally allow you to “toggle” as well. In addition, those buttons that have on/off states and have a color attached will have the background changed to the color when the device is ON and standard gray when the device is OFF. Other buttons with colors attached will always show the background in that color. Clicking on a “Gap” label will either hide or reveal the buttons between it and the next “Gap”.

Installation

If you have a version 1.x of FlexButtons installed the installation will *delete* any previously created layouts.

You can download this documentation and the software from my [website](http://www.kregli.com) (<http://www.kregli.com>). It's free – if you feel the need to pay for it, donate to some good charity.

Once you get the zip file (FlexButtons.zip) create a directory in any convenient location (e.g. C:\Programs\FlexButtons). Then extract the files from FlexButtons.zip to that directory (right click FlexButtons.zip and select Extract All...). Depending on the privileges you have, you could run into your security system. You'll have to work your way around that. Choosing a directory in your documents folder may make that easier.

You'll see FlexButtons.exe, FlexButtons.pdf (this document), FlexButtonsRepair.exe and a bunch of auxiliary files.

Double click on FlexButtons.exe and you should see FlexButtons appear in the upper left of your screen with the default set of buttons (be patient, it takes a few seconds to search for radios).



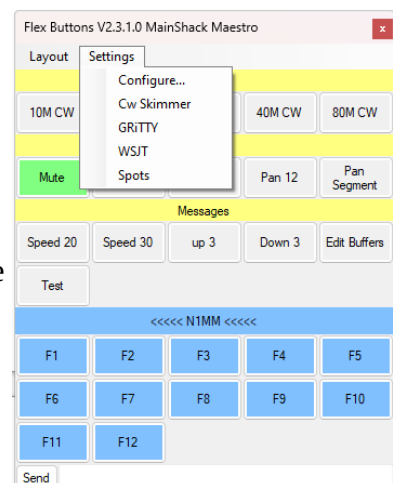
If you see the screen to the left, it just means that I didn't go to the trouble of getting a Windows signature on the file. Clicking "More Info" and "Run Anyway" will get you by this screen. It should only appear the first time you run a new version.

The information about the buttons is stored in the Windows Registry and the program will setup the necessary keys when it runs the first time (for those who know about the Registry it's [HKEY_CURRENT_USER\Software\K7KAR\FlexButtons]). If your account is short on privileges, you may have to use "Run as Administrator..." the first time you run the program. See the description of FlexButtonsRepair.exe towards the end of this document.

If your radio is on and connected to the same LAN as your computer the buttons should work. With this base you can change the configuration to meet your needs.

Configuration

Once the basic program is working you'll probably want to start on your own layout. You do this by clicking on Settings at the top of the window and selecting "Configure...". That will bring up the settings window with the current layout. The only way to get comfortable with configuring layouts is to try it. Expect to throw away your first few efforts.



The top row has the parameters that apply to the whole layout. The button height and width, the number of columns and the number of CWX Pads. The CWX Pads are free form text buffers that you can use to send interactive CW (or RTTY).

Below those you can select which layout you'd like to work on. You can click New, Delete, Import or Export the selected layout and you can Import a layout in an appropriately formatted ".dat" file. There should be some layouts you can import in your installation directory. For a "New" layout a dialog box will ask for the new name and when you click "OK" a copy of the current layout is made with the new name. You can then add or remove buttons from that layout.

To the right of the window are check boxes labeled "Top Most", "Skip TGXL Discovery" and Sidetone. Top Most allows Flex Buttons to appear on top of all other windows on your desktop. Skip TGXL Discovery can save several seconds on startup if you don't have a TGXL. The Sidetone and pitch will be generated when a CW/Send Text command is issued.

The "Use N1MM" checkbox causes FlexButtons to look for N1MM at startup and add the buttons for N1MM at the bottom of the layout. You can't edit those buttons. See the section on N1MM towards the end of this document. Spots allows you to funnel DX Cluster spots to the radio.

The bottom portion of the window is devoted to a representation of the current layout.

Layout

Button Height: 40, Button Width: 70, Columns: 5, CWX Pads: 1

☐ Top Most, ☒ Skip TGXL Discovery, ☒ Use N1MM

Sidetone: 500

Layout: Field Day CW, New, Delete, Export, Import

You can add buttons to the layout by entering the text and then clicking "Make Button". You can duplicate, delete or edit a button by right clicking on it.

dxc.ve7cc.net:23, Spots

Bands

10M CW	15M CW	20M CW	40M CW	80M CW
--------	--------	--------	--------	--------

Functions

Mute	WNB	ATU Tune	Pan 12	Pan Segment
------	-----	----------	--------	-------------

Messages

Speed 20	Speed 30	up 3	Down 3	Edit Buffers
----------	----------	------	--------	--------------

Test

OK, Cancel

You'll notice in the picture above that some buttons are alone on a row. Those are "Gap" buttons and can be used to separate groups of buttons in the display.

If you drag a button (just the cursor will move) you can place it where you'd like it to display. This takes a little practice. I find if you grab the upper left corner of the button and drag to the right-middle

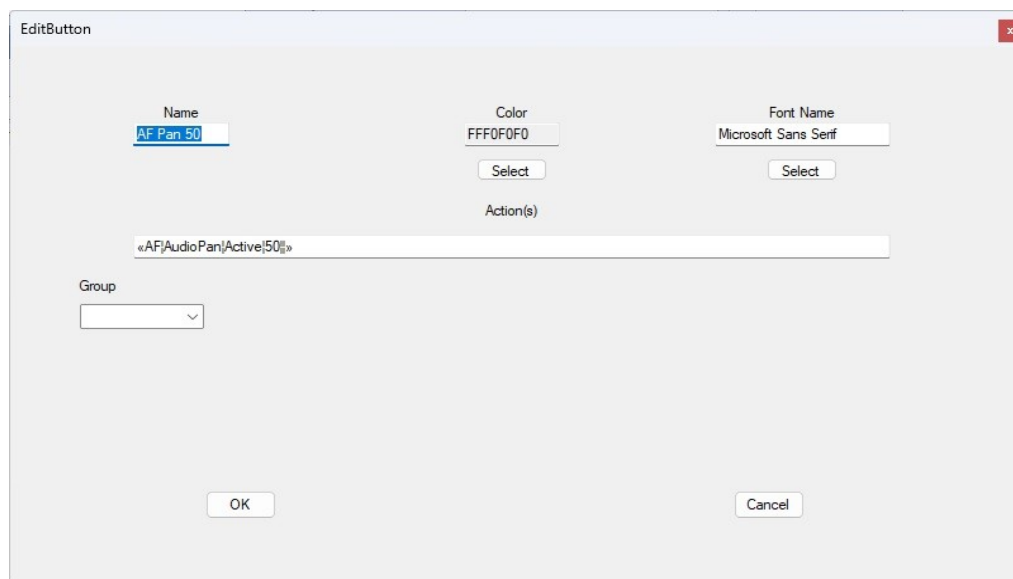
of the button you want it next to, then release it almost always works (the algorithm is a little complicated).

If you right click a button a menu will pop-up that will allow you to Edit, Delete or Duplicate the button. Dupe creates a button with the same characteristics as the existing button. Delete removes the button completely. Edit allows you to adjust the characteristics of the button.



You can also create a button with no characteristics by typing a name into the text box next to the Make Button button.

When you click on the Edit Button menu the following window appears:



Across the top you can change the name, color and font for the button. The second line shows the current action{s} for the button. Once you get the hang of it you can edit this directly (say for a Dupe button). At first you'll want to use combo boxes below the Action text box. As you build a new action you can add it on the end of the existing actions. If you want a whole new action just delete the contents of the Actions text box.

When you pull down the Group combo box you'll see the groups of actions listed. When you pick one a second combo box will appear and you can select the action from that. Then, depending on the action, other combo or text boxes will appear allowing you to add the parameters for the action. **Note: use the TAB key to indicate your entry is complete in a text box.** Once you've added all the parameters the "Add to Action" button will appear. As each new parameter entry box appears some text explaining what to do will appear.

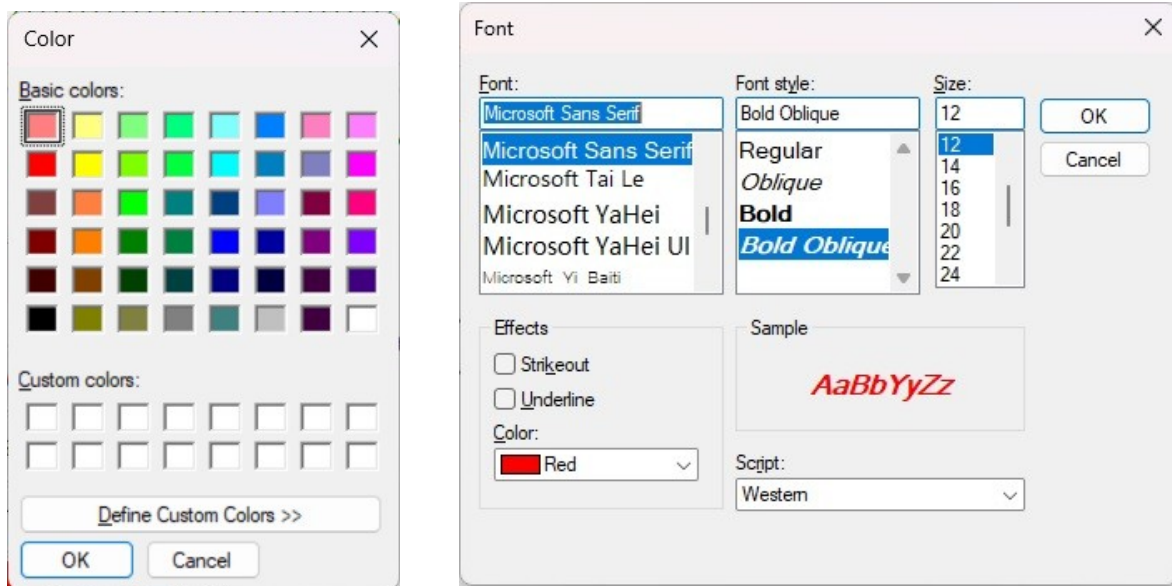
The screenshot shows the 'EditButton' dialog box. It has a title bar with a close button. The main area contains several input fields and buttons. At the top, there are three columns: 'Name' with a text box containing 'NR 50', 'Color' with a text box containing 'FFFFFF00' and a 'Select' button below it, and 'Font Name' with a text box containing 'Microsoft Sans Serif' and a 'Select' button below it. Below these is an 'Action(s)' text box which is currently empty. Underneath the 'Action(s)' box are five controls: 'Group' with a dropdown menu showing 'Frequency', 'Select' with a dropdown menu showing 'BandMode', 'Slice' with a dropdown menu showing 'Active', 'Range 0-13000' with a text box containing '14.02', and another 'Select' with a dropdown menu showing 'CW'. In the center of the dialog is a large orange button labeled 'Add to Actions'. At the bottom are two buttons: 'OK' on the left and 'Cancel' on the right.

NOTE: If you click the OK button before you add the action you'll have to do it all over. Once you have clicked the Add to Actions button, you can add another action to the button or save the characteristics by clicking OK.

One special case. The Scan action. This action allows you to set a macro going that automatically clicks a sequence of buttons spread out over a time interval. You might use this switch to various net frequencies every few seconds. In this case the program allows you to repeatedly select the buttons to press until you complete entry of the interval. The button names are shown in the action text box as you add them. Once you add the time interval the “Add to Actions” button will appear. Clicking it will create the action. Note: you can't add multiple actions with a a Run action.

This screenshot shows the 'EditButton' dialog box with the 'Scan' action configured. The 'Name' field still contains 'NR 50'. The 'Color' and 'Font Name' fields are the same. The 'Action(s)' text box now contains the text '14.030/CW, Bandwidth 20,'. Below this, the controls have changed: 'Group' is a dropdown showing 'Extern', 'Select' is a dropdown showing 'Scan', 'Next Button' is a dropdown showing 'Bandwidth 2', and 'Interval (sec)' is a text box containing '10'. The orange 'Add to Actions' button is still present in the center. The 'OK' button at the bottom left is now highlighted with a blue border, while the 'Cancel' button remains unchanged.

If you click the Select button below the Color or Font fields you will see the standard Windows dialogs for picking colors and fonts (the gray background is RGB(240,240,240)).



Some of the actions could use a little explanation. For the most part you can find explanations of what the settings do in the Flex SmartSDR documentation (or 6x00M User Guide).

CW Speed – There appears to be an anomaly in the FlexLib that returns a speed of 30 wpm regardless of the actual keyer setting. So, I keep the speed internally. It starts at 25wpm, but if you use a knob on an M model or the slider in SmartSDR I won't see the change and the next button you click will go off my internal speed rather than the Flex speed.

CW Send Text – This sends the text directly. If N1MM is in use FB will substitute the CALL, RST and Serial Number fields for C%, R% and S% in the message.

RTTY Send Text = This sends the text directly through the current DAX selection. N1MM substitutions are the same as for CW Send Text.

CW Send Buffer – This only works if SmartSDR is running and the CWX buffer is set up.

CWX Pads – These work like the CW Send Text, but you can enter the text any time. It will be deleted after it is sent. This will send RTTY if your radio is in RTTY mode.

Copy Slice – This copies all of the slice parameters from the source slice to the target slice so they end up looking identical.

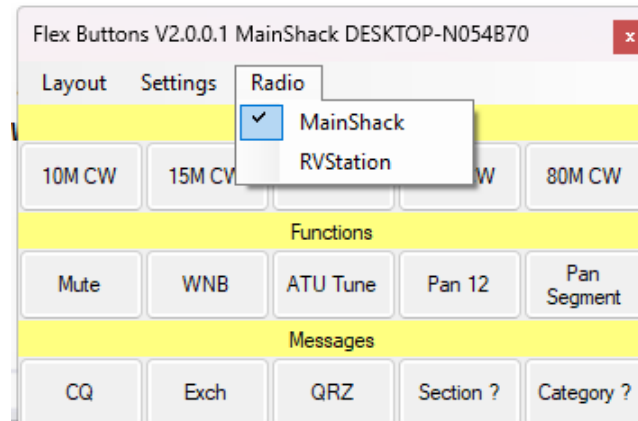
Swap Slice – This trades all of the slice parameters between the two slices.

Memory Recall – When you use saved memories (for say repeaters using a transverter), this will let you recall one into the active slice. You will probably want to use the ANT action to send the in/output to the transverter in/output.

RxFILTERBounds – This allows you to directly set the upper and lower bounds on the receive bandpass filter. You have to know how they should be set for the various modes or you won't get what you want. The RxFILTER action allows you to set the width you want and worries about where to set the upper and lower bounds depending on your mode at the time you click the button.

Multiple Radios/multiFlex

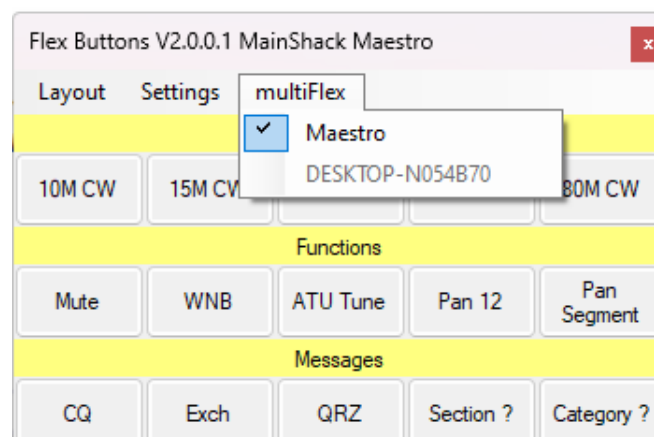
We deal with multiple radios. If, during the discovery process, we find more than one radio a third menu selection appears called “Radio”. Clicking on it will allow you to select which radio FlexButtons will connect to. The currently selected radio will be shown with a check.



multiFlex allows multiple programs to access a Flex 6x00 radio and provide services similar to SmartSDR. Currently, a maximum of two programs (stations) is allowed. On receive they can operate independently (band, mode, etc.). On transmit, only one has control.

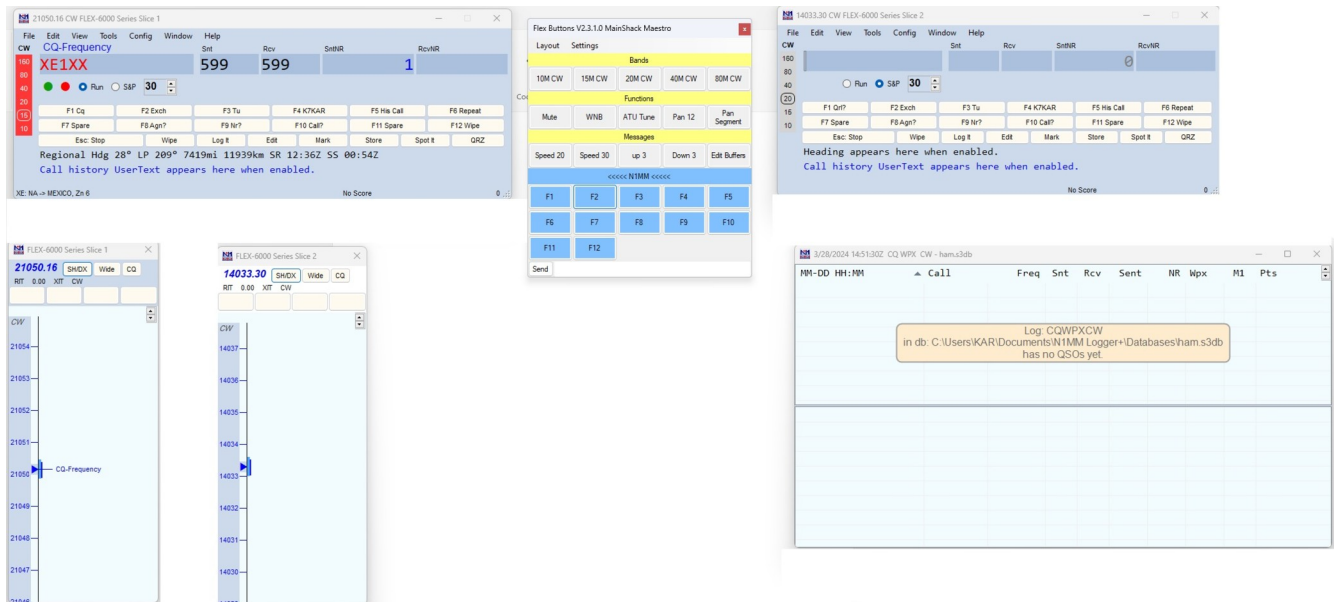
FlexButtons is NOT a program that counts as a multiFlex client, but it interacts with the radio, so it could be associated with either of the two stations. If you are operating in multiFlex mode FlexButtons will add a multiFlex menu. If you click on the multiFlex menu it will show the (two) multiFlex stations. The one that is in control will be checked.

The picture below is a multiFlex environment showing the multiFlex menu. It cannot change the controlling station. That must be done at the station desiring control.



Integrating with N1MM

Generating a sidetone for remote stations seems to be a fairly large target of discussion recently. I added the ability to locally generate a sidetone for CW messages sent by Flex Buttons, but those messages can't utilize info from a logging program. I recently spent some time figuring out how to extract the dynamic information from N1MM and can add the sidetone as well as ask N1MM to send a particular message. The limitation is that Flex Buttons won't read the keystrokes (F-Keys) sent to N1MM, you have to click buttons in Flex Buttons. The Windows security folks and Anti-Virus programs spend a lot of effort to keep one program from intercepting keystrokes from another these days due to the !@#! hackers who want to steal your passwords.



In the above picture an N1MM setup for SO2R and Flex Buttons in the middle showing a blue banner and buttons at the bottom. The banner text is “<<<<< N1MM <<<<<”. Indicating that Flex Buttons is looking at the left (Slice 1) screen for action. Clicking on the banner will change the text to “>>>>> N1MM >>>>>” indicating Slice 2.

Clicking on one of the F1 – F12 buttons will cause Flex Buttons to grab the callsign, sent RST and sent serial number from the N1MM screen, grab the line from the current CW messages file (*.mc), fill in the relevant info and generate the sidetone signal. At the same time it will effectively press the corresponding F-Key button in N1MM causing N1MM to send the actual CW message.

I haven't as yet covered all of the possible macros in N1MM such as speed up and speed down, but it works for pretty vanilla exchanges.

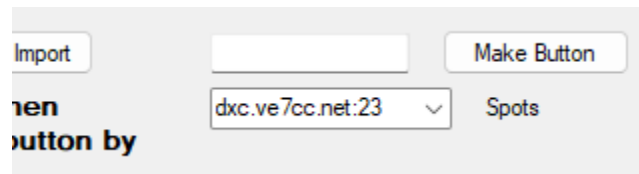
For modes other than CW clicking the F1-F12 buttons just sends the request to N1MM.

No changes are made to the N1MM database, macro files or ini file by Flex Buttons.

Another version of N1MM integration is with the CW (RTTY) SendText buttons. If N1MM is present and the UseN1MM flag is set you can create a SendText button with text "C% R% MT %N C%". This is a possible "Run Exchange" message where C% is replaced by his call, R% is replaced by the sent RST and N% is replaced by the current serial number. I think that covers most contest exchanges with variable data. Flex Buttons will find the call, RST and serial number that you have entered in the N1MM entry window.

Spots

This function will allow spots from one of the DX Clusters to be added to your radio's display. Most of the DX Clusters will work, but I've only tested it with the dxc.w3lpl.net and dxc.ve7cc.net sites. You can enter another site on the configuration screen by entering it into the combo box in the form url:port where the url doesn't need the http:// and the port is the telenet port number for the site (usually 23 or 7373).

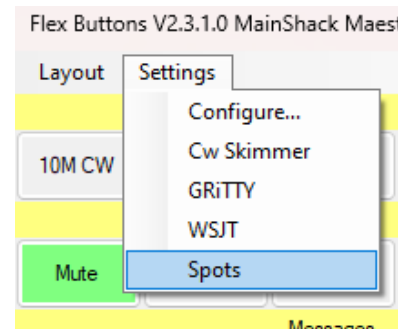


Import Make Button

When button by Spots

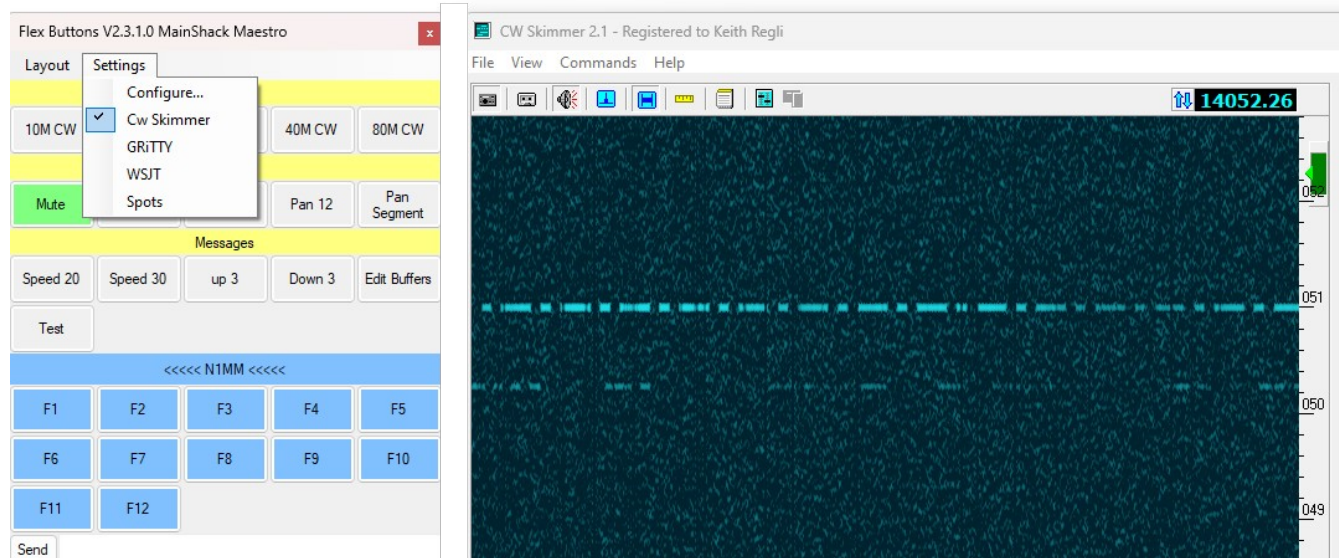
If you do not want Flex Buttons to feed spots to your radio leave the selection as None. This will also prevent CW Skimmer from feeding spots (see below).

When you select the Spots item from the Settings menu it will start the connection with the DX Cluster. It will also check the Spots item. Selecting Spots when it is clicked will stop the connection with the DX Cluster.



Integration with CW Skimmer, GRiTTY and WSJT

You can start one of these programs by selecting it from the Settings menu. The program can be stopped by selecting it again (a checkmark will appear while it is running). Selecting another of the programs will stop whichever one is currently running.



For GRiTTY and WSJT all that Flex Buttons does adjust their “.ini” files to put in the settings based on the radio and then start them.

CW Skimmer has adjustments to it’s “.ini” and maintains a connection so that spots from the skimmer can be added to your radio. The spots will be ignored unless you also have DX Cluster spots turned on.

Final Comments

Remember to start your radios first so that FlexButtons can pick them up when it starts. It can pick up some things while active, but not all.

There is no reason you cannot run multiple copies of FlexButtons with different layouts at the same time, but there is only one *current* layout and that will be the last layout selected, so configuring with multiple copies running is a bad idea.

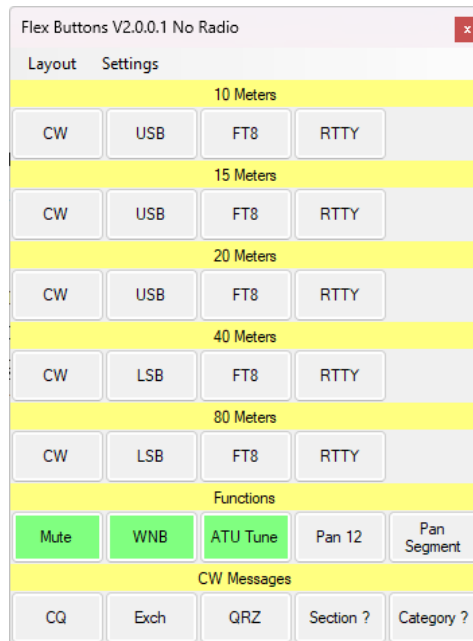
Here are a couple of layouts for really different operators.



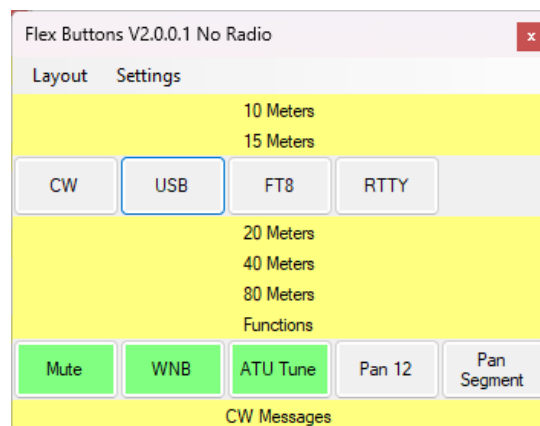
The one on the left is set for someone who checks into several nets and is interested in ARRL activities. Each of these buttons will set the radio to a specific frequency and mode (and possibly invoke the ATU).

The one on the right might be used by a contester. When they're running they'd mostly click the "RIT to 0" button. When they're doing S&P they'd either tune from the bottom of the band to the top and then move back to the bottom with a click (or the reverse). They might want to pan the entire band segment to see where the activity is or where the spots are and then switch back to 10 kHz pan to tune in stations.

Another layout scheme involves collapsing gaps. If you choose to have all bands/modes on buttons, you may like this scheme. The layout below:



Can be collapsed to something like:



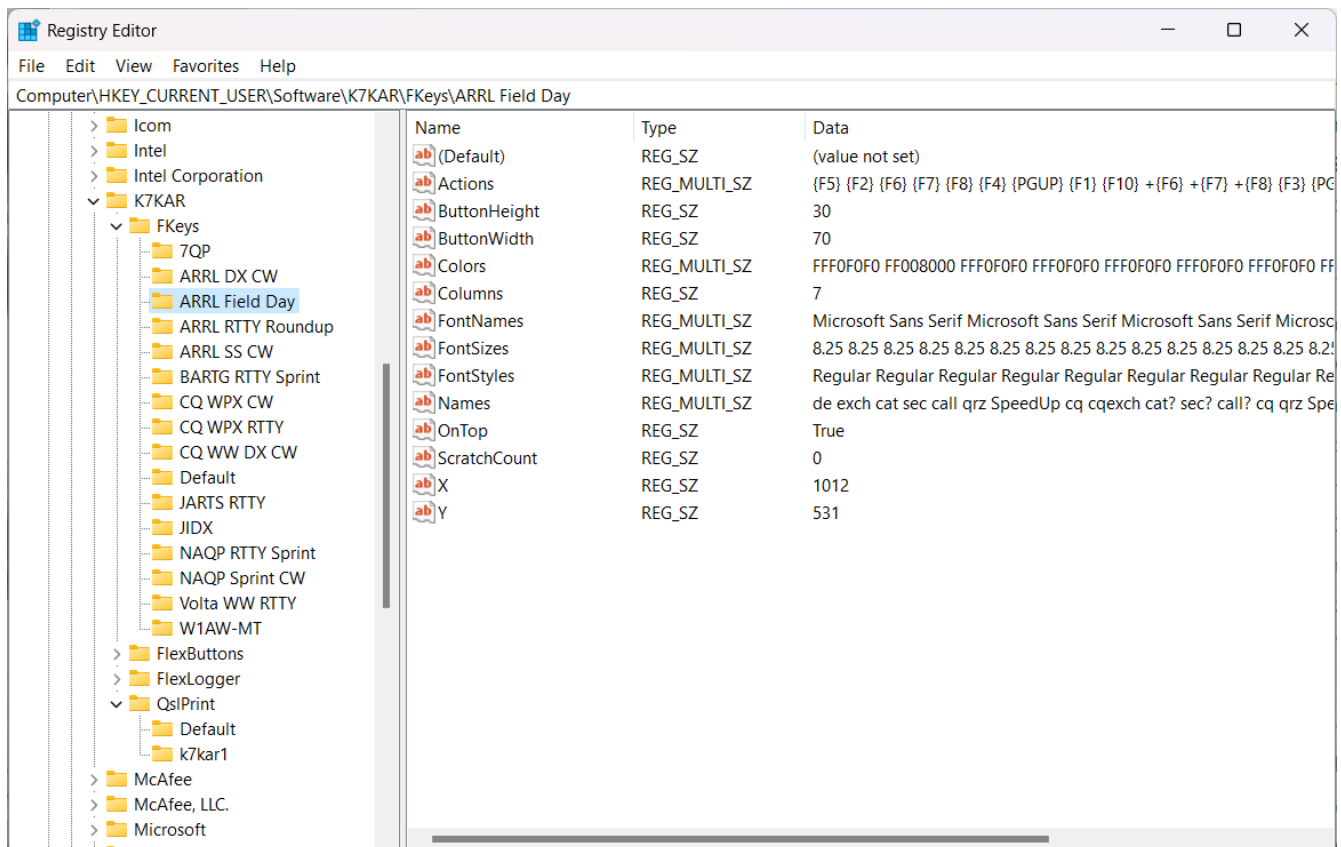
This allows you to save some screen real estate and focus on the band in question.

My original vision for this program was that I'd have five or six buttons that would do things that were a pain to do on the Maestro. After making it available to others, I found many would start with 20 or so buttons and want to add more. I suspect people will come up with all sorts of configurations that I've never imagined! Tell me about them – I might be able to make it easier to setup or use.

Have fun.! If you have questions or suggestions send an email to me at keith@kregli.com.

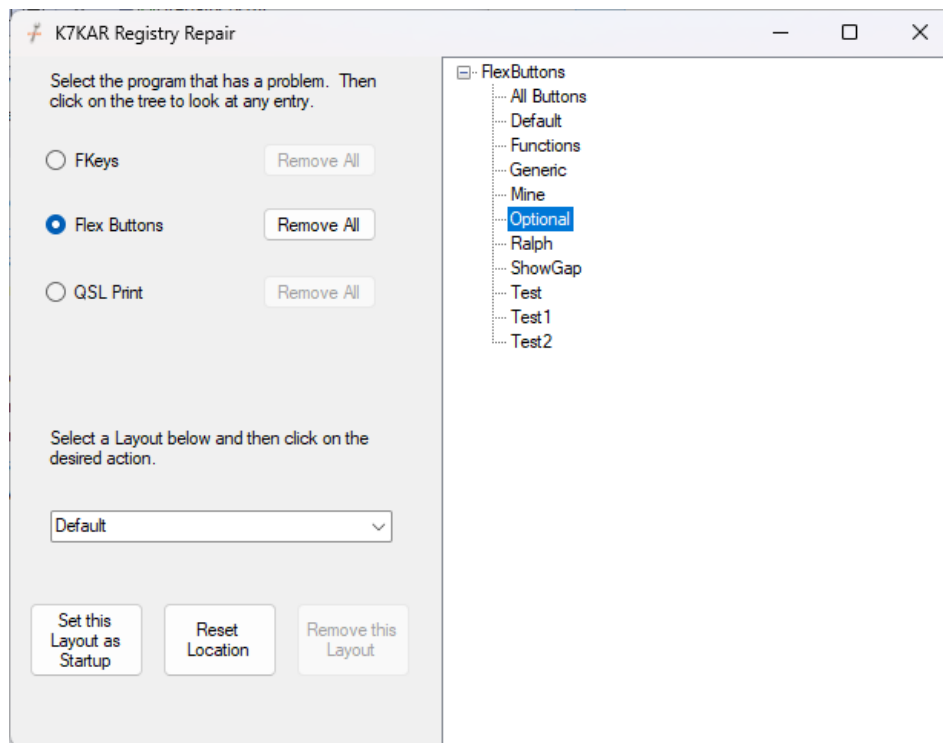
Repair K7KAR Utility (1.0)

This little utility can be used to fix up problems in the Registry associated with my programs. Generally these programs have Registry entries under HKEY_CURRENT_USER\Software\K7KAR. This is pretty typical for such programs. Under this key will be a key for each program (FKeys, FlexButtons, etc.). Under the program key there is some stuff that the program needs to operate. This stuff used to be stored in “*.ini” files, but now lives in the Registry. This makes it less accessible to users who might corrupt it, but makes it harder to fix when something goes wrong. Most of the problems with my programs are caused by me making changes to how things are stored from one version to the next (sorry). My programs have reached the point where each has some generic stuff stored at the top level (usually version # and last “layout” used). Beneath the top level is a key for each “layout”. The layouts may be a collection of buttons associated with a contest or QSL configurations. Under each layout are fonts, sizes, text, actions, etc. Below is a picture of a sample Registry view showing what this looks like:



Users are discouraged from using the Registry Editor because you can mess up your Windows installation if you are not careful. I made this utility so you can see what’s happening and fix most problems. This is version 1, so it doesn’t do as much as I’d like; but it is a good start.

Let’s take a look at the program.



The “Remove All” button allows you to remove Flex Buttons from the registry. You’ll lose any layouts you’ve made. If you run Flex Buttons it will re-initialize the registry entries.

You can pick a layout to start up with – this is useful if you really mess up a layout and can’t figure out how to get out of it.

Reset Location is useful if for some reason the saved position for Flex Buttons is off your display.

Remove this Layout is handy for removing a messed up layout.

If you click on one of the layouts in the right-hand panel you’ll see the information about that layout as shown below. It does a cursory validation of the layout.

