WKUSB-AF

including

WinKeyer3 IC & CPO

User Manual

K1EL
HAM RADIO KITS

www.k1el.com

Version 3.0

Introduction

This document will describe WKUSB-AF and its interfaces. All the information you will need to connect WKUSB-AF to your station will be included in this manual along with specific guides for getting it to work with N1MM+ Logger, HRD/DM780, N3FJP AC Log, FLdigi, and MRP40.

What is WinKeyer and why do I need it ?

Due to timing latencies inherent in multi-tasking operating systems like MS Windows, it is very difficult to generate accurately timed Morse in software. All it takes is a higher priority task to demand service and Morse generation will be delayed resulting in wrongly timed dits, dahs, or the spacing between them. A WinKeyer based device, like WKUSB-AF, attaches to a PC's USB port and appears as a serial com port to an application running on the PC. The application, instead of generating Morse in software, sends the letters to WinKeyer. The application has full control over sending speed as well as weighting, spacing, and PTT generation. WKUSB-AF has a paddle interface so a user can break in immediately to handle fast exchanges in contest situations. Electrically isolated keying outputs provide safety and prevent ground loops. Two separate keying outputs allow the user to run two different radios from one keyer without swapping cables. The feature list below shows some of the advanced options supported by WinKeyer. A WPM speed control is provided so that an operator can quickly change Morse transmitting speed.

WKUSB-AF Features

- USB Interface with Mini -B connector
- Iambic CW Paddle Interface
- Iambic A, B, Ultimatic & "Bug" paddle modes
- Two separate keying outputs, configured as
- Key1 with PTT or separate Key1 / Key2 output
- LED transmit indicators
- Solid state relay outputs rated at 350V @120 mA
- Powered from a PC, charger, or battery USB port
- Internal Sidetone speaker or
- CPO Sinewave sidetone audio output
- Front Panel Speed and CPO Volume Control
- Embedded message commands
- Standalone Keyer Operation
- 3D Printed PLA enclosure
- HSCW and QRSS Capability
- Standalone Practice Mode
- ESD protection on paddle and USB inputs
- RTTY FSK transmit mode
- CH340 USB Interface IC

- Uses K1EL's latest WK3.1 IC
- Adjustable Speed 5-99 WPM
- Adjustable Weighting and dit/dah ratio
- Adjustable Keying Compensation
- Adjustable Letterspacing
- Pushbutton shortcuts, see Page 55
- Adjustable PTT lead in and tail delays
- Optional autospacing
- Farnsworth spacing option
- Adjustable dit/dah ratio
- Inexpensive cabling options
- Paddle only sidetone
- Adjustable speed pot range
- Power supply monitoring
- Twelve stackable memory slots
- Three multi-purpose top mounted pushbuttons
- Automatic beacon capability
- Firmware can be updated over USB

How is WKUSB-AF different than WKUSB ?

WKUSB-AF can be thought of as a WKUSB and code practice oscillator (CPO) merged together in one enclosure. This gives WKUSB the ability to produce a sinewave sidetone output that can drive a speaker, headphones, or PC soundcard input. The volume and pitch of the tone are adjustable. WKUSB-AF, like WKUSB, has an internal sidetone speaker that is used mainly for entering standalone commands but it can also output transmit sidetone in case an external speaker is not connected. WKUSB's internal sidetone speaker is disabled by the standalone command A. There are two control knobs on the front panel, one for speed control and one for CPO audio volume. The speed control also doubles as a sidetone pitch control. press and hold the middle pushbutton and audio sidetone will be enabled. Adjust the pitch with the speed control, when done press the red command pushbutton. In a similar way, pressing and holding the right most pushbutton will put WUSB-AF in tune output mode. Press the red pushbutton or paddles to end tune. WKUSB-AF only has three message pushbuttons but by pressing two of them you can send five messages. Lastly, WKUSB-AF is like WKmini in that it only has one output jack which provides two individual key output ports or one key output port with PTT.

CH340 USB-IF

WKUSB-AF uses a CH340 based USB to serial interface which requires a CH340 USB driver as described in Appendix B. The CH340 is a third generation controller which works just as well, if not better, than an FTDI controller with drivers available for Windows, Linux, and Mac.

Product Warranty, Support, and Liability

WKUSB-AF is fully warranted to the original purchaser against defects in materials and workmanship for one year after purchase. This warranty does not cover damage caused by accident, improper care, or lightning damage. Please contact us before returning your WKUSB-AF for repair and we will issue an RMA.

Please submit support questions by e-mail to k1el.kitsinfo@gmail.com

While every effort has been made to insure that the WKUSB-AF design is safe and the documentation is clear and accurate, it is still possible to cause equipment damage or incur personal injury if:

WKUSB-AF is not used as intended, WKUSB-AF is connected incorrectly, Safety guidelines outlined in this document are not followed, or WKUSB-AF is modified in any way.

K1EL cannot be held responsible in these or other similar events.

WKUSB-AF contains no user serviceable components or configurable jumpers. Please do not tamper with or modify the internal circuitry of WKUSB-AF as this will void the unit's warranty and may result in unsafe operating conditions.

Let's Get Started ! - USB drivers

Don't plug WKUSB-AF into your PC until you have read this paragraph !!

WKUSB-AF C uses a CH340 USB interface IC which is very reliable and easy to set up. Windows 10 and 11 include this driver. If for some reason you don't have it installed, links to the CH340 driver can be found on the WKUSB-AF website. There are two versions; a self installing .exe and a zip file containing the individual drivers as used by Windows Update. Please read through Appendix B for our recommended install procedure.

WKUSB-AF Front Panel Controls

There are five controls from the front of WKUSB-AF's. The left rotary control is used to adjust the transmitting speed and CPO pitch. The right rotary control adjusts sidetone audio volume when driving an external speaker. Three pushbuttons on top are used to send/enter messages and enter standalone command mode. Two LED indicators for KEY and PTT are also present on the front panel. When PTT is disabled, the left LED shows Key 1 activity and right LED shows Key 2 activity.

## WKUSB-AF Rear Panel Connections

There are four connectors on WKUSB-AF's rear panel:

The **Audio Out** (AF) connector is an 1/8inch/3.5mm stereo jack which provides a sine wave sidetone audio output with adjustable volume that is capable of driving headphones or speaker. WKUSB-AF produces mono output which is wired to a stereo output jack. The left and right channels are connected together internally. This allows the use of stereo headphones. Please do

not connect a mono plug into this jack this connector, it has to be a stereo type. A mono plug will short the audio signal to ground. Refer to the diagram on the next page for illustrations.

Stereo and Mono output connector wiring.
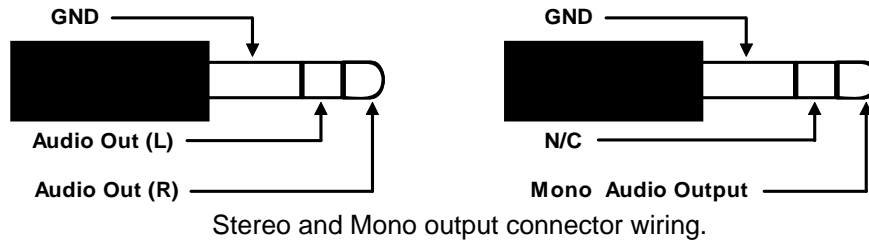
The **KEY OUT** (KEY) connector is an 1/8 inch/3.5mm stereo jack. It has two open drain outputs; Key1 and Key2/PTT1. It can be setup as a single keyed output with an optional PTT or as two individual Key1 and Key2 outputs. The outputs are isolated from WKUSB-AF ground and other signals by an optically coupled, solid state relay which insures a 5000V isolation barrier. The outputs can switch up to +/- 350 V, at a current up to 120 mA. This rating usually allows keying of a vacuum tube transmitter using grid block keying. However, for safety reasons please confirm the keying requirements of the transmitter before connecting it to WKUSB-AF. The diagram below illustrates the connector pinout.

Key Out Connector
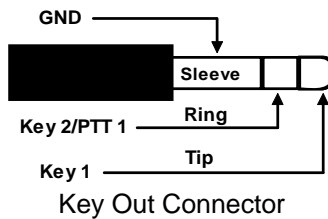
PTT is only provided for KEY1 and is off by default. It can be enabled by software configuration or in standalone mode by WK3tools or paddle command.

Mini-B Interface Cable

USB I/F
The USB connector is designed to mate with a Mini-B cable. A Mini-B to USB A cable is shown at left. The interface will run at USB 1.0 and 2.0 speeds and will also work with USB 3.0 which is backward compatible with USB 2.0 interfaces by design.

The **Paddle Input** (PDL) connector is an 1/8"/3.5mm stereo jack which will accept iambic or single lever (sideswiper) input. A bug is not acceptable. The paddle inputs are activated when switched to the common ground (GND).

Paddle Connector
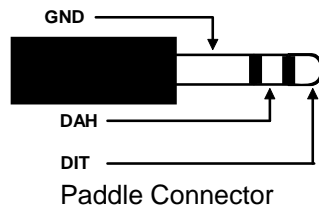
WKUSB-AF Setup

Once the USB driver is installed and you have plugged WK into your PC, you can determine WK's port assignment using K1EL's WKscan app: https://www.k1elsystems.com/WKscan.html. The com port assignment can also be found using Device Manager as detailed in Appendix A.

WKscan Tutorial
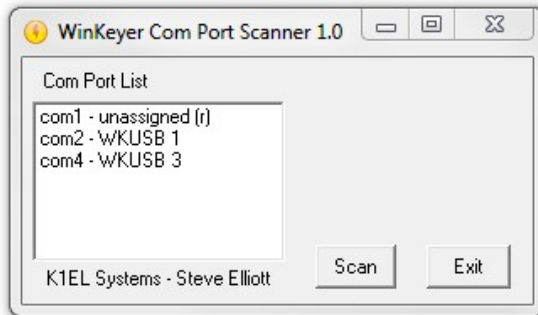
After you have installed WKscan, start it up and you will see the dialog box shown at left. Click Scan and WKscan will display all of the USB serial ports found on the system. The ports will be identified as unassigned, in use, or attached to a WK device. In this example we have a WinKeyer1 device on COM2 and a WinKeyer3 device on COM4. Yes, multiple WKs can be operated from a single PC running two applications.

Once you know WK's com port, you are ready to run WK3demo.

WK3demo tutorial

WK3DEMO is a general purpose test application that can exercise all of WKUSB-AF's capabilities:

https://www.k1elsystems.com/files/WK3demo_42_setup.exe

Once the application is installed, start it up and you will see the dialog box show at left. All you need to do to get started is to set serial communications port. Click on Setup and the com port selection list is shown. Only the currently active USB com ports are shown. Select a WK com port identified by WKscan.

You can click on ? shown in the upper right corner of the dialog box for a brief help menu.
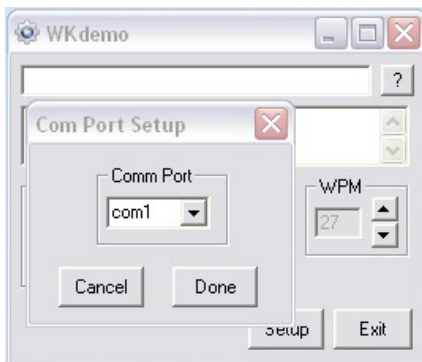
Click on Open to connect to WKUSB-AF. If all is well, the Open pushbutton will change to Close and the WK IC version will be displayed. Click on Setup again and this time the Setup dialog box will be displayed as shown on the next page. This presents all of WK3's various configuration options which can be modified to your requirements.
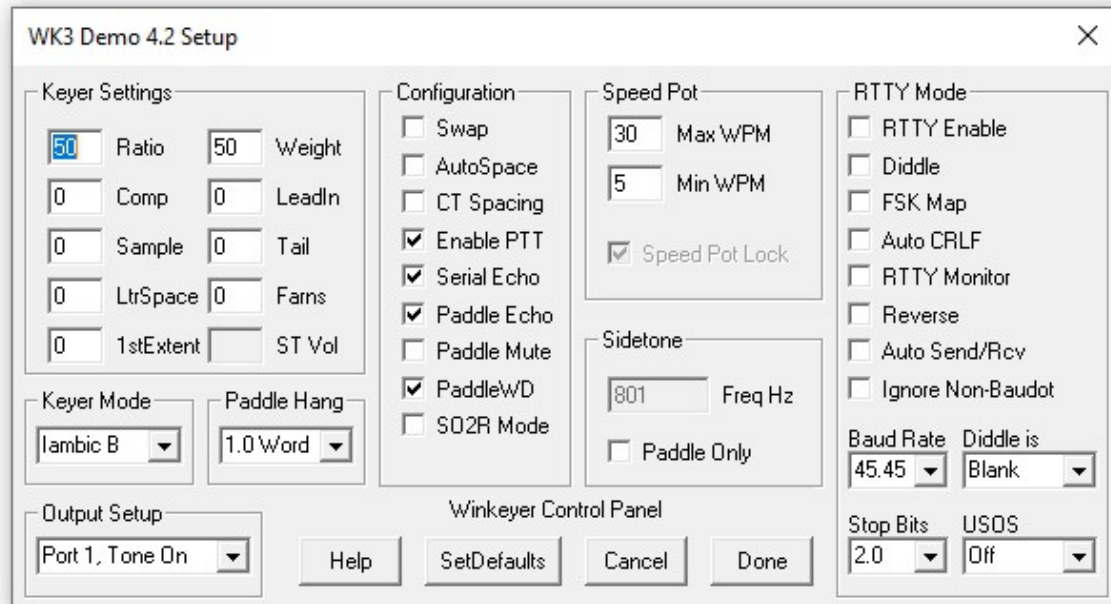
Click Cancel for now and go back to the main dialog box. Type into the lower keyboard entry window and you will hear the text sent in Morse sidetone from WKUSB-AF's internal speaker. Enable Serial Echo in Setup and what you type will be echoed in the upper text window as sent.

Now hook up some paddles and test paddle entry. If Paddle Echo is enabled, letters entered on the paddles will be echoed in the upper text window. Use the Setup Dialog to try different keying options. You probably noticed that if you change settings on the Setup dialog box, you have click Done and return to the main application to transfer those new settings to WKUSB-AF.

WKUSB-AF has two sidetone options. The built-in sidetone speaker operates just like it does in a standard WKUSB. You can adjust the sidetone frequency and set it to sound for paddle entry only. You could also turn internal sidetone off and use WKUSB-AFs CPO sidetone output instead. Attach an external speaker or headphones and then set the sidetone volume control to a comfortable level. Note that even when internal sidetone has been disabled, it will still be used for standalone commands. External sidetone is not activated for standalone commands. To disable CPO sidetone, turn the volume fully counterclockwise to effectively turn it off.

There are several speed pot options, Speed Pot Lock forces WKUSB-AF to only use the speed pot for speed control. When Speed Pot Lock is unchecked, the Up/Down arrows next to the WPM readout on the main dialog box can adjust the speed or you can directly enter a speed in the WPM edit box. In other words, speed is controlled solely by the host application. This is how the majority of PC logging and contest applications operate. Speed is controlled by keyboard commands, macros, or messages with embedded speed commands. The name of the game in contesting is to have one central user focus, that being the keyboard.

Although WKtest packs a lot of power in a small app, it is meant to be mainly a hardware test tool. It gives you a good idea of what WKUSB-AF is capable of and assures you that it is operating properly. The next logical step is to move on to a 'real' application, such as N1MM+. Before that we will look the WK3tools application and how standalone operation works.

WK3tools application

When a WinKeyer device is not controlled by a host application, it will run in standalone mode. WKUSB-AF can be powered over the USB port from a phone charger, PC, or USB power pack. WKUSB-AF in standalone provides a complete Morse keyer with lots of capabilities. Normally, configuration changes are made using commands entered on the paddles. The WK3tools app allows you to make these changes from a PC application.

https://www.k1elsystems.com/files/WK3tools_52_setup.exe

After installation, start the application and the following dialog box will appear:
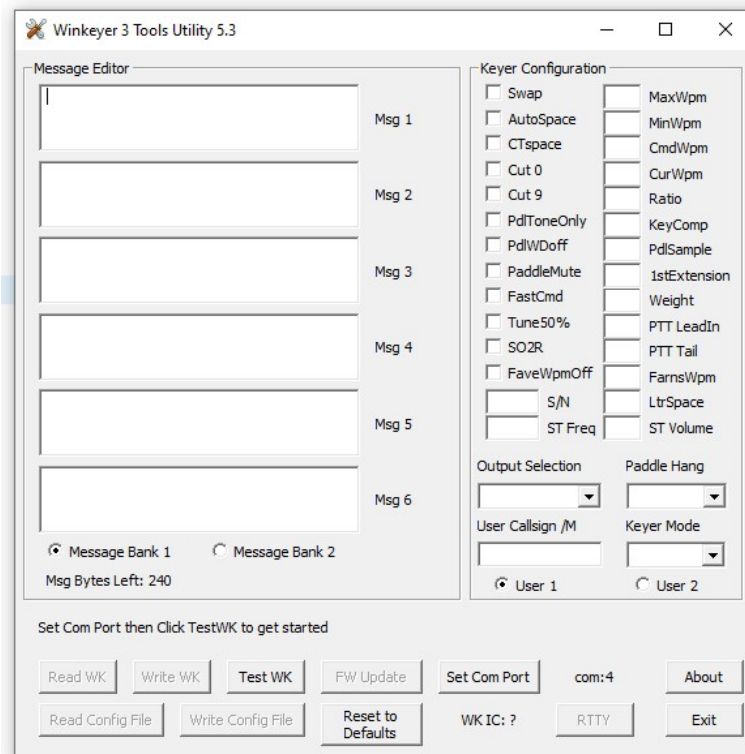
It will be empty since you haven't read in WK's current configuration settings. When WK is powered up for the very first time, all settings are loaded with defaults. To view these, connect WK to your PC, set the ComPort, and click on ReadWK. After a couple of seconds the dialog box will be updated to show you the current settings:
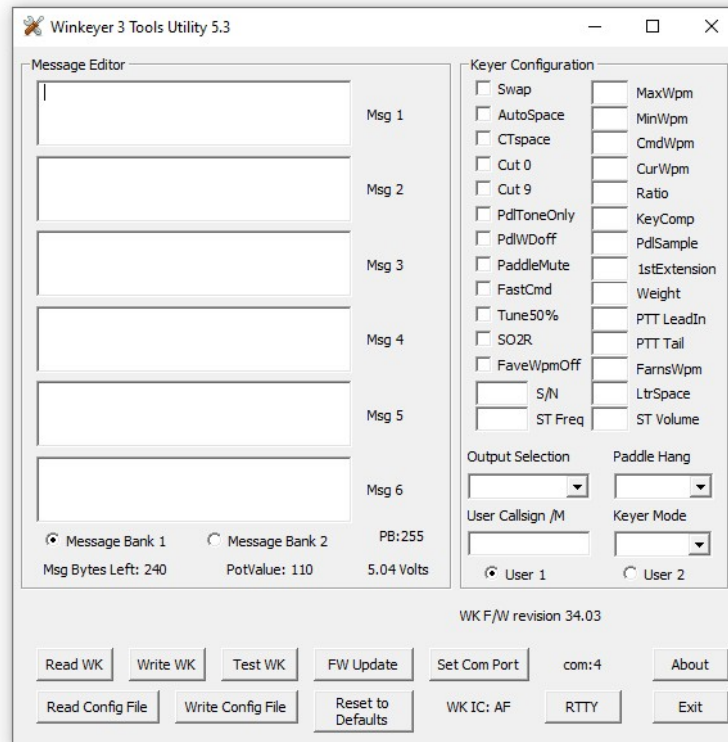
Now you can go in and tailor the standalone settings any way you want. The biggest timesaver is the ability to load messages. A tally of available message storage left is shown below the message text windows. Even though the WKUSB-AF has only three message pushbuttons, it has the capability to handle 12 messages in two banks. Messages 4 and 5, in each bank, are accessed by pressing two buttons (see page 54). Message 6 is accessed by calling it from another message. Don't forget that you have to write the messages back to WK with the WriteWK button when you are finished. If you want to test them, press the buttons on top of WK.

There is an option to setup standalone RTTY operation. Typically you would setup User 2 for RTTY operation so you can easily switch between the modes without having to make changes by paddle commands.

You can enter embedded commands by preceding the command letter with a /
For example a speed change can be embedded in a message like this:

/S20 CQ CQ CQ DE /S18 K1EL K1EL /S20

To insert a DN prosign simply enter two // as in K1EL//QRP
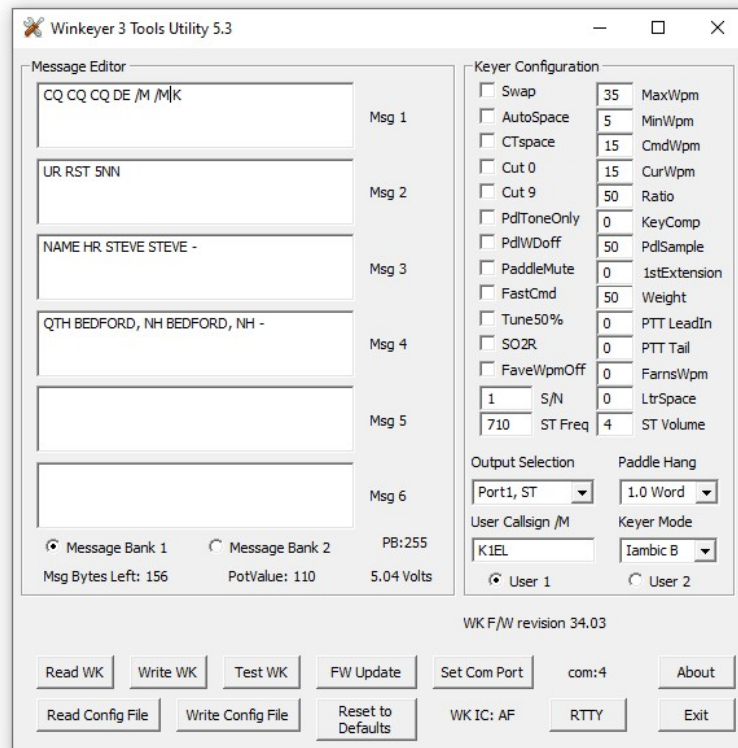
Once you have saved messages and settings, they are ready for use. You can now disconnect WK from the PC. You don't need to use WK3tools again until you want to make changes. Don't forget you can still make quick changes to WK settings by using paddle commands, no PC required. A detailed description of standalone paddle commands can be found in Appendix E.

WK3tools will only work with WinKeyers that have a WK3 or newer WK-IC. WKscan will tell you the IC version. You will not be able to access a WinKeyer that has a WK2 or older IC.

Using WK3tools to Restore 'Factory' Settings
The 'Restore to Defaults' control loads the menu with original default settings. All of the control settings are initialized and messages are erased. Don't forget to write these new settings back to WK before you exit.

WK3tools Help File
A separate help document for WK3tools can be downloaded from the WK3tools webpage:

https://www.k1elsystems.com/files/WK3ToolsOperationGuide.pdf

Upgrading WKUSB-AF Firmware
WK3tools is used to install new firmware revisions. A special webpage is provided for WK3 upgrade information:

https://www.k1elsystems.com/WinKeyer_31.html

RF Immunity

There are several RF filters included in the WKUSB-AF circuit design. Bypass capacitors on the Key and PTT lines prevent RF from coming into WKUSB-AF. Bypass capacitors shunt any RF arriving over the paddle leads, and an LC filter is employed on the USB power input. These components also reduce WKUSB-AF RFI radiation.

PCs and associated peripherals can produce RF emissions on their own that may cause RFI to your radio receiver. In addition, most keyboards have poor conducted RF emissions immunity and operation can be compromised in high RF environments. To guard against this, additional filtering may be required on the keyboard, usually a ferrite bead on the keyboard cable will address this.

FCC Regulations Compliance

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference and (2) this device must accept any interference received, including interference that may cause undesired operation.

Radio and Television Interference

Official FCC compliance testing is in process but has been verified by in house testing.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:
- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

You may also find helpful the following booklet, prepared by the FCC: "How to Identify and Resolve Radio-TV Interference Problems." This booklet is available from the U.S. Government Printing Office, Washington D.C. 20402.

Changes and Modifications not expressly approved by the manufacturer or registrant of this equipment can void your authority to operate this equipment under Federal Communications Commission's rules.

In order to maintain compliance with FCC regulations shielded cables must be used with this equipment. Operation with non-approved equipment or unshielded cables is likely to result in interference to radio & television reception.

WKUSB-AF FAQ #1: What is Standalone Mode and how is it different than Host Mode?

The original Serial WinKeyer1 was designed to operate only while connected to a PC, under control of a PC based application, this is Host Mode. WinKeyer1 obtained power directly from the PC over unused RS232 control lines; if it was disconnected from a PC, it was inoperative. Soon hams started putting batteries in WinKeyers and using them standalone, in other words, not connected to a PC. This spawned many requests to add features to make a standalone WK more useful. When WKUSB was designed, we decided to include a complete 'second' keyer in the box and Standalone Mode was born. This second keyer shares the core 'K1EL Keyer Engine' with Host Mode but in other respects it is independent. It can operate on battery power or connected to a PC. It has its own settings memory and 12 message slots. Commands can be issued via paddle commands or through the use of WK3tools. Standalone settings do not affect Host Mode settings. In Host Mode, keyer settings are stored on the PC by the application and are loaded when you connect to WKUSB-AF. In Standalone Mode the settings reside in EEPROM inside WKUSB-AF. If you disconnect from an application, WKUSB-AF will automatically go into Standalone mode and revert back to the saved settings from EEPROM.

WKUSB-AF FAQ #2: Does WKUSB-AF have internal batteries like WKUSB does ?

No, WKUSB-AF is powered only from the USB I/F connector. It has to be plugged into a PC, a phone charger, or USB battery. Due to the extra power draw of the audio amplifier circuit, WKUSB-AF does not go into full low power sleep like WKUSB does when not active. It does reduce current draw in semi-sleeo mode to about 10mA.

WKUSB-AF FAQ #3: What about RTTY transmit, how does that work ?

Starting with WK version 3.1, RTTY transmit capability was added to WK. In addition to being able to run hardware FSK RTTY with N1MM+, FLdigi, and WK3demo, you can also run RTTY from a paddle in standalone mode. In depth details on how the RTTY API works can be found in Appendix A on page 36. Setting up RTTY for N1MM+, FLdigi, and WKdemo are covered in this document on pages 22, 32, and 33. Paddle mode RTTY is covered on page 60. WKUSB-AF is only capable of RTTY FSK transmit, it does not have any RTTY receive capability.

Most importantly, you do not need to mess around with Arduino programming, sketches, or image files to use WKUSB-AF in RTTY mode. The mode is included in the standard feature set of WinKeyer 3.1.Changing back and forth between CW and RTTY is quick and easy,

Connecting WKUSB-AF to your station

Minimally all you need to do is connect the USB cable from WKUSB-AF to a USB port on your PC and connect the key output to the keying input on your rig. Since WKUSB-AF performs the keying function it drives your rig in the same manner as a straight key. In other words there are only two leads, ground and CW keying control. For that reason a keying cable is quite easy to build.

K1EL Systems sells a WKUSB-AF universal keying cable kit that covers 95% of keying requirements.

Keying cable details

An 1/8" stereo plug is required at the keyer end while the other end will have a connector that will mate to your rig's key input. This is usually an 1/8 or 1/4 inch phone plug, we highly recommend using a stereo plug at least at the transmitter end. A two wire cable is all that is required to key a typical transceiver. The key line goes to tip and ground goes to the sleeve.
.
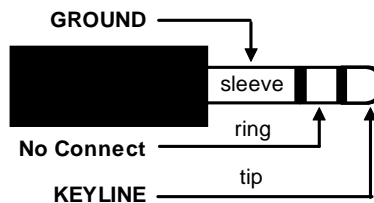
Transceiver/Transmitter end of cable

Some transceivers have two keying inputs, one for a paddle and a second for an external keyer. A dedicated keyer input makes it easy, just plug WKUSB-AF in and go. In the single keyer input case, make sure you disable the radio's internal keyer, if it has one, and set it up for straight key operation. You will also need to make sure that the radio is in full or semi break-in mode so the transceiver will automatically change to transmit when it is externally keyed,

It's always a good idea to test your keying cable before using it with WKUSB-AF. Plug the cable into your transceiver and leave the keyer end unconnected. Now set up your transceiver for external keying and then short the keyer cable end, tip to sleeve, with a clip lead. If this doesn't key the transmitter, then either the cable is not made correctly or the radio is not in break in mode. If you get automatic dits or dahs, then the keyer in the radio is still enabled. Some transceivers want you to use a stereo plug even if the internal keyer is disabled, it's a good idea to use a stereo connector at the radio end no matter what and leave the ring unconnected.

If you want to key a transmitter that uses vacuum tube finals, WKUSB-AF will usually work fine because it has a solid state keying relay rated for +\-350V switching at 120ma. This covers grid block keying but it probably will NOT be suitable for cathode keying. In any case always check the transmitter's manual for keying requirements <u>before</u> plugging WKUSB-AF in.

WKUSB-AF's key output connector is electrically isolated from WKUSB-AF's enclosure and ground. This means that you can directly connect your transceiver to WKUSB-AF with PC attached and not worry about ground loops or RF getting into your PC through WKUSB-AF. Make sure your shack ground is adequate to prevent stray RF from coupling into the paddle or USB interfaces which can result in host application hangs or disconnects.

PTT Connection
PTT will not be used in most radio setups. It is provided to run an external antenna switch or to pre-key a linear amplifier. If you are just driving a barefoot transceiver, PTT will not be required.

If you do need PTT, then WKUSB-AF provides useful functionality. PTT is asserted before KEY, is held for the entire transmit interval, and released after KEY. Both lead-in and tail delays are adjustable. If PTT is not working, make sure that it is enabled in your configuration.

Paddle Connection
If you decide to use a set of keying paddles with WKUSB-AF, they plug into the paddle jack input on the rear of the unit. An 1/8 inch stereo jack is used on WKUSB-AF. If your paddles have a 1/4 inch size plug then a 1/8" to 1/4" stereo adapter will be necessary.

WK3demo is useful for verifying that WKUSB-AF is keying your rig properly. If you run into problems, make sure you have selected the correct WK key output port. This is done in the Output Config control in the WKdemo setup dialog box. Also check to make sure the connector is fully plugged into the WKUSB-AF jack. The tune function is useful for signal tracing.

A cheap path to a keying cable

In today's world of cell phones and portable music players, 1/8" stereo to 1/8" stereo cables are very easy to find and are inexpensive. If your transceiver requires an 1/8" keying plug you are done. If it needs a 1/4" plug, purchase an 1/8" to 1/4" stereo adapter.

An 1/8" stereo plug to dual RCA cable can be very useful. Find two RCA to 1/8" to 1/4" adapters and you have a keying cable that can key two radios.

The point is that you can come up with just about any cable you need by juggling cable types and adapters. Or buy one of our cable kits from us.

Note that 3.5mm and 1/8" connectors are the same size, so don't let that confuse you.

Troubleshooting hints

1) *When I plug WKUSB-AF into a USB charger I don't get the initial R sent in sidetone.* This means that that the WK3 IC is not starting up. Make sure the charger is working. If you are using a USB charging battery be aware that most of these have a minimum working current requirement. WKUSB-AF draws about 20 mA which may be below the turn on threshold of the battery. If after double checking you don't get the R, that signifies a fault in the unit. Please contact K1EL for further instructions.

2) *WKUSB-AF won't key my radio.* First check to make sure your keying cable is wired correctly and then check to make sure it is plugged into WKUSB-AF's KEY output. In standalone mode, be sure you haven't accidentally issued a TX Mute command.

3) *I get dits or dahs only or CW letters are misformed.* This usually happens when you have not disabled your transceiver's internal keyer. Make sure you have set the transceiver to straight key mode. Usually this is controlled by a transceiver menu selection.

4) *When I power WKUSB-AF up, I get one R in sidetone but nothing else works after that.* This is a sign that one of the dit or dah paddles is stuck low or there is short circuit in the cable. Remove the paddle connector and see if that clears the problem. .

5) *When I try to run WKUSB-AF from WK3demo I get the error message Attach WK: No Echo.* This means that the com port selected was a valid one but not the one that WKUSB-AF is connected to. Check the port selection using WKscan or device manager

6) *WKUSB-AF is plugged in but WKscan does not show a com port with a WK device.* This is a sign that either a USB driver has not been installed or the USB hardware on WKUSB-AF is not working. It could be a bad USB cable or bad connection to the PC. Note that WKUSB-AF's USB interface is factory tested before a board is shipped so if it doesn't work the it's usually the PC connector or cable. Make sure that the USB cable is fully inserted at either WKUSB-AF or at the PC.

7) *I get an "Access to Port xx denied" message.* This probably means that one of the other K1EL programs or a non-K1EL program is open, perhaps in the background out of sight, and is still using the port, thereby locking out all other programs from accessing the port. To solve this issue, be certain that all other programs are closed and then try again.

8) *WKUSB-AF is 'kicked off line' while you are transmitting.* There's a good chance RF is getting into the keyer or the PC disrupting USB communication between the PC and WKUSB-AF. To solve this issue, first make sure the grounding system in your shack is properly designed and has the shortest possible path to a ground rod outside. If that doesn't help, you can try a 'band aid' fix by installing snap-on ferrite "beads" on each end of the keyer to PC USB cable.

9) *I setup WKUSB-AF and all was working until the PC went to sleep overnight, now the host app doesn't see WKUSB-AF.* The solution is to close the application and restart. WKUSB-AF is designed to be USB compliant and handle the condition where the host PC goes to sleep and then wakes up. Unfortunately, not all host applications are designed to preserve the connection to WKUSB-AF. A working solution is to disable the PC from going into full sleep in the PC's Power Settings. Allowing only a display and hard drive to sleep will still save a great deal of power while keeping the CPU awake to preserve USB connection.

Paddle Input Priority

WKUSB-AF accepts input from either host port or paddle. Paddle input will always take priority and will interrupt messages or Morse sent from a Host application. When a paddle break-in occurs, any remaining message data will be cancelled.

No USB Sense or low power standby

Unlike WKUSB which has internal batteries, WKUSB-AF is operational as long as it is plugged into a power supply and does not go into low power standby mode.

Serial Baud Rate

WK's standard baud rate is 1200 baud with an alternate baud rate of 9600 selected by command. This is all handled by the host application so there is no need or requirement to set the USB port's baud rate, stop bit, or parity settings. All of these things are configured automatically by the host application when it opens and connects to WKUSB-AF.

Reset WK3/Restore Factory Defaults

There are cases when a host app misbehaves and can corrupt WKUSB-AF's basic setup parameters. We know of no app that does this but when under heavy use with RF floating around unexpected things can happen, ESD is another issue that can cause this to happen. The easiest way to restore WKUSB-AF's EEPROM settings is through WK3tools. Unplug WKUSB-AF from the PC and then plug it back in after 10 seconds. Start up WK3tools, set the com port and click WKtest to be sure it can connect. Then simply click Reset to Defaults followed by WriteWK.

ASCII Code Assignments, Prosign Mapping

WKUSB-AF maps several unused ASCII codes to standard prosigns as shown in the table below. Some abbreviations are not mapped (null), WKUSB-AF will ignore these. Codes below 0x20 are commands, 0x30 to 0x39 are numerals 0-9, 0x41 to 0x5A are letters A-Z. All codes above 0x5D are ignored except for | (0x7c) which is the ½ dit pad code.

| ASCII	Hex		Prosign		ASCII	Hex		Prosign
	0x20	*Is space*			.	0x2E	*Is period*	
!	0x21	*Is mapped to*	(null)		/	0x2F	*Is mapped to*	DN
"	0x22	*Is mapped to*	RR		:	0x3A	*Is mapped to*	KN
#	0x23	*Is mapped to*	(null)		;	0x3B	*Is mapped to*	AA
$	0x24	*Is mapped to*	SX		<	0x3C	*Is mapped to*	AR
%	0x25	*Is mapped to*	(null)		=	0x3D	*Is mapped to*	BT
&	0x26	*Is mapped to*	(null)		>	0x3E	*Is mapped to*	SK
'	0x27	*Is mapped to*	WG		?	0x3F	*Is question*	
(	0x28	*Is mapped to*	KN		@	0x40	*Is mapped to*	AC
)	0x29	*Is mapped to*	KK		[	0x5B	*Is mapped to*	AS
*	0x2A	*Is mapped to*	(null)		\	0x5C	*Is mapped to*	DN
+	0x2B	*Is mapped to*	AR		]	0x5D	*Is mapped to*	KN
,	0x2C	*Is comma*			|	0x7C	*Is ½ dit pad*	
-	0x2D	*Is mapped to*	DU					

N1MM Interfacing

You will first need to know your WKUSB-AF's comport assignment, if you don't know it, follow the WKscan Tutorial on page 6 or use Device Manager as described in Appendix C (page 40). Be sure to close WKscan after using it.

Start up N1MM+ and click on the Config Tab and then select the entry as shown:
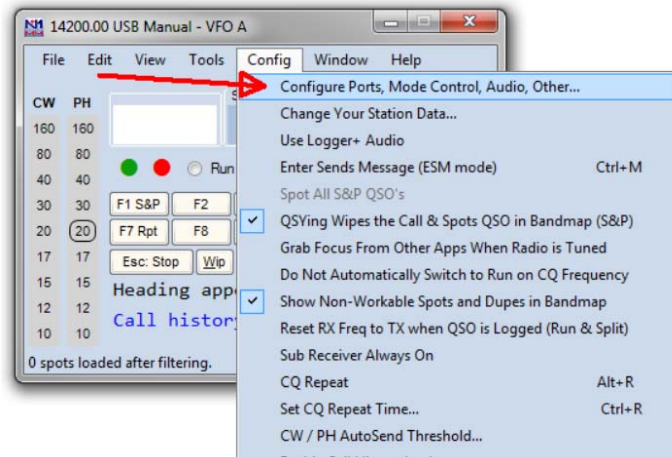
This brings up the Configurer dialog box. This is where devices attached to N1MM+ are set up. For this example we will set up a WinKeyer device on COM2.
1) Select COM2 from the drop down box.
2) Specify that it is a CW device by clicking the CW/Other checkbox
3) Continue on to the Com Port setup dialog box by clicking Set
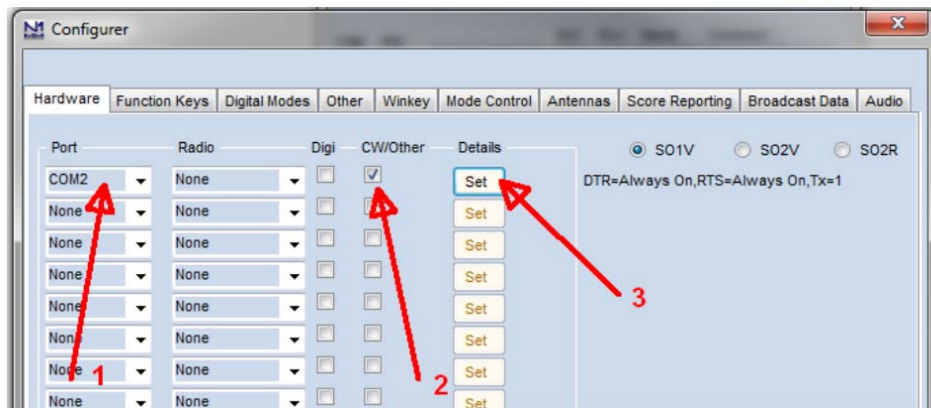
Clicking Set brings up the Com2 configuration dialog box. All we need to do here is click the WinKey checkbox, leave all the other settings as is.
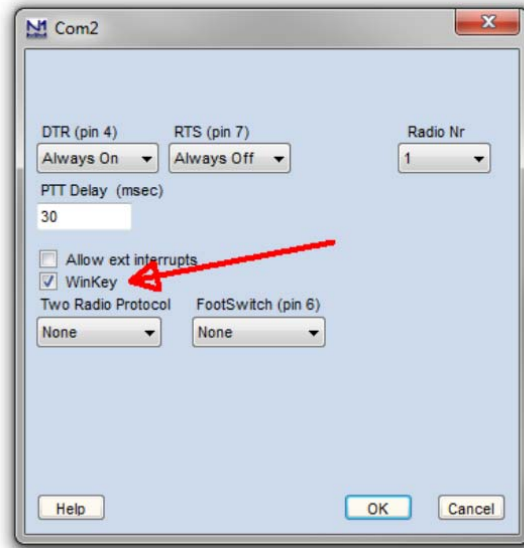
Now click OK to close the Com2 configuration dialog box which brings us back the Configurer dialog box. Here we select the Winkey tab as illustrated below:
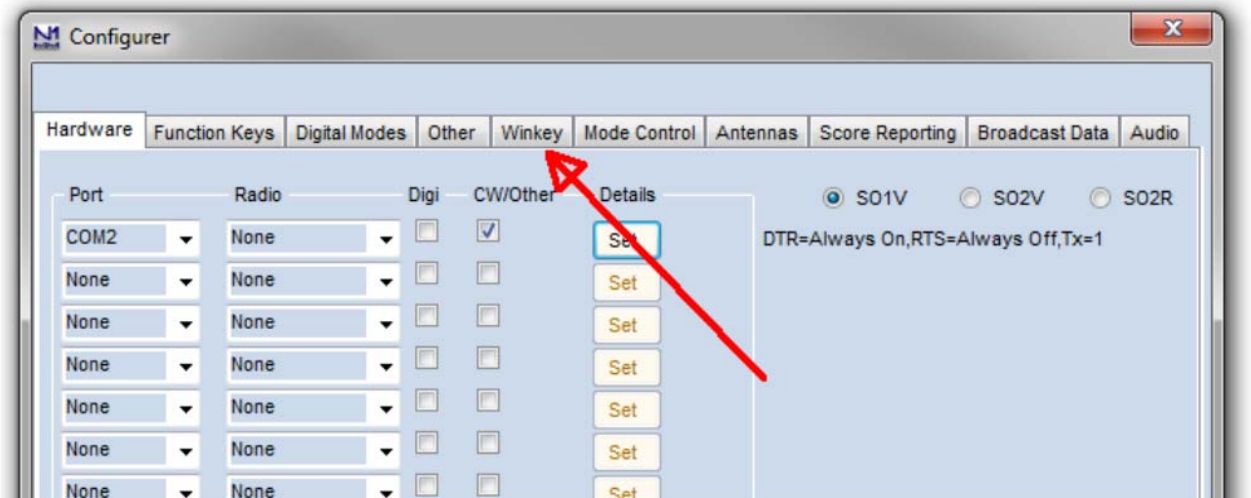
Clicking on the Winkey tab brings up the WinKeyer specific configuration dialog box which is shown on the next page.
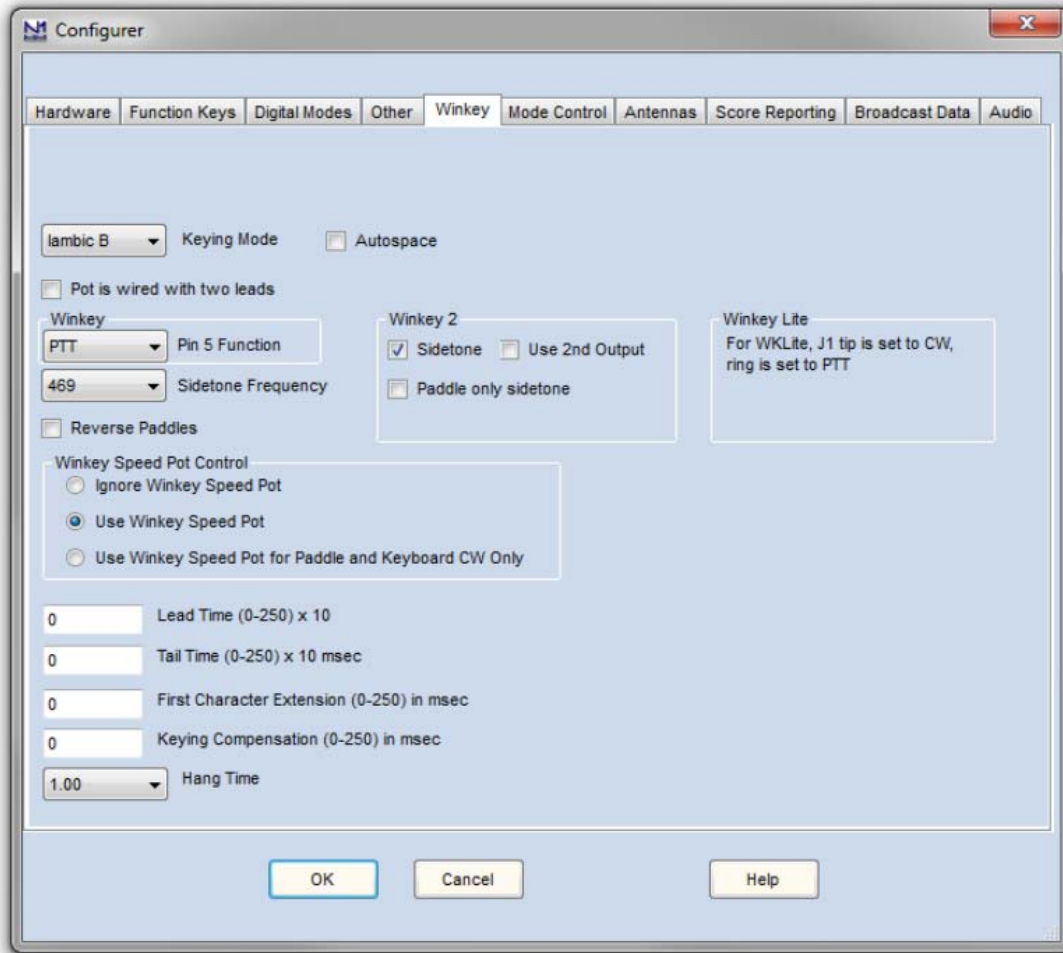
There are very descriptive hints on most of the controls.

Winkey 2 options also apply to WK3 based keyers.

Disable Paddle Only Sidetone for initial testing purposes.

Winkey Speed Control has three options:
1) Ignore Speed Pot which only allows the host to control sending speed
2) Use Speed Pot which tells the host that the speed pot will set sending speed
3) Paddle & Keyboard where the host controls only the message speed

Lead and tail settings apply to PTT delay Keying compensation adjusts dit and dah spacing relative to sending speed.

Hang Time is similar to PTT tail except that it is proportional to sending speed.

When you are finished, click OK and then another OK to close the Configurer. Now that we are back to N1MM+ main control dialog box we can do some WKUSB-AF testing. First off select CW mode by clicking on a band in the CW column:
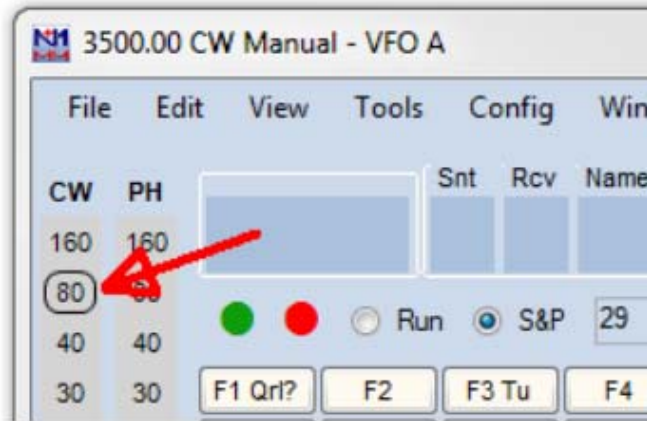
Now on the PC's keyboard enter CTL-T to put WKUSB-AF into tune mode. Cancel tune mode by entering CTL-T a second time. With tune enabled you should hear sidetone coming from WKUSB-AF. Now either enter CTL-K or select the CW Key Window as shown:
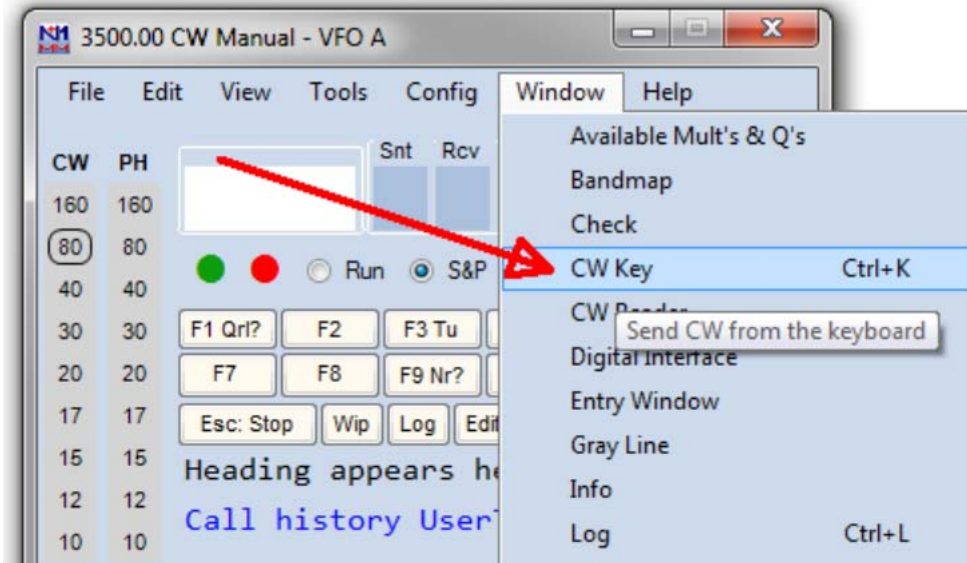
The CW keyboard window is now displayed and you can type into it to send letters to WKUSB-AF. Note that when the Send CW window is up, the main N1MM+ dialog box is disabled. For a final test, turn the speed pot and you should see the CW Speed display in the main N1MM+ dialog box change. (This assumes that the speed pot has been enabled in the Winkey configuration dialog box)
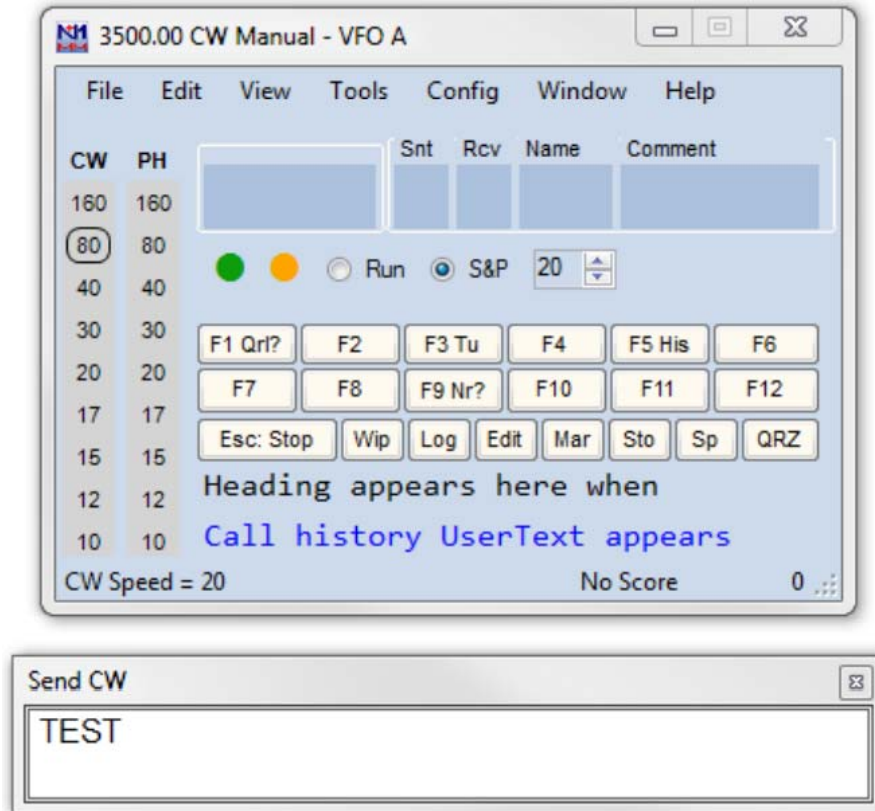

That's it for basic N1MM+ and WKUSB-AF testing. Now it's time to learn all the amazing features in N1MM+ for setting up messages and getting up to speed with automatic CW contest exchanges.

Running WKUSB-AF with N1MM+ in RTTY mode

Since WKUSB-AF has a WK 3.1 controller IC, you can run it with N1MM+ in RTTY mode. Bring up the Configurer dialog for WinKeyer and you will see a new RTTY mode selection option:
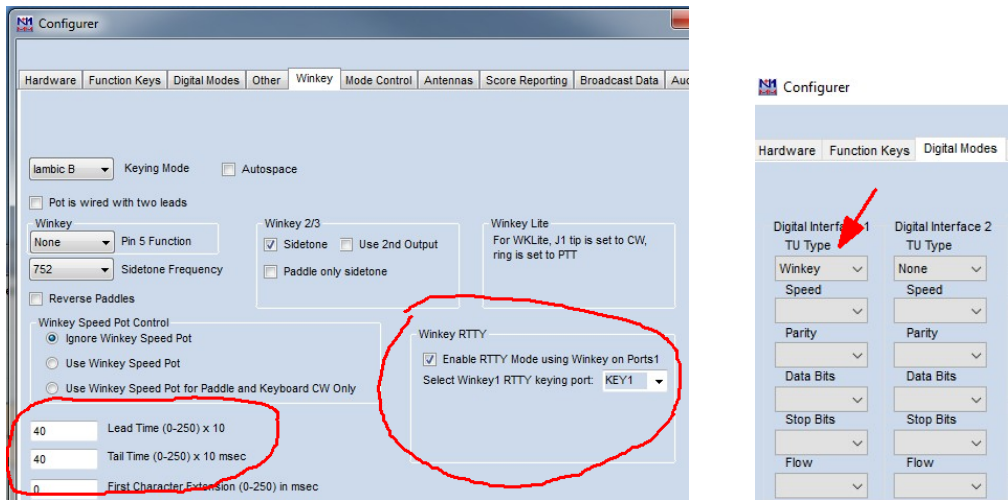

Click on Enable RTTY and then set a delay in both Lead Time and Tail Time; 40 mSec in each is a good starting point. Now click on the Configurer Digital Modes tab and in the left most drop down, select Winkey. Now close the Configurer and go back to the main N1MM+ window. From there we will open up the digital interface window:
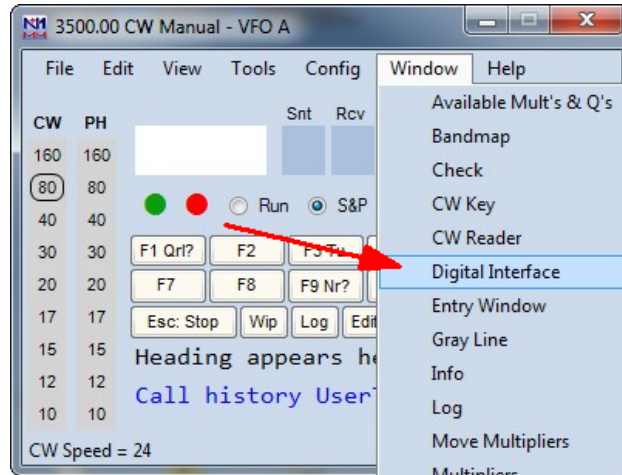

There are several things to set up in the digital interface window, first we will setup the macro keys for Tx, Rx, and ESC. Carefully follow the next instructions.
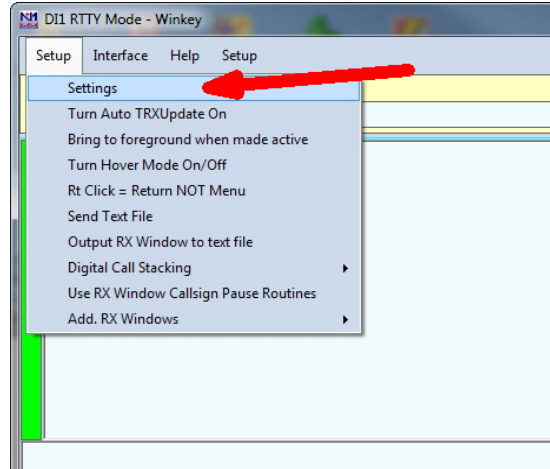
Click *Setup->Settings* to bring up the Digital Setup Window, click on the Message Setup tab and then select "Other, TinyFSK, Winkey" from the drop down menu.
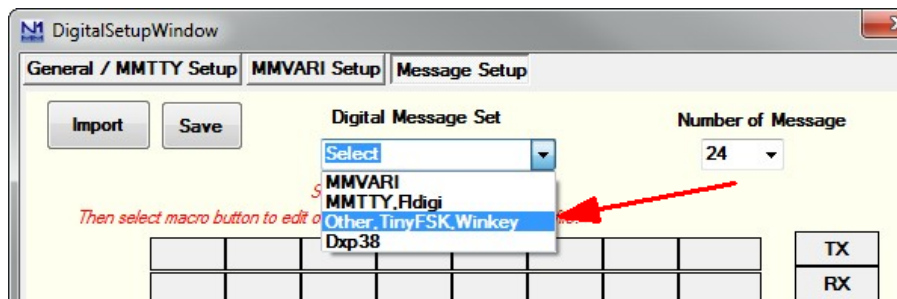

Using the picture below as a guide, click **Tx**, then enter **[** in the message Text Window followed by **Save Settings**. Repeat this process by entering **]** for **Rx** and then **** for ESC. Note that you have to re-select "Other,TinyFSK,Winkey" each time.
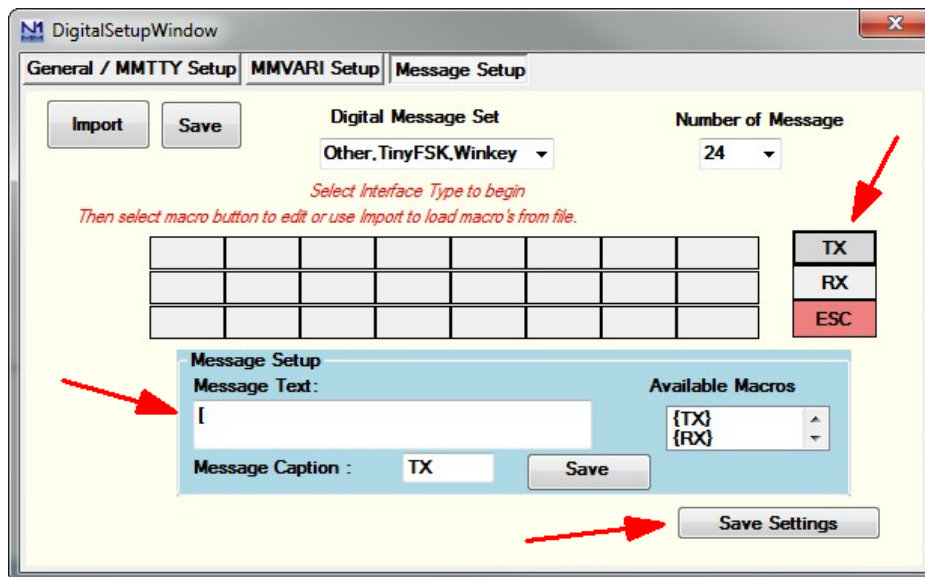
The fastest way to get into RTTY mode is to enter RTTY in the entry box as shown and hit Enter.
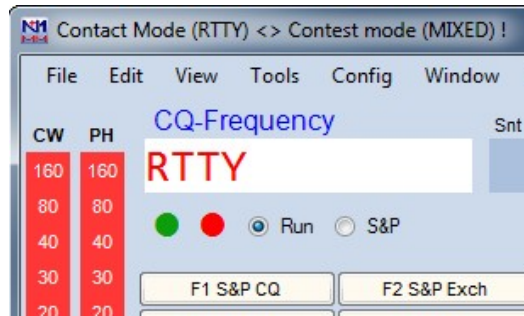
Now go back to the Digital Interface window, and select your desired RTTY operational modes:
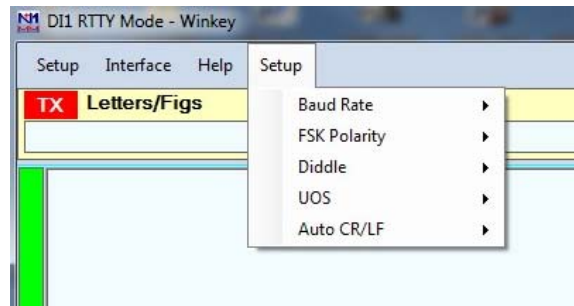
Now you are all set to run N1MM+ in RTTY mode with WKUSB-AF. Entering an Fn macro key will trigger its message and you will see the RTTY string echoed in the Digital Interface Window as they are transmitted. You can enter free test in the Digital Interface Window by first clicking the Tx button, and then typing into the input edit window. When you are done typing, click Rx or enter] and N1MM+ will go back into listen mode. Clicking Esc:Stop in the main window or hitting the ESC key will halt transmission immediately.

HRD Interface Guide

You will first need to know your WKUSB-AF's comport assignment, if you don't know it, follow the WKscan Tutorial on page 6 or use Device Manager as described in Appendix C (page 40). Be sure to close WKscan after using it.

Start HRD and click on the DM780 icon in the upper left corner of the application. If it's not visible, you may have to enable program view as show on the left. (View->Program)
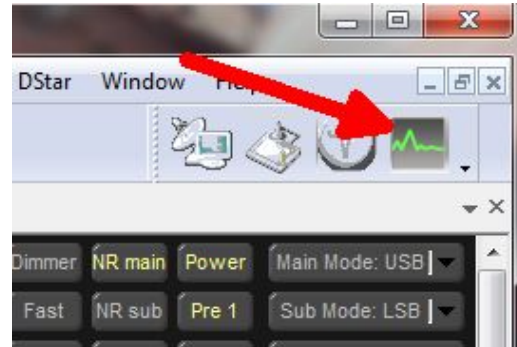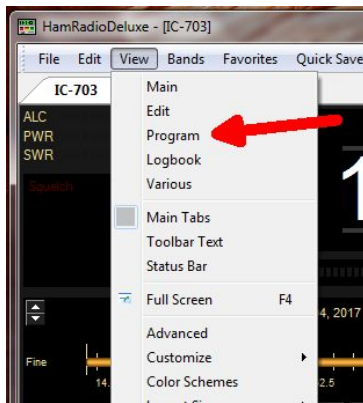
DM780 takes a few moments to start up, once it does, select CW mode from the mode list and then from the CW tab, select CW(Winkey):
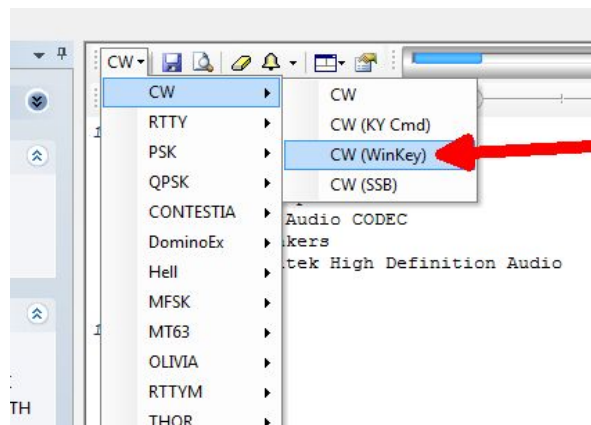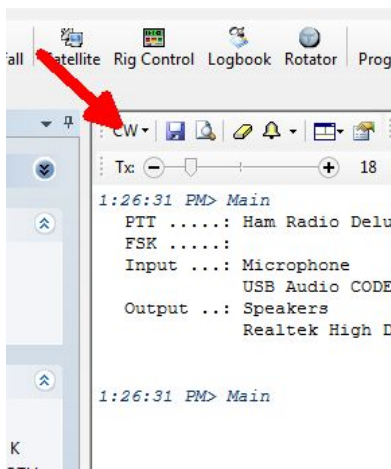
This brings up the WinKeyer dialog box which has multiple tabs with many configuration choices. We will start with the Keyer Tab where we set WK's com port, and then click Connect.

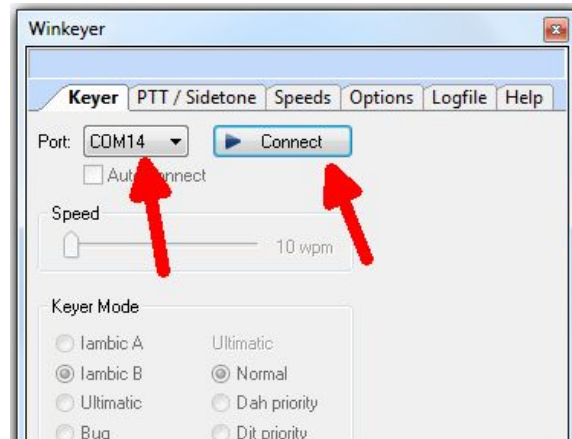Now Connect changes to Connected and items in the dialog box are no longer grayed out. If you click Auto-connect, DM780 will re-connect to WinKeyer automatically on startup.

A slider allows you to change WinKeyer speed while you are in this dialog box for testing. Keyer mode displays a choice of paddle input options. Iambic B is the most often used so that is the default. If you select Ultimatic, three different variants for this mode are enabled

Autospace applies to paddle input only while contest spacing applies to message sending. When contest spacing is on, wordspaces are reduced to 6 dit times vs. standard 7 dit times.

Select paddle echoback and paddle entry will be displayed in with message sending. When paddle watchdog is on, transmit is disabled after 128 consecutive dits or dahs.

Paddle swap allows you to select dits on the right paddle instead of left.

Serial loopback must be on in order for DM780 to keep track of outgoing Morse.

If you click on Test, a canned message will be sent with status displayed.
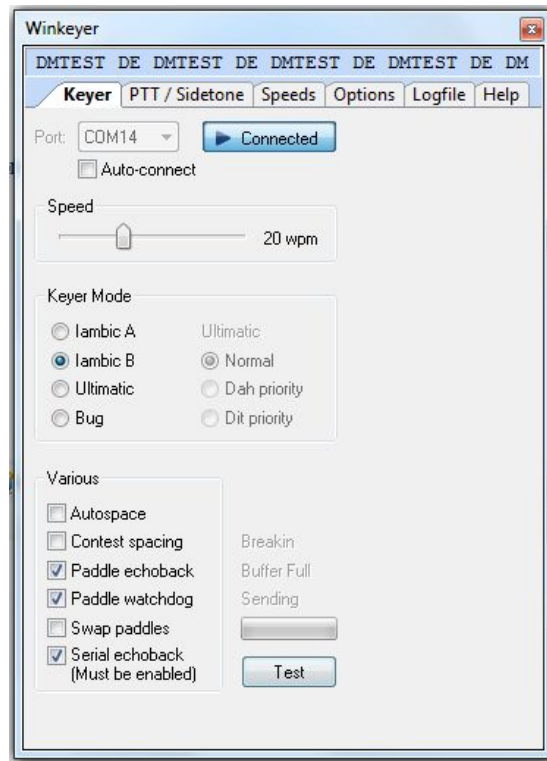
If you click on the PTT/Sidetone tab a list of hardware options are displayed. WinKeyer1 settings are historical and can be ignored. Even though WKUSB-AF uses a WinKeyer 3 IC, the WinKeyer2 settings are the ones we are concerned with. This is where you setup the output port operation. If you want a single key port with PTT, click PTT enable and PTT Port 1. If you want two separate output ports don't click any of these options.

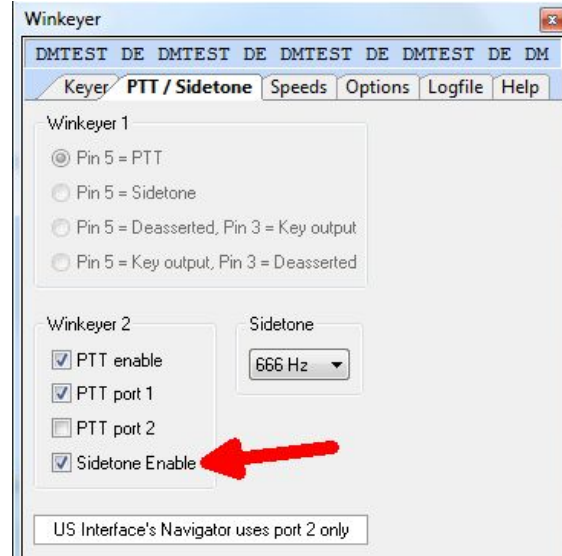A range of sidetone frequencies are available. Note that sidetone enable can be unchecked if you would rather disable it and use your radio's sidetone instead.

The Speeds Tab is mostly concerned with speed pot settings. You can change speed pot swing by adjusting its minimum and maximum values.

Farnsworth can be enabled if desired, but for practical purposes, it is used mostly for code practice and doesn't apply to normal amateur radio exchanges. But it's there if you want it.

Note that the WinKeyer IC version is displayed at the top of this dialog box.
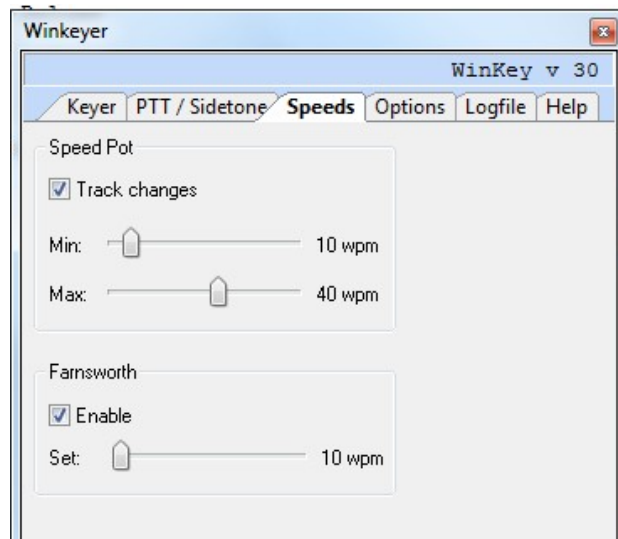
The Options tab contains settings to adjust the timing of the Morse sent by WinKeyer. These settings are discussed in detail in Appendix D of this document.

Also listed are settings for PTT timing. PTT turns on before sending starts and shuts off after sending is complete. Lead specifies how long PTT will turn on before keying starts. The Tail delay specifies how long PTT will stay on after keying ends. There is a separate tail delay setting for paddle sending called hang delay. Most operators want PTT to not drop out between letters but also drop out very soon after they stop sending. Hang time sets the delay in wordspace times making it proportional to sending speed.
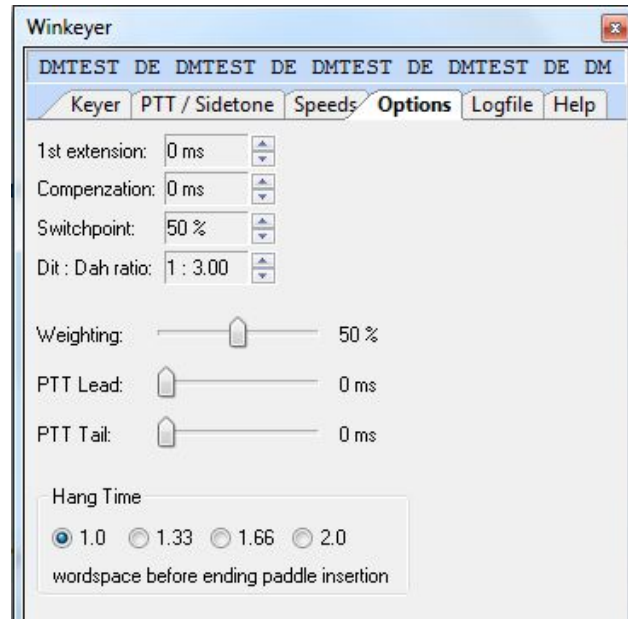
Now WKUSB-AF is ready to work with HRD/DM780. It is a very complex and multi-featured application and I am only able to cover just the WKUSB-AF interface portion. It will take some time to get familiar with the application.

N3FJP AC Log Interface Guide

Start AC Log. From the main application page, select the Settings->Transmit->CW Setup tab.
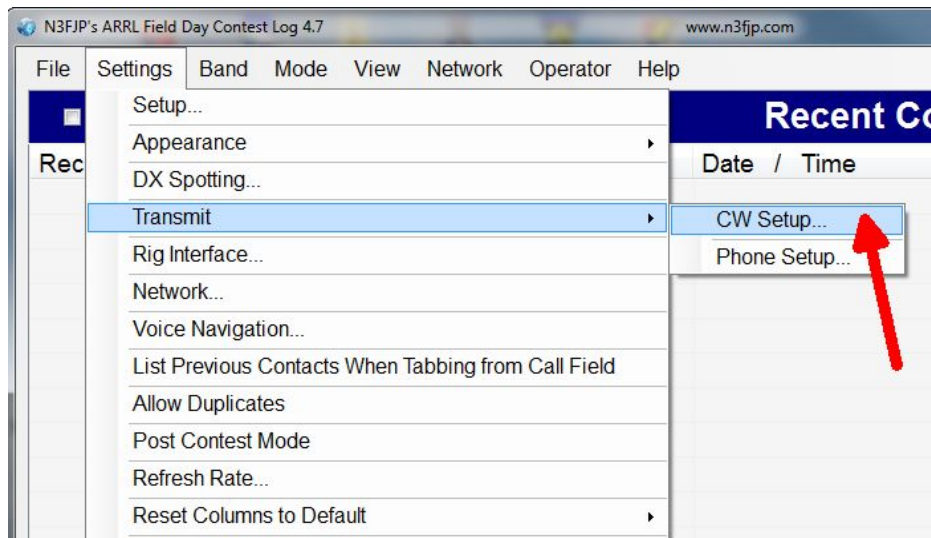
That will bring up the CW Setup Form shown below. A list of the active com ports are shown, click on the one WKUSB-AF is assigned to. Next select WinKeyer and then click Configure WinKeyer.

This brings up a form that presents the WinKeyer options that can be modified. First thing to do is click Open. That connects AC Log to WKUSB-AF, the status will change to Open and WKUSB-AF's firmware revision will be displayed. Most of the options should be familiar, any that aren't can be looked up in Appendix E. Sidetone can be turned off by unchecking Sidetone Enable if you would rather use your radio's sidetone.
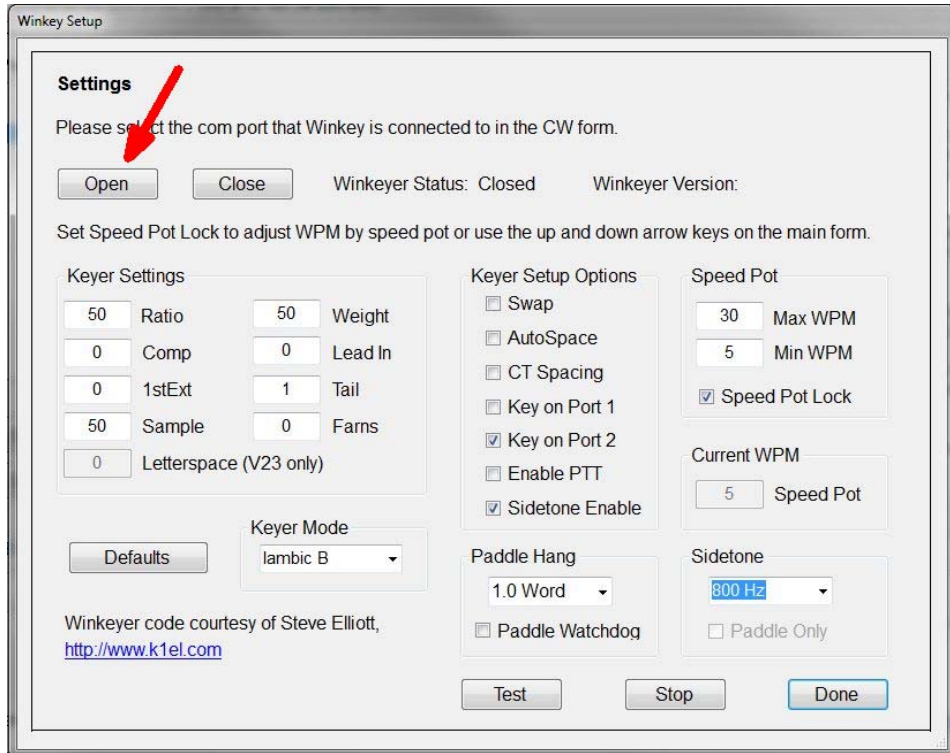
Click Done and you will return to the CW Setup form, you can enter messages in the F1 through F11 slots. F1 is set up as a repeating message with adjustable loop delay. When you click Test, the message in slot F9 will be sent. Click Save Settings to save you configuration and then click Done to return to AC Log where you can start using WKUSB-AF on air.
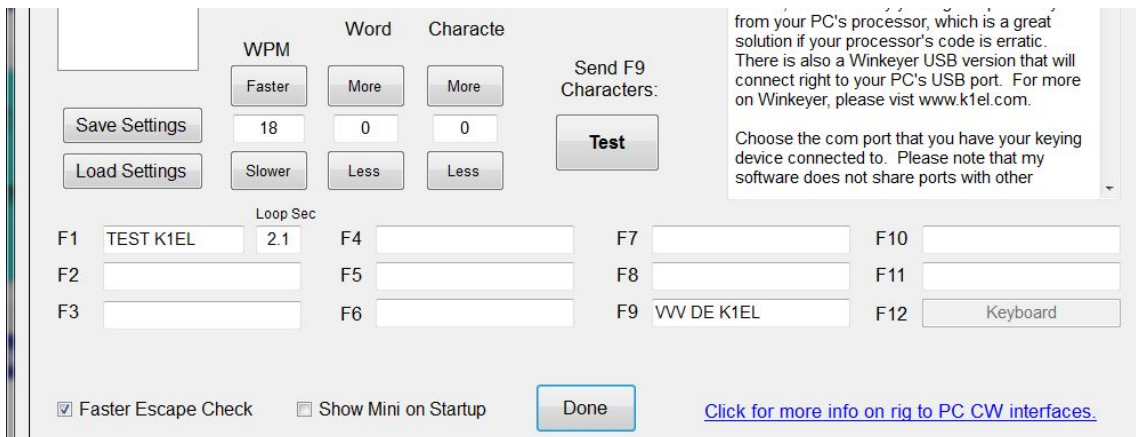
MRP40 Interface Guide

This is one of my favorite applications to use with WKUSB-AF. It's not a logging program but instead it's a conversational, Morse terminal. It has an excellent CW reader that uses your PC's sound card to perform DSP tone detection. CW reception is displayed in a multi-channel, sideways, waterfall display. The best part is that it includes a WinKeyer interface allowing you to send as well as receive. It's very easy to hook up WKUSB-AF. From the main page, click:

Options->Tx Settings->Send via Winkeyer
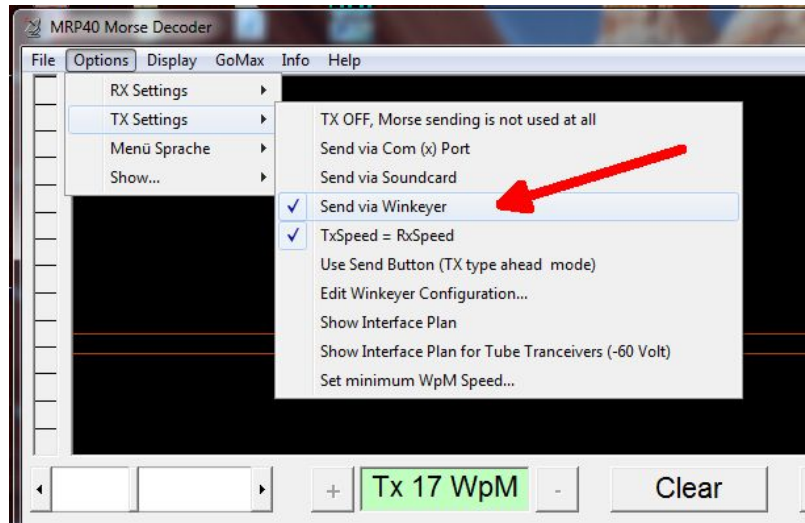
A simple WinKeyer configuration dialog box is displayed. Enter the com port number as reported by WKscan and click Save. WKUSB-AF is now active and you can set up configuration options. Select paddle Keyer Type, Iambic B is most commonly used. Then if you want to use a single key port with PTT select Key Port 1 and PTT, else select either Key Port 1 or Key Port 2 leaving PTT off.

Click and hold Tune and WKUSB-AF will key its output until you release. To get back to the main application click on the X in the upper left corner of the dialog box.
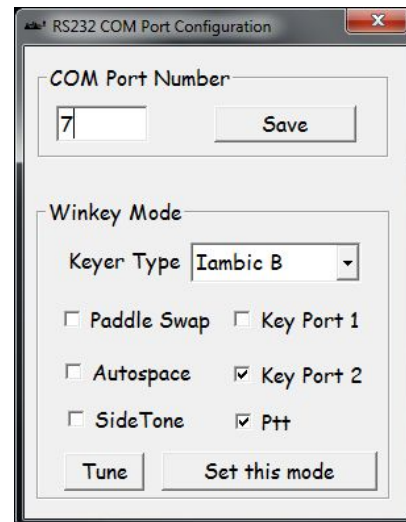
Now simply type in the lower text window and Morse will be sent by WKUSB-AF. I recommend setting TxSpeed=RxSpeed in TX Settings, that way if you copy someone you want to work, your transmit speed will be set to match theirs automatically. I also like to adjust the text size to display more lines. That option is under Display.

FLdigi Interface Guide

This is another very powerful amateur radio application. It's quite easy to set FLdigi up to use WinKeyer in either CW or RTTY mode. If you don't know the comport that your WKUSB-AF is connected to, use our WKscan app. To start, we need to setup WinKeyer configuration:
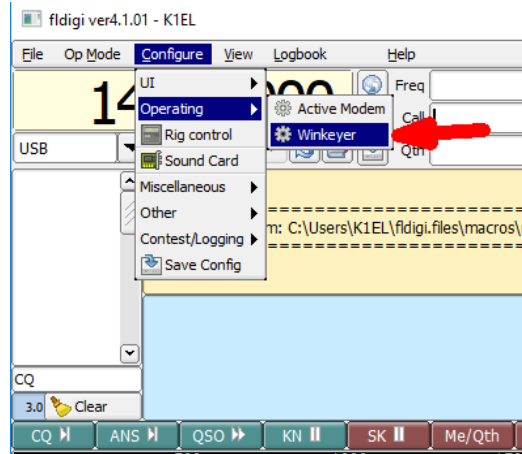
Selecting WinKeyer will bring up the following configuration dialog box,
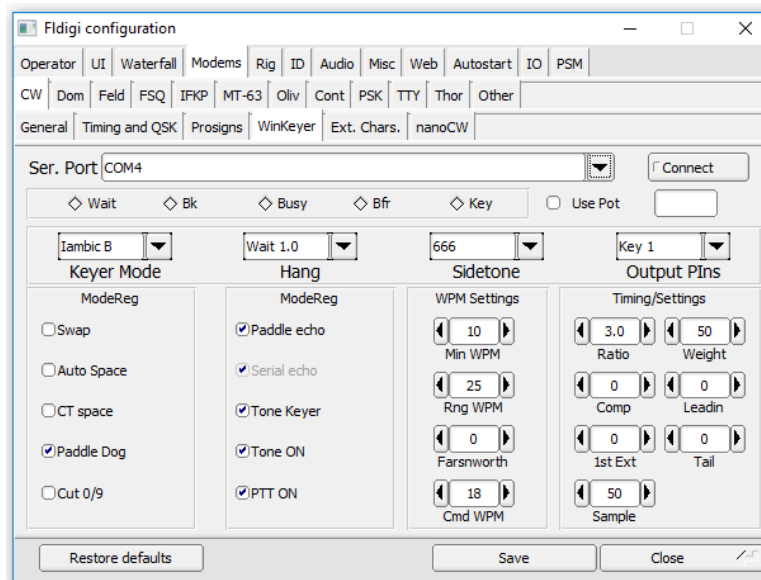
Enter WKUSB-AF's com port and the review the default WinKeyer settings. For the most part, the defaults will suffice but you may want to change paddle swap or use a different keyer mode. Click on Connect and FLdigi will open up the WK interface. The following message will displayed on the main application window to indicate that the interface is open and ready to operate:
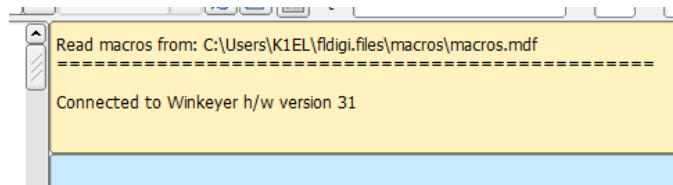
Once WinKeyer has been selected and configured, FLdigi will open it automatically when is starts next time so you won't have to go through this setup procedure every time.

Running WK3 with FLdigi in RTTY mode

Since WKUSB-AF has a WK 3.1 controller IC, you can run it with FLdigi in RTTY mode. Bring up the configuration dialog and select the TTY tab and the WinKeyer 3 tab:
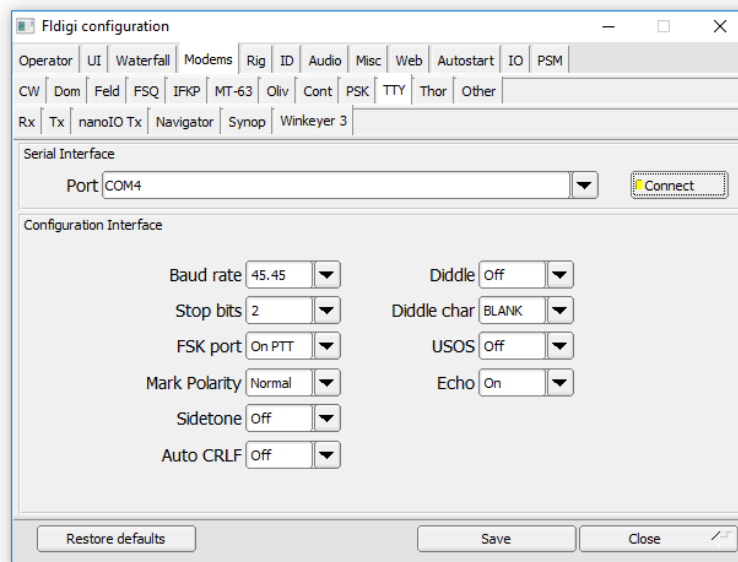
All of WK's RTTY settings are presented here. Select the correct com port and adjust the defaults as required then click connect.

To set RTTY mode in the main operating window, simply choose Op Mode followed by RTTY at the speed you want to run at. WK RTTY operation, like WK CW, is well integrated into FLdigi and once set up you should be good to go. FLdigi offers excellent documentation on how to run their application and that is the best place to go if you need help getting started.
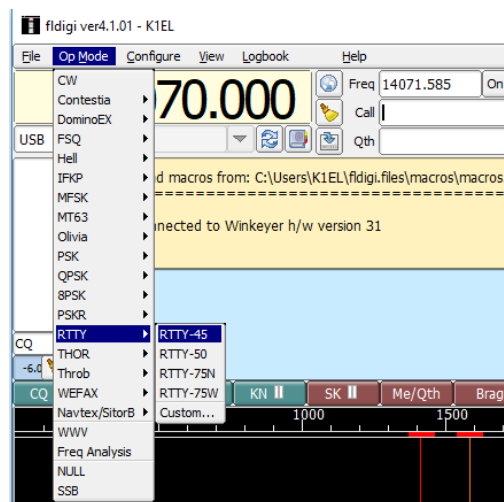
Running WK3demo Version 4.2 in RTTY mode

An easy way to test WKUSB-AF RTTY is to run it with WK3demo 4.2. Since WKdemo only handles RTTY transmit, you will need a means to read and decode received RTTY signals. MTTY or FLdigi are good options as well other PC based RTTY decoders. Many amateur radio transceivers have RTTY decode built in which would make this very easy.

WK3demo 4.2 has an RTTY menu that is enabled when you connect with a keyer that has a version 3.1 WK IC. This is what you should see when you open WKUSB-AF and it has WK3.1:
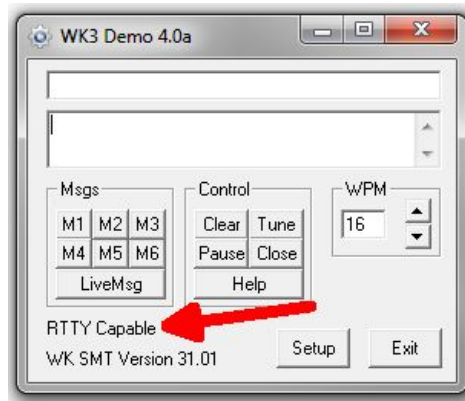
Now if you click on setup you will be able to access the RTTY controls:
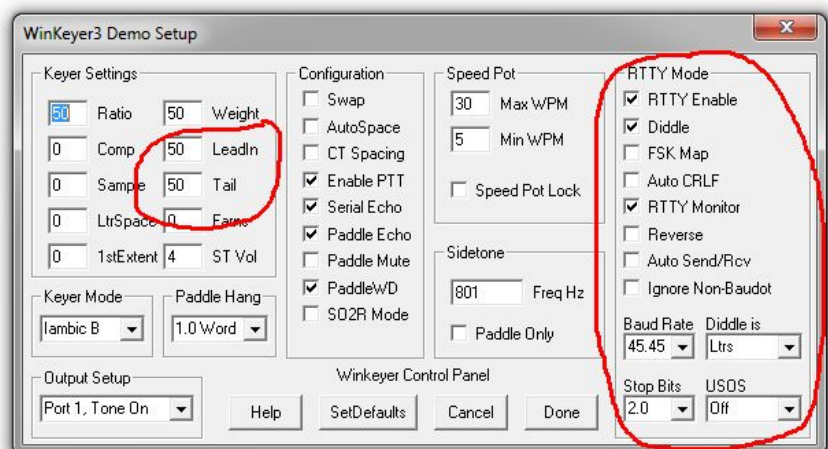
These are the recommended settings that will satisfy most setups. First of all, select RTTY Enable, Diddle, and RTTY monitor. The only other settings that you may have to adjust in this group are FSK Map and Reverse. FSKMAP will swap the PTT and KEY Outputs. Normally WK's KEY out goes to your transceiver's KEY input and PTT goes to the transceiver's FSK input. FSKMAP will swap these two outputs. We will run in direct FSK mode, not AFSK so we connect to the digital FSK input not a mic input. Selecting RTTY monitor will echo letters to the transmit window as they are being sent.

Be sure to enter values for PTT Lead In and Tail. This will insure that the beginning and end of your RTTY transmissions do not get truncated.

Click on Done to get back to the main menu. Once RTTY enable is set, WK will act quite differently. It will not send CW, only RTTY. In addition, the CPO will output sinewave RTTY mark

and space tones. Let's send a test CQ, first we enter a [in the entry window. This will key your transceiver and assert the FSK output to the mark state. Since we selected Diddle on, it will start sending blank characters until we actually enter some text.
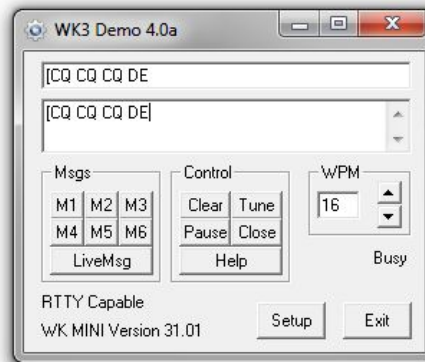
As we enter text it will be echoed to the upper window after it is sent. When you are finished, enter a trailing] which will unkey the transceiver so you can listen for a reply. It may be helpful to monitor your transmission to be sure you are operating correctly. If KEY and PTT are swapped you will hear the transmission clipped on and off since it is being keyed by FSK. When it's set up correctly the transmit output will be on all the time and be frequency shift keyed by FSK. Note that it might be necessary to toggle Reverse. The best way to tell is to listen to other stations and set Reverse to match their shift polarity.

If you want to insert a CRLF sequence enter the } character. Two other actions you may need are Pause and Clear, both of these work by clicking the appropriate button. Tune will key the transceiver with no FSK.

Now let's enter an RTTY message. On the WK3demo main dialog box, move the cursor over a message button and right click on it. The following dialog box will be presented:
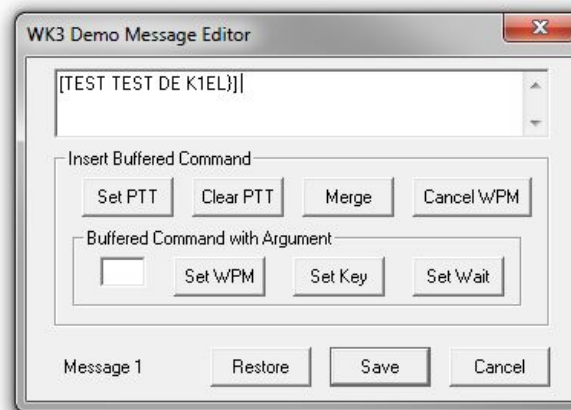
We start an RTTY message with [and end it with]. Note that a CRLF token } is included at the end just before the] so that a CRLF will be sent at the end of the message. Save the message and now when you left click the message button your message will be transmitted.

Interfacing WKUSB-AF to other applications

The basic procedure will be the same for other apps that use WinKeyer. Some apps first require a port mapping procedure like N1MM or HRD while others take you directly to the WinKeyer interface dialog box where you select the port and set up the configuration. If you run into any problems, first contact the application developer since they will be able to offer the best help. If that is not possible post an inquiry on the K1EL Yahoo Group. Odds are that there is someone else in the group who has used the app and they can give you first hand advice. If no one picks up the trail, we will jump in and do what we can. Note that we have a large list of apps that support WinKeyer on our website.

Support and Warranty

WKUSB-AF is fully warranted to the original purchaser against defects in materials and workmanship for one year after purchase. This warranty does not cover damage caused by accident, improper care, or lightning damage. Please contact us before returning your WKUSB-AF for repair and we will issue an RMA. Please submit questions by e-mail to:

k1el.kitsinfo@gmail.com

Watch the K1EL website for updates and new products: https://www.k1elsystems.com

Revision History
```
2-20-22    –      Revision 1.0
```

Appendix A - WK Version 3.1 Firmware Description

WKUSB-AF ships with the latest 3.1 firmware. Besides many bug fixes, several new features have been added to 3.1. The most notable is the ability to generate an accurately timed RTTY serial transmit stream. RTTY can be generated in host mode and also standalone mode. This is especially useful for dealing with transceivers that only provide RTTY receive capability. Using WK 3.1, you can generate RTTY without a PC using a paddle as input.

RTTY Feature Set
WinKeyer supports standard 5 level Baudot transmission. Four operating speeds are supported, 45.45, 50, 75, and 100 baud with either 1.5 or 2 stop bits. Optional diddle generation, automatic end of line carriage return/line feed and UOS transmit are supported.

RTTY Host Mode
The RTTY API is described in the latest WK3 Datasheet. It is fairly easy for an application developer to add RTTY capability to an existing application that already supports WK since all of the basic interface procedures are the same. That includes buffer full handshaking, pause, immediate abort, and transmission of ASCII characters in Baudot (instead of Morse) with automatic FIGS/LTRS generation.

RTTY Standalone Mode
The user issues special paddle commands to initiate RTTY mode and also to control Rx/Tx change over. Since WK can keep track of line length and automatically word wrap at the end of a line, it makes it easy to send Baudot strings by paddle. There are special Morse commands to start and stop an RTTY transmission, as well as insert a CRLF or send an RY test string. In addition, all of WK's standard message slots are available for RTTY as well as the User callsign slot. Since WK has two separate user setups, one setup could be used for CW and the other for RTTY. That allows you to switch back and forth very easily.

WK3 RTTY standalone commands

ADMIN Commands: Press CMD button, wait for R, and then enter X followed by:

Y – RTTY Mode toggle, WK responds with A when RTTY is enabled and an N when turned off.

K – Reverse toggle, this reverses the sense of mark and space. WK responds with an A when reverse is turned on and an N when turned off

L – Diddle toggle, WK responds with an A when enabled and an N when off.

Z – FSK MAP toggle. WK uses PTT and KEY to transmit RTTY. When in transmit mode, PTT is used to key the transceiver and KEY is the FSK output. When FSKMAP is enabled the function of KEY and PTT is reversed; KEY keys the transceiver and PTT is the FSK output.

Procedural commands: when WK is in RTTY mode these commands can be entered at any time, a pushbutton press is not required.

AA (di-dah-di-dah) toggles WK3 in and out of RTTY transmit mode. When transmit mode is enabled the transceiver is keyed and FSK is asserted as specified by the Reverse and FSKMAP settings. If Diddle is enabled it will start and stay on as long as RTTY is enabled.

MM (dah-dah-dah-dah) Tells WK to generate a CRLF sequence.

MN (dah-dah-dah-dit) Tells WK to generate an RY string for testing. A CRLF is generated followed by a line full of repeating RYRYRY patterns.

WK will send messages in Baudot format when RTTY is enabled. WK will automatically go into RTTY transmit mode before the message is sent and leave RTTY mode after the message is complete. Special RTTY codes, MM and MN, can be inserted into messages if desired. AA can't be entered since it is used to end message entry. A message is aborted by pressing the paddles.

Note that the RTTY settings are preserved by using the save command.

Using WK3demo in RTTY mode

This is covered in detail starting on Page 34.

Other Changes for Version 3.1

K1EL WK Applications

There are two new app releases to support WK 3.1.

WK3tools 5.3 is required for upgrading WK3.1 keyers. ALSO, You will find that RTTY setup has been added to the WK3tools standalone utility.

WK3demo 4.2 has upgrades to support RTTY mode. You can enter messages and configure host mode RTTY transmit operation.

Sidetone Volume Command
A somewhat limited volume control has been added to WK3.1.

First Extent Command
This command was removed from Version 3.0 but has been reinstated for Version 3.1

Serial Number Changes
The serial number can be preserved in Version 3.1. The current value will be written to EEPROM when a save command is issued. Serial numbers up to 50,000 are now allowed.

Report Revision Standalone Command
A new extended standalone command has been added that reports the keyer major and minor version. Enter **X Q** and WK will report the major and minor revisions in Morse.

Many Bug fixes
A large number of small bugs were fixed in version 3.1. For example, all illegal paddle input is now flagged and reported. In version 3.0 there were cases where a standalone command entry error could cause a watchdog timeout.

WK API changes
Besides the new RTTY commands several new commands were added to load and unload the RTTY, Volume and X2MODE values.

Appendix B - Installing the CH340 USB Driver for WKUSB-AF

As mentioned, don't plug WKUSB-AF in yet, we have to deal with the USB driver first.

For Mac or Linux, there is detailed installation info can be found here:
https://learn.sparkfun.com/tutorials/how-to-install-ch340-drivers/all

For Windows, first thing to do is create a folder on your PC called CH340 or something similar, the name doesn't really matter. Then download CH341SER.exe and Windows-CH340-Driver.zip from https://www.k1elsystems.com/WKminiRevB.html and copy them into the new folder. Unzip the Windows-CH340-Driver.zip file.

Now plug WKmini into your PC and wait to see if Windows Update will automatically search for a USB driver. This usually doesn't happen, but if it does click Skip obtaining driver….
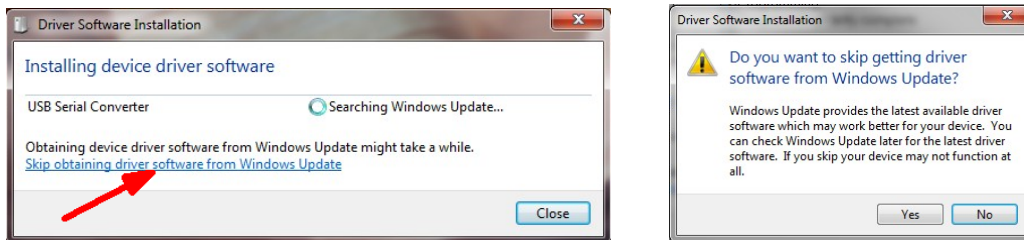
Windows will then look on your local hard drive for the driver and will install it automatically.
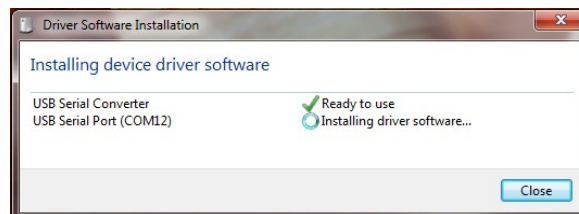
If Windows Update did not start, simply run CH341SER.exe and this dialog box will be displayed:
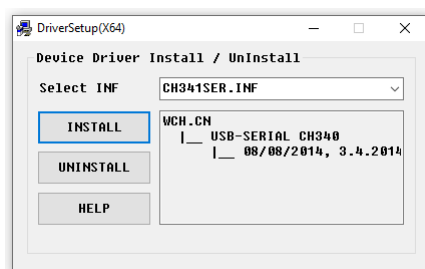
Click Install and after a few minutes, the following message box will display:
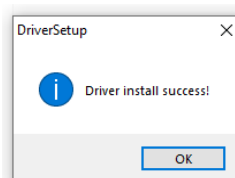
Now return to page 5 to continue WKUSB-AF hardware setup and testing.

Appendix C - Determine WKUSB-AF's COM port when you don't have WKscan:

Left click on Start and then right click on *MyComputer*:

Click on *Properties*:

Click on *Hardware*:

Click on *Device Manager*:

Then Double Click on *Ports (COM & LPT)* to expand the port list:

The port labeled *USB Serial Port (COMx)* is the WKUSB-AF port:
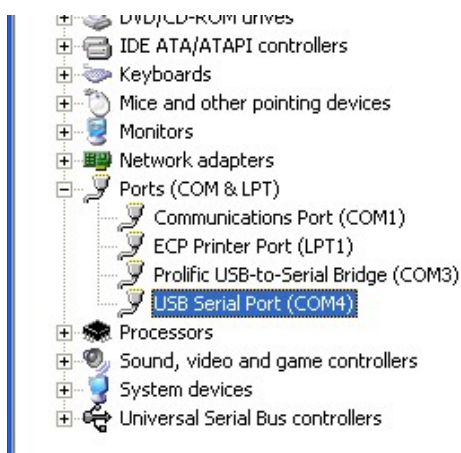
If you want to change the port assignment, double click on USB Serial Port and

Then Click on *Port Settings*:
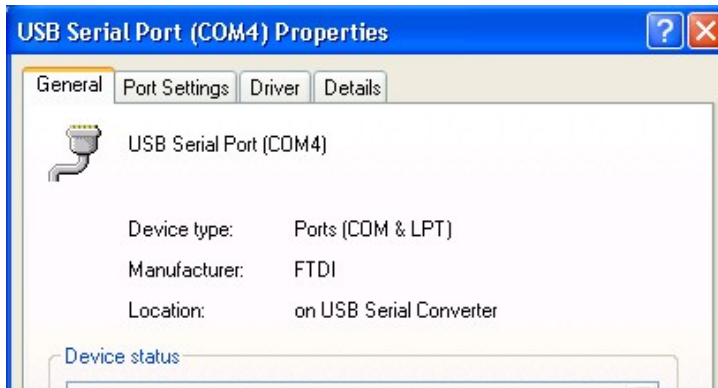
Click on *Advanced*:
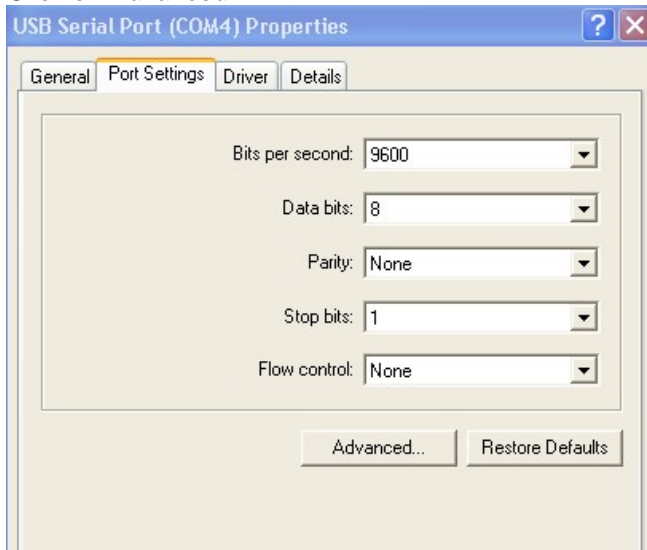
Finally you will see the port selection control:
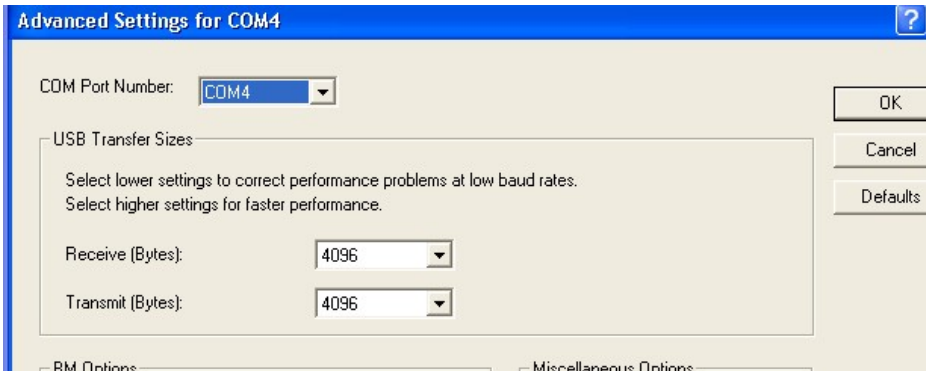
By clicking on the down arrow to the right of the COM Port Number List Box you will get a list of available com ports. Click on the one you want and you are finished.
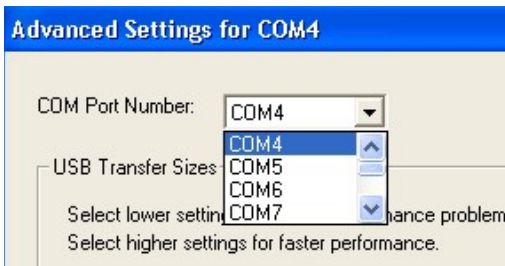
In the event you installed a new USB com port or reassigned a different port the com port you last used may not be available. In that case you will need to map WKUSB-AF to a new com port.

Appendix D – WinKeyer Settings Defined

Weighting
A proportional amount is either added or subtracted from the length of all transmitted dits and dahs. A value of 50 selects no weighting adjustment. Values less than 50 reduce weighting and values greater than 50 increase weighting. Note that weighting does not affect sending speed because any increase in keyed time is subtracted from spacing time. Reduction in weighting results in a thinner sounding keying while increased weighting results in a heavier sound. Since weighting tracks speed, a given weighting will sound the same at all speeds.

Decreased Weighting

Normal R

Increased Weighting
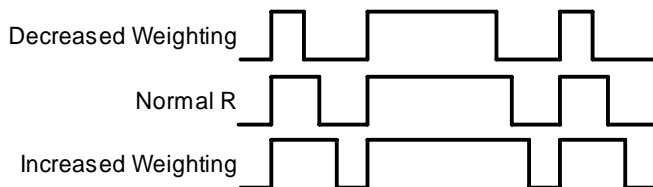
WEIGHTED DIT TIME =
(((WEIGHT×50)/50)×DIT TIME)

WEIGHTED DAH TIME =
(((WEIGHT×50)/50)×DAH TIME)

Weighting Example

Keying Compensation
A fixed amount is added to the length of all dits and dahs. QSK keying on modern transceivers can cause shortening of dit and dah elements which is especially noticeable at high speeds. WK3 can compensate for the shortening by adding a uniform length to each dit and dah element. The

adjustments are made in one-millisecond steps and the maximum adjustment allowed is 250 mSecs. Key compensation is very similar to Weighting in that an adjustment added to a dit or dah is then subtracted from the spacing between them so the overall speed is not changed. The difference between weighting and compensation is that compensation is independent of speed, so if 10 mSec of key compensation is selected, 10 msec will be always be added regardless of speed. Be aware that at high speeds, large values of key compensation can reduce inter-element space to zero.

In the following figure, an R is represented without key compensation (nn = 00) and with 12 mSecs of key compensation (nn = 12). Note that each dit or dah starts at the same place even with compensation on.
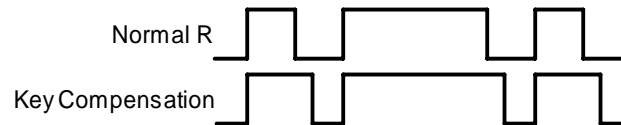
Keying Compensation Example

Letterspace
Specifies an additional letterspace, in percentage of a wordspace, to be applied between all letters. Minimum adjustment is 0% while maximum addition is 62%.

Paddle Sample Delay
This setting controls when WKUSB-AF will start looking for a new paddle press after sensing the current one. If there is not enough delay, the keyer could send unwanted dits or dahs. If there is too much delay, the operator is held back because they can't paddle ahead. The default value is one dit time (50) and is adjustable in percent of a dit time. A setting of 55 – 65 may seem familiar to Accukeyer users If the paddle sensitivity is set to zero, dit and dah paddle memory is disabled.

The delay is calculated with this formula:

`DELAY_TIME = (SWITCHPOINT×DIT_TIME)/50` where Switchpoint is a value between 10 and 90.

Dit/Dah Ratio
The ratio between the length of a dit vs. the length of a dah is controlled by this setting. The agreed upon standard ratio is 1:3 (dit:dah), a dah is three times longer than a dit.

The formula to determine dah/dit ratio is: **DAH/DIT = 3∗(nn/50)**

 A value of 50 selects 1:3, a value of 33 would select 1:2, and a value of 66 would select 1:4. Any value other than 50 causes distortion of the Morse waveform. Some ops use this option to make their CW sound less "machine like" but a little goes a long way !
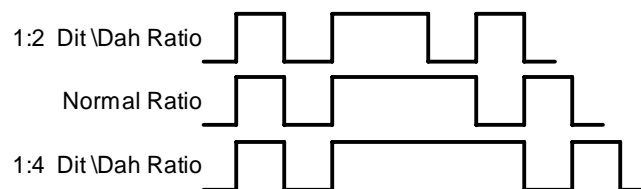
Three ratio settings for the letter R

PTT Lead and Tail Times

WKUSB-AF's PTT output is used to control an accessory device in addition to normal CW transmitter keying. In most cases this device is a linear amplifier but it could be an antenna changeover relay or PTT input of a transmitter. In all of these cases there are delay requirements that must be met to insure that the accessory device is switched on before transmission begins and is held on until transmission completes. This prevents damage to the accessory device due to hot switching. WKUSB-AF provides three independent PTT delays to meet this requirement.
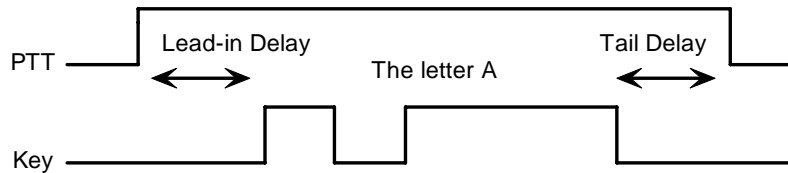
PTT Lead-in and Tail Example

The first of these is Lead-In delay which initiates a keying event. When transmit starts, PTT will be asserted first and then, after the Lead-In delay expires, the key output will be asserted. Lead-In can be set to a value from 0 to 250 milliseconds in 10 millisecond increments.

Tail Delay specifies the amount of time PTT will be released after Key is released. Like Lead-In delay, it is adjustable from 0 to 250 milliseconds in 10 millisecond increments.

In setting tail delay there are two considerations, the first is to prevent hot switching and the second is to add delay to fill in between letters sent by paddle. Most ops don't want their amplifier to be switched in and out between letters. An issue arises in that Tail delay is not solely proportional to sending speed. This is problematic when someone sets a comfortable tail delay at a slow speed and then increases the speed to a much faster rate. At the faster rate, the tail delay will hold too long after keying stops. Alternatively, if a comfortable delay is set at a faster WPM rate, PTT will then drop out between letters at a slower speed. To solve this problem, paddle tail delay is governed by a separate parameter called hang time.

Hang Time is adjustable in four steps and is measured only in wordspace and dit times rather than fixed milliseconds. This means that the delay will accurately track sending speed. In other words it will decrease automatically as sending speed increases and vice versa. The four settings for Hang Time are:

        HangTime = 0: wait 1 wordspace + 1 dit before ending paddle insertion
        HangTime = 1: wait 1 wordspace + 2 dits before ending paddle insertion
        HangTime = 2: wait 1 wordspace + 4 dits before ending paddle insertion
        HangTime = 3: wait 1 wordspace + 8 dits before ending paddle insertion

To sum up, Tail delay should be set to provide just enough delay to prevent hot switching for machine sent CW while hang delay is designed to hold PTT between letters independent of sending speed. PTT will automatically be held between machine sent letters and words but will drop out quickly after a message has been sent as long as there is no trailing wordspace.

Paddle Watchdog

A paddle watchdog counter disables keying output after 128 consecutive dits or dahs. WKUSB-AF assumes that this is an unintended condition. This watchdog can be disabled if desired.

Paddle Swap

This bit swaps the operation of the dit and dah paddles. It's used mostly to accommodate left handed ops or to correct for a miswire in a keying cable.

Paddle Only Sidetone

Local sidetone is generated only for paddle generated Morse, sidetone is inactive when text is sent form the host.

Paddle Mute
This locks out the paddle so that it will not generate Morse when pressed.

Paddle Echo back
When enabled, all characters entered on the paddles will be echoed back to the host.

Serial Echoback
Serial Echo Back tells WKUSB-AF to echo each Morse letter that originated at the host. It can be used to allow a host application to stay exactly in sync with Morse letters as they are sent. Each letter is sent to the host after it has been sent in Morse. This permits the host to track WKUSB-AF's progress in real time. Note that buffered commands and their parameters are not echoed back to the host.

Autospace
Autospace works in the following way: If you pause for more than one dit time between a dit or dah, WKUSB-AF will interpret this as a letter-space and will not send the next dit or dah until the full letter-space time has been met. The normal letter-space is 3 dit spaces. WKUSB-AF has a paddle event memory so that you can enter dits or dahs during the inter-letter space and WKUSB-AF will send them as they were entered. With a little practice, autospace can help you to send better.

Contest Spacing
When contest spacing is enabled, wordspace time is reduced by one dit. Instead of seven dits per wordspace, contest spacing selects six dits per wordspace. The idea is that exchanges can be sent faster with reduced space between words.

Paddle Keyed Modes
There are six different keying modes supported by WKUSB-AF: Iambic mode A, Iambic mode B, Straight Key/Bug, Ultimatic, Dit priority mode, and Dah priority mode. In either iambic mode, alternating dits and dahs are sent while both paddles are held closed. In mode B an extra alternate dit or dah is sent after both paddles are released. In straight key/bug mode a dah paddle press will key the transmitter for as long as the paddle is pressed and dits will be generated automatically when the dit paddle is pressed. In Ultimatic mode, when both paddles are pressed, the keyer will send a continuous stream of whichever paddle was last pressed. Hold dah then press dit->sends dits. Hold dit then press dah -> dahs are sent. Ultimatic dit and dah priority mode will generate dits and dahs automatically in response to single paddle presses, but when both paddles are pressed either dit or dah has priority.

Speed Pot Lock
This tells WKUSB-AF to ignore host speed control and to instead lock the current sending speed to the speed pot.

Speed Pot Minimum and Maximum
The limits for the speed pot can be specified. The minimum must be greater or equal to 5 while the maximum must be less than or equal to 99 WPM. This allows the speed pot to be tailored to the user's needs.

Sidetone Enable
Sidetone can be turned on or off. Local sidetone is not needed if the user is going to use the sidetone generated by the transceiver they are using. This is the case in the majority of setups.

Sidetone Frequency
WKUSB-AF's local sidetone frequency is adjustable in fine increments from about 500Hz to 4000Hz

Appendix E WKUSB-AF (WK3) Standalone Mode

Standalone Command Mode (WKUSB-AF is not connected to a PC application)

If the command push-button is pressed and held, WKUSB-AF (WK) will respond after about two seconds with the letter *R* in sidetone only. This means WK is ready to accept a command, you simply enter the command letter in Morse on the paddles and the command will be executed. Some commands require additional information which WK will prompt you for by outputting the letter *E* (for enter). All commands provide some sort of feedback to tell you if the command was understood and executed properly, in most cases either an *A* or *R*. If an illegal command is entered WK will respond with a question mark.

Important Note ! When in command mode, transmitter keying is disabled and replies are sent in sidetone only. Thus in order to use command mode you must have a sidetone speaker of some sort. If sidetone had been disabled with the **A** command it will be re-enabled automatically when entering command mode.

Standalone Command List

A -	**Select sidetone on or off**	**N -**	**Load 4 Digit Serial Number**
B -	**Start Easy Beacon**	**O -**	**Select output key port**
C -	**Set Command Speed in WPM**	**P -**	**Start Practice Mode**
D -	**Decrement Serial Number**	**Q -**	**Query current settings**
E -	**Swap message banks**	**R -**	**Review message without transmitting**
F -	**Set Farnsworth Speed**	**S -**	**Set preferred WPM speed**
G -	**Set 50% Duty Cycle Tune**	**T -**	**Load PTT Tail Time**
H -	**Set Paddle Hang Time**	**U -**	**Select Autospacing on/off**
I -	**Set Letterspace Adjust**	**V -**	**Set Keying compensation in mSec**
J -	**Set Paddle Sensitivity**	**W -**	**Set Key Weight**
K -	**Select Keyer Mode**	**X -**	**Select Extended Command Mode**
L -	**Set PTT Lead-In Time**	**Y -**	**Set Dit/Dah Ratio**
M -	**Mute Transmit (CPO mode)**	**Z -**	**Select sidetone frequency**

In the command descriptions below, the **[n]** or **[nn]** notation means that additional parameters must be entered on the paddles after the command.   A letter displayed in **BOLD** is something you enter, ***BOLD ITALIC*** is what WK responds with. A [pb] means that WK will wait for you to press one of the message pushbuttons.

A - Sidetone enable is toggled when this command is entered. Toggle means if the sidetone was on when this command was issued it will be turned off and vice versa. WK will acknowledge this command by responding with an *A (enAbled)*. Note: If sidetone is disabled it will be re-enabled temporarily for command mode.

B [m,dd] - Easy Beacon: Any message slot in the current bank can be repeated at a specified rate. After entering **B**, the user is prompted to enter a single digit message number (1-6) and then prompted to enter a two digit beacon cycle time (1-99) in seconds. A beacon is interrupted by hitting either paddle. The beacon is timed from start of beacon to start of next beacon. For example if you set a delay time of 10 seconds, the beacon will start every 10 seconds regardless how long the message is. If the message is longer than delay time, then there will be no gap between messages.

C [nn] – Command WPM: WK uses different speeds for command transactions and keyed transmit. Changes in transmit speed will not affect command speed. After the **C** command is issued enter the speed in WPM. If the speed is valid, WK responds with an *R* otherwise a *? See the* **S** *command for details on setting transmit Morse speed.*

D - Decrement Serial Number by 1: WK responds with a *D* after the decrement.

E - Swap message banks: WK has two separate message banks, six messages each. In response to this command a single dit is echoed when bank one is selected and two dits when bank two 2 is selected.

F [nn] - Set Farnsworth Speed: This is a used primarily for code practice. Letters are sent at the Farnsworth speed while maintaining the default code speed. For example, if Farnsworth is set to 25 WPM and the operating speed is set to 7 WPM, individual letters will be sent at 25 WPM while spacing between letters is at a 7 WPM rate. To disable Farnsworth mode, set it to zero.

G – Set tune duty cycle default: This sets the default tune duty cycle which can be either 50% or 100%. Some folks prefer a 50% tuning duty cycle to reduce power dissipation in transmitter finals and/or linear amplifier. An *A* is echoed when 50% is enabled, an *N* is echoed when set to 100%. Note that once tune starts, the duty cycle can be changed by hitting the paddles.

H [n] - Set Transmit PTT Hang Delay Time: Sets a PTT delay that is proportional to sending speed. The delay begins after paddle sending stops. You can select one of four delays:

        HangTime = 0: wait 1 wordspace + 1 dit before ending PTT
        HangTime = 1: wait 1 wordspace + 2 dits before ending PTT
        HangTime = 2: wait 1 wordspace + 4 dits before ending PTT
        HangTime = 3: wait 1 wordspace + 8 dits before ending PTT

After entering the command letter, you will be prompted with an *E* to enter the desired hang time as a number 0 to 3 as indicated in the table. Hang delay is different than PTT tail delay in that it is proportional to code speed while PTT delay is a primarily a fixed delay time. This means you don't have to change the PTT delay every time you change sending speed.

I [nn] - Set Letterspace Adjustment: nn is a value 0 to 31, specifying an additional letterspace to be applied between letters. Multiply **nn** by two to arrive at the actual adjustment percentage. For example a value of 7 applies 14% additional letterspace between letters. The maximum adjustment is 62%.

J [nn] - Paddle Sample Point controls when WK will start looking for a new paddle press after sensing the current one. If there is not enough delay the keyer will send unwanted dits or dahs, if there is too much delay it may slow you down because you can't send ahead of the keyer. The default value is one dit time (50) and is adjustable in a fraction of a dit time. Faster operators report a setting somewhat less than default is more pleasing. **If the paddle sensitivity is set to zero, both dit and dah paddle memories are disabled.** The delay is calculated with this formula:

DELAY_TIME = (nn×DIT_TIME)/50 where switch point is a value between 01 and 99.

K - Set Keying Mode: There are six different keying modes supported by WK: Iambic mode A, Iambic mode B, Straight Key/Bug, Ultimatic, Dit priority mode, and Dah priority mode.

After the **K** command is issued the current mode is set by entering a single letter:
        Iambic B: Enter B
        Iambic A: Enter A
        Ultimatic: Enter U
        Straight Key: Enter S (This is also the Bug setting)
        Dit Priority: Enter E (Ultimatic: dits take priority when both paddles are pressed)
        Dah Priority: Enter T (Ultimatic: dahs take priority when both paddles are pressed)

L [nn] - PTT Lead In Time can be set to a value between 0 and 99 which is a subset of the entire possible range of 0 to 255. See the Set PTT Tail description on page 44 for more information.

M - Toggle transmit mute: Mute transmit when you want to use WK as a Code Practice Oscillator (CPO) When muted, WK will send CW in sidetone only. In response to this command, the WK will echo an *A* when mute is turned on and an *N* when mute is turned off.

N [nnnn] - Load 4 Digit Serial Number All four digits must be entered including leading zeroes. The serial number is played by inserting a send message token /N into a message. The serial number is automatically incremented after sending. See *Embedded Command* section for more details.

O - Swap Key Output Port: Each time the **O** command is issued, the key port is toggled back and forth between key port 1 and key port 2. When port 1 is selected, WK responds with a single dit and two dits are echoed when port 2 is selected. If you have two transceivers, this feature frees you from moving cables around when you want to switch radios.

P [m,d] - Start Practice Mode: A dual mode multi-level code practice program is built into WK.There are two styles of practice, receive only and echo (receive/respond) practice. There are four levels of practice organized by easy to difficult letter groups. The four levels are:

Level 1: E T A N I M W S G D U K O R
Level 2: C Q P J F B V Y H X Z L   including level 1
Level 3: 1 2 3 4 5 6 7 8 9 0   including level 1 & 2
Level 4: ? / , . AR SK BT AS   including level 1 & 2 & 3

The syntax for entering a practice command is:

P m d where m is R for receive or E for echo practice and d is a single digit 1 to 4 to select difficulty level.

P R 2 selects Level 2 receive practice
P E 4 selects Level 4 echo practice

Receive Practice Description: Random characters from the selected level are sent in groups of five. Practice will continue until the command pushbutton is pressed.

Echo Receive/Transmit Practice Description: WK will send a random character from the selected level and you must respond by echoing the character back on the paddles. If you get it right WK will repeat the first character followed by a new character. Now you must echo back both characters. WK will continue to add characters until a set of five characters has been completed successfully. After that it will start over with a new set. If you miss a character WK will respond with 8 dits and start over with the same sequence of characters. When you want to end practice, respond with di-dah-di-dah or press the command pushbutton. By default practice will be sent on the active output port, enable transmit mute (M command) to inhibit Tx.

In both modes, the sending speed can't be changed with the speed pot. Note that the new speed will start after the current round of characters has been finished,

A pseudo random letter generator is used to generate the sequence of letters. Normally the seed for the generator is a random number but sometimes it is useful to generate the same sequence over and over during training sessions. Use the P command with an S to set a fixed seed. The format for the command then is P S d where d is a seed value between 1 and 99. A seed value of 0 will return to fully random operation.

Q - Query WK Current Settings: WK will respond with current settings sent in the following format:

WPM	*is sent first*
N	*followed by Serial Number*
M	*followed by free msg memory space in letters available*
C	*followed by command WPM*
W	*followed by weight*
L	*followed by lead time*
T	*followed by tail time*
E	*followed by 1st extent delay*
V	*followed by key compensation*
F	*followed by Farnsworth WPM*
I	*followed by Letterspace adjustment*
J	*followed by Paddle Sample Adjust*
Y	*followed by dit/dah ratio*
B	*followed by speed pot min WPM*
T	*followed by speed pot max WPM*

**R [pb] - Review a message without transmitting:** After **R** is entered, WK will respond with an **E**. Press the message button of the message you wish to send. The message will be sent in sidetone only. If you try to send an empty slot, WK will respond with **MT**. Embedded commands will be sent as is without expansion. In other words /S10TEST will be sent as: **DAH-DI-DI-DAH-DIT 10TEST**

S [nn] - Set Favorite Speed in WPM: a new value is entered between 5 and 99. This speed is activated when the speed pot is turned fully counterclockwise. The minimum WPM is 5 WPM, the maximum speed is 99 WPM. If the speed entered is valid, WK responds with an *R*, otherwise a *?*

T [nn] - Set PTT Tail Time: WK provides a transmitter PTT output that can be used to switch a transmitter or linear amplifier over to transmit mode in advance of actual CW keying. You have control over the time delay between when PTT is asserted and when CW keying will start, this is lead-in. You also have control over how long the transmitter will stay in transmit after keying has stopped; this is tail delay. The tail delay is handled differently for CW sent by paddle and CW sent by message. Paddle delay is controlled by the Hang Time setting while message PTT delay is controlled by the Tail setting. The formula to calculate tail time is:
Tail Delay = Three Dit Times + (Tail Setting *times* 10 milliseconds)

Examples:
   At 20 WPM, Tail set to 7, Tail Delay = (3x60)+(7x10) = 250 mSec
   At 40 WPM, Tail set to 7, Tail Delay = (3x30)+(7x10) = 160 mSec
   At 20 WPM, Tail set to 0, Tail Delay = (3x60)+(0x10) = 180 mSec
   At 15 WPM, Tail set to 55, Tail Delay = (3x80)+(55x10) = 790 mSec
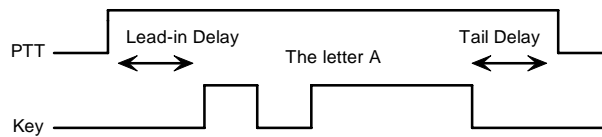
PTT Lead-in and Tail Timing Example

In general we want a very short tail time when sending messages and we want PTT to stay asserted between letters while sending with a paddle set. That's the reason the delay is dictated by two mechanisms.

U - Turn Autospace Mode Off and On When autospace is enabled WK will automatically insert proper inter-letter space between letters. Each time the **U** command is issued WK will toggle between modes responding with an *A* for autospace enabled an *N* for autospace disabled.

Here is how autospace works: If you pause for more than one dit time between a dit or dah WK will interpret this as a letter-space and will not send the next dit or dah until the letter-space time has been met. Normal letter-space is 3 dit spaces but this can be increased by using the **I** command. WK has a paddle event memory so that you can enter dits or dahs during the inter-letter space and WK will send them as they were entered. With a little practice, autospace will help you to send near perfect Morse. If autospace seems to be less forgiving than you like, you can increase the letterspace setting to relax the timing.

V [nn] - Keying Compensation allows a fixed amount of time to be added to the length of all dits and dahs. QSK keying on modern transceivers can cause shortening of these elements which is especially noticeable at high speeds. WK allows the length of the elements to be increased uniformly to compensate for this. The adjustments can be made in one-millisecond steps. The maximum adjustment is 31 mSecs. Key compensation is very similar to Weighting in that any adjustment added to the dits and dahs is subtracted from the spacing so the resulting speed is not changed. The difference between weighting and keying compensation is that compensation is independent of speed, so if 10 mSec of key compensation is selected, 10 mSec will be always be added regardless of speed. So be careful at high speeds with large values of keying compensation, dits and dahs may run together with no spacing at all.

Letter R without compensation

Letter R with compensation
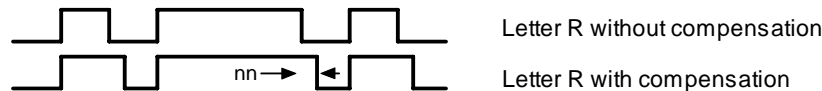
Key Compensation Timing

W [nn] - Keying Weight can be adjusted in percentage from 25% to 75%. When set to 50 % the dit time is equal to the inter-element time, which is normal. Values less than 50 reduce weighting while values greater than 50 increase weighting. Note that weighting does not affect sending speed because any increase in keyed time is subtracted from spacing time. Reduction in weighting results in a thinner sound while increased weighting results in a heavier sound. Since weighting tracks speed, a given weighting will sound the same at all speeds.

25 % weighting

50 % weighting
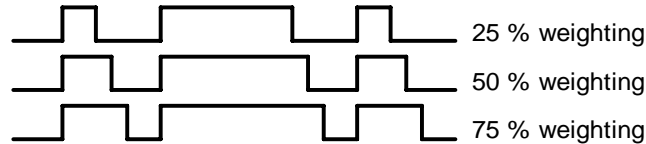
75 % weighting

Three Key Weighting Examples

X - Enter Extended Command Mode
An additional set of commands are located in a sub menu. These are commands that are not entered very often so an additional paddle action is not a significant issue. Extended commands require two entries, an **X** followed by a sub command. Here is the procedure in detail:

Press and hold the command pushbutton and WK will respond with an *R*
Enter an **X** and WK will respond with an *E* (command request)
Enter desired Extended Command with additional parameters, if required.

A list of extended commands is provided in the section titled *Extended Command List* found after this section.

Y [nn] - Set Dit/Dah Ratio: nn ranges from 33 to 66.   Entering *Y 50* sets the standard 1:3 ratio. For example a value of **33** selects a dit/dah ratio of 1:2 while a value of **66** selects 1:4. The ratio formula is:

Ratio of 1:N where N = (nn * 3)/ 50   example nn = 40 give a ratio of 1:((40*3)/50) = 1:2.4

Z - Change Sidetone Frequency: After this command is entered the sidetone oscillator will be keyed at a steady rate with transmit muted. Pressing the paddles will raise or lower the frequency. The range varies nearly continuously from 300 Hz at the low end to 2000Hz on the high end. It takes a while to sweep through the whole range. Pressing the command pushbutton will end this command and store the new sidetone frequency. Like all settings, use the Preserve Settings extended command to save the new sidetone setting in EEPROM.

Extended Command List

A – Pushbutton Input Diagnostic: is used to verify the correct operation of the switch network connected to WK input pin. This design allows four switches and a potentiometer to share a single pin on WK. The state of the network will produce a unique numeric value which WK uses to identify a pushbutton press.

After the Extended **A** command is issued, the numeric network state will be sent in Morse sidetone. For proper pushbutton operation, the input values must fall within the following ranges:

Command PB	0 to 40	PB 3 .... 101 to 130
PB 2	41 to 100	PB 4 .... 131 to 160

To stop this diagnostic state, press and hold either paddle until the readout stops.

C - Toggle Contest Spacing: When contest spacing is enabled, word space is set to 6 dit times as opposed to the normal 7 dit times. An *A* is sent when CT space is enabled, an *N* is sent when it is turned off.

E[nn] - First Element Extension: Forces the first dit or dah of a transmission to be elongated to allow for receive to transmit delay. Enter a value between 0 and 99 mSecs for nn.

F - Toggle Fast Command Response Time: Normally WK will enter command mode when the command pushbutton is pressed for 2 seconds. This may be too long for some operators. When fast response is enabled, the delay time is reduced to 1.3 seconds. An *A* is sent when fast is enabled, an *N* is sent when it is turned off.

G - Report Speed Pot Value: WK will respond with the current speed pot value in sidetone.

K - Reverse Toggle: When in RTTY mode, this will toggle the mark and space states. Normally this can be done at the transmitter but this allows an easy way to do it. An *A* is sent when reverse is enabled, an *N* is sent when it is turned off.

L - Diddle Toggle: When in RTTY mode, this will toggle the automatic generation of diddle which fills idle time during RTTY transmit. It is particularly effective during paddle RTTY. An *A* is sent when diddle is on, an *N* is sent when diddle is turned off.

M - Load Callsign: Use this command to load a stored callsign. It works just like a message entry and the destination is the currently enabled user slot. Each user slot has one unique callsign. Since the callsign is treated like any other message, embedded commands can be included if desired. The only way to play back a callsign is by embedding the **/M** command in a message.

N - Select Cut Number for number 9: When enabled, an **N** will be substituted for the number nine when sending a serial number. Command response: an *A* is sent when enabled, an *N* is sent when disabled.

O – Unassigned

P[n] - Set PTT Mode: (**n**=0 off, **n**=1 on) The user can choose whether to use automatic PTT while keying. This is a toggle command and after it is entered, WK will respond with an *A* for PTT enabled, or *N* for PTT disabled. Automatic PTT must be off if you want to control PTT manually by the **/U** embedded message command.

Q - Report Major and Minor Firmware Revisions: WK will send the two revision numbers, Major followed by Minor in Morse Sidetone. For WK 3,1,3 = 3 1 3

R [nn nn] - Set Speed Pot minimum and maximum: WK will prompt for two values, the minimum WPM and the maximum WPM. An error will be flagged if the minimum value entered is greater than the maximum, or if either value is less than 5. Maximum WPM is 99. If the speeds are valid, WK responds with an *R*, otherwise *?*

S - Preserve Settings in EEPROM: will respond with a letter *R* to signify that settings were saved successfully.

T - Select Cut Number for Zero: When enabled, a **T** will be substituted for the number zero when sending a serial number. An *A* response means this cut is enabled, an *N* response means it is disabled.

U - Swap Users: WK provides two complete user configurations, each with a unique callsign. Both message banks are available to both users. This allows a universal set of messages to be created that will work with either user. For example a message like **CQ CQ CQ DE /M /M K** will send the selected user's callsign. In response to this command a single dit is echoed when user 1 is selected and two dits when user 2 is selected.

V - Voltage Readout: WK will measure the PIC supply voltage and send it in Morse sidetone. For example a voltage of 4.52 will be sent as 4r52 with r indicating the decimal point.

W - Modify Sidetone Volume: The sidetone oscillator will be keyed at a steady rate with transmit muted. Pressing the paddles will raise or lower the volume. Even though 4 levels are possible, only 2 or 3 are useful. Pressing the command pushbutton will end this command and store the new sidetone frequency. Like all settings, use the Preserve Settings extended command to save the new volume setting in EEPROM.

X - Toggle paddle swap (Exchange dit/dah inputs): WK will respond with a letter *A* to signify when paddle swap is enabled and an *N* when turned off.

Y - RTTY Enable Toggle: Standalone RTTY transmit mode is toggled off and on. See 60. An **A** will be sent in sidetone when RTTY is turned on and an **N** when turned off. When RTTY is enabled, it is not possible to send CW by paddle or message. This setting is ignored when WK enters host mode.

Z - FSKMAP Toggle: When in RTTY mode, this will toggle the KEY and PTT assignments to allow remapping FSK and KEY assignments. An **A** is sent when FSKMAP is on, an **N** is sent when FSKMAP is turned off. FSK PTT and KEY=KEY when on while FSK=KEY and PTT=FSK when off.

Standalone Message Functionality

Messages are loaded by pressing and holding the command button until WK responds with an **R.** Press the message pushbutton of the memory slot you wish to enter. When WK is ready to accept a new message it will respond with an **E**. If you wait too long, WK will respond with a **?** and you have to start over. After WK responds with an **E**, enter a message directly on the paddles at a steady rate, making sure to leave proper space between letters. To insert a word space simply pause for longer than a word space and WK will respond with an **E** to signify a word space insertion. You can force a wordspace insertion by entering di-di-dah-dah (**IM**). This allows you to put a wordspace at the beginning of a message or insert more than one wordspace in a row. A ½ letterspace pad character can be inserted by entering di-di-dah-dah-dit (**IG**).

When the message has been completely entered, press the command push-button or enter di-dah-di-dah (**AA**) and WK will respond with an **R** to signify that the message was accepted and stored. If you make a mistake while entering a message, press and hold the command pushbutton and WK will backspace through the letters that have been entered. When you reach the position you want, release the button and new letters can then be added after that position. If the message memory becomes full while entering a message, WK will stop further loading, respond with an **F**, and then return WK back to non-command mode. There are 240 letters in message memory that can be distributed in any way between twelve message slots. The length of the individual message slots is not fixed. This means, for example, you could have one message of 80 characters, one message with 5 characters, and a third with 10 characters and still have 141 locations left to split among the remaining three slots. Keep in mind that word spaces, as well as embedded commands, occupy memory locations.

When you stop at the end of a message, WK will usually enter a wordspace before you have a chance to press the command pushbutton. There are cases when you do not want a wordspace delay at the end of a message since which will delay exchange turnaround. A way around this is to end with the di-dah-di-dah (**AA**) character. This will prevent inhibit the trailing wordspace so the message with stop immediately after the last letter.

If you are having problems loading messages into WK, make sure you leave adequate space between letters and are not sending much faster or slower than current command speed. If, for example, you enter an **A** followed by a **T** and end up with a **W**, you are not allowing enough space between letters. It's a fine line though because if you allow too much space WK will interpret that as a word space. Temporarily lowering the command speed (see command **C**) can help while you learn the process at a lowered speed.

To play a message back, simply press the desired message button and the message will be sent. There are two ways to abort a message: 1) Press both command and PB4 or 2) Press and hold either paddle.

If you want to review the message without keying the transmitter, use the **R** (review) command. Note that review embedded commands verbatim, for example the ½ letterspace pad character will be sent as di-di-dah-dah-dit and a 15 WPM speed change command would sent as dah-di-di-dah-dit S 15.

Standalone 'Two Press' Message Button Functionality

WKUSB-AF only has three pushbuttons, so messages 4 and 5 are accessed by pressing two pushbuttons in the following two finger sequence:

Press and **hold** either message button 2 or 3. (this specifies message 4 or 5)
Press the command pushbutton.
Release both push buttons.

Message 4 and 5 are sent in a similar manner by first pressing and holding pushbutton 2 or 3, then pressing the command pushbutton, and then releasing both. This works when you want to send message 4 or 5 and also when you want to load message 4 or 5. Press and hold the command PC wait for the **R** and then use the two finger message entry sequence.

Pushbutton Shortcut Commands

Standalone Quick MorseTune Command

In standalone mode, press and hold PB3 for about 2 seconds and the key and PTT outputs will be enabled for transmitter tuning. Pressing the command pushbutton will end tune. You can toggle between 50% and 100% tune duty cycle by pressing the paddles. Tune will be automatically turned off after 99 seconds.

RTTY Quick Tune Mode

When WKUSB-AF is in RTTY mode, tune is enabled by pressing and holding either PB2 or PB3 for 2 seconds. The KEY and PTT outputs, as determined by the FSKMAP and Reverse settings will be asserted until the CMD PB is pressed. Holding PB2 will generate a mark and PB3 will generate a space. A representative mark or space tone will be output from the CPO while tune is active.

Changing CPO Pitch (Frequency)

In standalone mode, press and hold PB2 for 2 seconds and CPO sinewave audio will be turned on. The pitch of the tone can then be changed by turning the speed control. Press the CMD pushbutton to end set mode and save the new frequency. If the speed control was changed to set a new pitch, the user will have to manually return the speed control to its original setting.

Standalone Quick Serial Number Decrement

Sometimes during contest operation, a serial number has to be resent. Since the serial number is automatically incremented after it's sent, we need a way to decrement the serial number. This can be done in one of two ways, using the **D** paddle command or by pressing and holding the CMD PB and then quickly pressing either paddle. In either case, after the serial number is decremented, WK will acknowledge by echoing a single dit.

There is a third way to decrement a serial number with a /**D** command embedded in a message. A message would be set up that decrements the serial number before sending it. Here is an example of two messages, one that sends a serial number is a message and a second that repeats that same serial number:

```
Message 1: 5NN /N QSL ?
Message 2: /D/N QSL ?
```

Message 1 is sent in the initial exchange and if the serial number needs to be resent, message 2 would be used. The second message decrements the serial number before sending it. Both of these messages leave the serial number incremented by one after it is sent.

Embedded Message Command List

It is an easy procedure to embed commands in a message. The format is the fraction bar *DN* (D and N sent together as one letter) followed by the desired command letter. If you want to insert the DN prosign into a message but don't want it to be interpreted as a command simply enter DN twice.
Example: **K1EL/1** would be entered as **K1EL//1.** In this document DN is signified by the / character.

Embedded message command table

Command	Description
/Bnn	Set a beacon cycle time of nn seconds (nn=00 to 99). Put this at the beginning of a message to set the beacon period. **/B10CQ CQ DE K1EL K** sends a CQ every 10 seconds.
/Cn	Call message in slot n (1-6), return and resume current message.
/D	Decrement serial number.
/En	Select message banks n=1 for bank 1, n=2 for bank 2
/Hn	Set HSCW speed. See table below for determining n.
/Inn	Increase letterspacing within a message, nn is a value from 0 to 31 percent times 2.
/Knn	Key transmitter for nn seconds. (nn=00 to 99)
/M	Send user callsign.
/N	Send Serial Number and automatically increment it after sending..
/On	Select key output port; (n=1 for port 1 or n=2 for port 2)
/P	Pause and wait for paddle entry and then continue after one word space time. The pause is ended three ways 1) paddle entry 2) Press msg PB2/PB3 or 3) Press the cmd PB to cancel.
/Q	Set QRSS speed. See table below for determining n.
/Snn	Set a speed change within a message. (nn=WPM, 5 to 99) **/S30FAST /S10SLOW**

/Un	PTT set (**n**=1 for off, **n**=2 for on) For this to work, PTT must be disabled (see **X P** command)
/V	Send the supply voltage in Morse, useful for beacons.
/Wnn	Wait for nn seconds. (nn=00 to 99)
/X	Cancel speed override, for example cancel HSCW, QRSS, or /Snn speed.
/Yn	Force a relative speed change up. Add **n** to the current WPM. n=(0-9)
/Zn	Force a relative speed change down. Subtract **n** from the current WPM. n=(0-9)

/1	Jump to message 1	**/2**	Jump to message 2
/3	Jump to message 3	**/4**	Jump to message 4
/5	Jump to message 5	**/6**	Jump to message 6
<u>**IG**</u>	Insert ½ letterspace pad	**(DI-DI-DAH-DAH-DIT)**	
<u>**IM**</u>	Insert wordspace pad	**(DI-DI-DAH-DAH)**	

Rate Table

n	HSCW Rate	QRSS Rate
0	1000 lpm (200 wpm)	3 sec dit
1	1500 lpm (300 wpm)	6 sec dit
2	2000 lpm (400 wpm)	10 sec dit
3	3000 lpm (600 wpm)	12 sec dit
4	4000 lpm (800 wpm)	30 sec dit
5	6000 lpm (1200 wpm)	60 sec dit

Embedded Command Examples

/B60BCON DE K1EL BEDFORD NH send BCON DE K1EL BEDFORD NH every 60 seconds

UR RST IS /P QSL pause to allow the user to enter the RST then resume automatically

/K05 /W10 VVV DE K1EL key down for 5 secs, wait 10 secs, and then send VVV DE K1EL

CQ CQ CQ DE /M /M /M send a 3x3 CQ using the user callsign

/H2CQ CQ DE K1EL K1EL K1EL/S15 DE K1EL send 1st part at 1500 lpm and the 2nd at 15 WPM

CQ CQ CQ DE K1/I10TMT/I00 K send message with 20% extra space in TMT. A more manageable way of doing this would be to load **K1/I10TMT/I00** in the callsign slot to get this spacing any time /M is used.

CQ CQ CQ DE /Z4K1EL K1EL K1EL/Y4 K send the callsigns 4 WPM slower then return to normal WPM.

QTH IS /E/C1/E NAME IS STEVE send the QTH from the alternate message bank and return for the name.

/Q2EL /1 continuously send EL at QRSS10 speed (this message is in slot 1). Avoid inserting a space between the QRSS command and the start of text: /Q2 EL unless you want a long delay at start of message.

/B10K1EL BCON/W2/VVOLTS send K1EL BCON, wait 2 secs, send XrXX VOLTS repeat every 10 secs

SOM/E2/C3/E2/U2/W5/U1/S50K1EL//1/XEOM send SOM, set msg bank 2, call msg 3, set msg bank 1, turn on PTT for 5 seconds. Change speed to 50 WPM send K1EL/1, cancel 50WPM send EOM.

Standalone Beacon Operation

There are several ways to run a beacon from WK in standalone mode. The first way is to load a message followed by a delay and jump. This would look like this, loaded into slot one:

K1EL BCON /W12 /1

K1EL BCON will be sent followed by a 12 second delay after which there is a jump back to the beginning.

The second way is to use the embedded beacon command, which is easier:

/B12K1EL BCON

Again K1EL BCON will be repeated every 12 seconds, but now the time between beacons will be exactly 12 seconds. In the previous example, the time between beacons would be 12 seconds plus the time it takes to send the message. This is more in line with what most users a looking for.

Finally, the third method is an 'on the fly' beacon. In this mode, any message slot can be sent as a beacon, for example if K1EL BCON is loaded into slot three, the user would press the command button, wait for the R, and then enter **B 3 12**. (wait for the WK to echo an **E** after **B** and **3**) This will beacon message 3 every 12 seconds. The advantage is that the message does not specifically have to be formatted as a beacon. The disadvantage is you have to enter the beacon paddle command every time you want to send it.

One application of a beacon is to set up a CQ loop that will send CQ, wait for some length of time, and then send it again until someone answers. If the beacon is in the wait state, you can simply start sendiing and this will end the beacon. If the beacon is in the middle of the message, tapping the paddle once or a PB3+Command "two press" (see page 54) will end the beacon and allow you to listen and respond to a station.

Standalone QRSS/HSCW Operation

The /H and /Q buffered commands allow HSCW or QRSS strings to be sent in standalone mode. (Note that QRSS speeds are not supported in Host Mode) HSCW and QRSS strings can be aborted with either a paddle press or a PB3+Command "two press" (see page 54). Upon abort normal keying speed is resumed.

PTT Functionality

PTT in standalone mode works the same way as it does in host mode. Note that in standalone mode the maximum allowable value for paddle entry is 99. This means the largest lead-in or tail delay that can be entered by paddle command is 990 mSec. If you need to enter a longer delay in standalone mode, use the WK3tools application which allows these values to be entered from the host over the USB port.

When entering a message by paddle and minimal tail time is essential, be sure to end all messages by entering di-dah-di-dah (**AA**) to prevent WK from automatically appending a wordspace. If there is a wordspace at the end of a message WK will hold PTT till the end of the wordspace.

Preserve Settings

WK setting changes made by paddle commands are not automatically copied into permanent storage. That means the settings will be lost if power is cycled. To save the settings permanently, use extended **S** command. Press and hold the command PB until the *R* and enter an **X** followed by a **S**. This will save current settings in nonvolatile memory so that all settings will be retained on power cycling. WK will respond with an R after the save is finished, Note that messages are saved directly into nonvolatile memory so the **S** command is not required to preserve messages.

Restore User Defaults (Warm Restart)

It's possible that a paddle command could be entered incorrectly which puts WK into an undesired state. Rather than try to figure out what went wrong, an easier way to handle this is restore settings from EEPROM. You can do this by either cycling power or by forcing a warm restart. To do the later, first press and hold the command pushbutton. An *R* will be sent followed by a **MN** (dah-dah-dah-dit) 5 seconds later which lets you know user defaults have been reloaded.

Restore Factory Defaults

Refer to Page 16 for a description of how to reset factory settings using WK3tools.

Keyer Lock

A lock feature is provided to disable paddle input and message pushbuttons. This is useful to prevent accidental keying. To lock the keyer, press the command pushbutton, wait for the R, and then enter a period (di-dah-di-dah-di-dah). WK will respond with an L to let you know it is now in a locked state. To unlock the keyer, press and hold the both paddles closed for about 8 seconds and WK will wake up and send an R.

Shortcuts and Speed Ups

Fast Command responses – WK uses *A* for positive responses instead of an *R* or *Y*

Cut Numbers on Command Entry – When entering numeric values use cut numbers to save time. For example when changing the command speed to 19 WPM, use **S U N** instead of **S 2 9**
These are the command cuts used in WK: T=0, A=1, U=2, V=3, 4=4, 5=5, 6=6, B=7, D=8, N=9

Single digit number entry – If you are entering a parameter that usually requires two digits but you only need to enter one, just enter the single digit and WK will figure out that there is only one. For example instead of entering **T 0 7** you can simply enter **T 7**.

WK will respond with an *E* or *I* when swapping users, message banks, or output port. Think of it as a single dit for 1 and a double dit for 2. This is much faster than responding with a Morse 1 or 2.

Fast Message Entry – Just press the command button till you get an R then simply press the message button you want to load. To stop load mode, either press the command button or enter di-dah-di-dah.

Fast Tune Mode – Press and hold pushbutton 3 to start tune. Press the command button to end tune.

Fast Serial Number decrement – Press the command button and then press either paddle to decrement the serial number. WK echoes a single dit to signify that the command was accepted.

Fast Command Response – If you think WK takes too long to respond to the command button, enable fast command reply mode which halves the delay time. This is the extended command **F**.

Command Prompts – WK will respond to commands that require additional input with an *E*. The E stands for "Entry Required" and is fast and efficient. For example if you enter the Weight command **W**, WK will respond with an *E* to let you know it is waiting for the value.

WKUSB-AF Standalone Tutorial

On 1st time power up, or by the restore factory defaults command, WK will be reset to these settings:

Operating WPM:15	Command WPM:15	Internal Sidetone: 800 Hz	Weight:50%
KeyComp:0	Letterspace Adjust:0	SampleAdjust:50%	KeyMode:Iambic B
Internal Sidetone:On	Autospace:Off	Key/PTT: Port 1	Serial Number:0001
Min Speed Pot: 5 WPM	Max Speed Pot: 35 WPM		No 0 or 9 Cuts
User: 1	Message Bank: 1	Tune Duty Cycle: 100%	Farnsworth: Disabled

You will need to attach a paddleset to proceed with the tutorial. After WK has been powered up, pressing the paddle keys will generate dits and dahs both in internal sidetone and keyed output. Let's enter a simple command to change the internal sidetone frequency. Press and hold the command pushbutton (CMD PB) until WK answers with an **R**. Then, without hesitation, enter a **Z** on the paddles. If it did not understand the command or you are late entering the command, WK will respond with a question mark. After a successful command entry, WK will respond by turning on a constant sidetone with keyed output disabled. By pressing the paddles right or left you can raise or lower the sidetone frequency. When you are satisfied with the pitch, end the command by pressing the command PB. WKUSB-AF uses a small internal speaker and you'll find that higher frequencies produce the loudest sidetone. CPO sidetone can be tested by connecting a set of stereo headphones or speaker with a stereo plug into WKUSB-AF's audio output. Start with the volume fully counterclockwise (lowest setting) and then press a paddle to generate sidetone. As you raise the volume you will hear sidetone in both internal and external speakers. The internal and external sidetone frequencies will be different.

Now we will enter a paddle command, one that will disable the internal sidetone speaker. Press and hold the command pushbutton (Red PB) until WK answers with an **R**. and then, without hesitation enter an **A** on the paddles. WK will answer with an **N** to let you know it accepted the command and turned internal sidetone off. Note that only the internal speaker is used for command entry. Now paddle sending will only be heard in the external speaker and command sidetone will only be heard from the internal speaker.
Now we will enter an extended command, one that will swap the paddles. Press and hold the command pushbutton (Red PB) until WK answers with an **R**. and then, without hesitation enter an **X** on the paddles. WK

will answer with an **E** to tell you that it is waiting for an extended command entry. In this case, enter another **X** which is the swap command. WK will reply with an **A** letting you know that paddle swap is now en**A**bled and the dit and dah paddles have been reversed. To swap the paddles back, repeat the command and this time WK will respond with an **N** for **N**ot enabled.

An important thing to remember is that changes made by paddle commands are only kept while power is on or until the keyer is opened by a host application. When power is restored, or application closed, defaults are reloaded from EEPROM. If you want to preserve setting changes, use the extended command **X S** and they will be stored in WK's internal EEPROM.

Now try changing the keying mode. Enter the **K** command followed by a letter signifying a keying mode (see page 49). Here are three examples, **K B** will set iambic mode B, **K A** will set iambic mode A**,** or **K U** will set Ultimatic mode. You might want to experiment with a couple of the modes just to try them. The sample adjust command allows you to tweak the paddle sample point; the default is a value of 50. Setting the sample point to zero will disable both dit and dah paddle memories.

Let's learn about the speed pot. If you turn the speed pot while sending you will notice that the sending speed will change. It will go faster clockwise (CW) and slower counterclockwise (CCW). When the pot is turned all the way CCW, favorite speed is selected. Favorite speed is set by the **S** command. To change it, press the CMD PB, wait for the **R**, and then enter **S**. WK will respond with an **E** telling you it's waiting for you to enter the favorite WPM. Enter a **1** followed by a **0** to set 10 WPM. Now you will set 10 WPM when the speed pot is fully CCW.

The default range for the speed pot is 5 to 35 WPM. Let's change the range using the extended **R** command (see page 53). Let's set a 10 WPM minimum and a 25 WPM maximum. Press the CMD PB, wait for the **R**, and then enter **X**, wait for the **E** and then enter **R** for pot range. WK will respond with an **E** telling you it's waiting for you to enter the minimum WPM value. Enter a **1** followed by a **0**. WK will then send another E while it waits for the maximum WPM value, enter a **2** followed by a **5**. If you did it right, WK will answer with an **R** and now the speed range will be 10 WPM to 25 WPM. Repeat the command and use the **T** shortcut for the zero, in other words enter a **1** followed by a **T**.

The WPM rate used while entering commands can be changed using the **C** command.

The Letterspace, Keying Compensation, Weight, and Dit/Dah Ratio commands adjust the way Morse is generated. Read each command descriptions on pages 49, 51, and 52 to understand how they work.

You can save your settings in WK's internal EEPROM with the extended command **S** for **S**ave settings.

The keyer's output port is toggled with the **O** command. This allows you to key one of two radios from the same keyer. This saves swapping cables around when you want to move from one radio to another. It does require a special keying cable wiring as described on page 5. PTT must be off to change ports. PTT is off by default, it is enabled by the **EXT P** command.

Now try entering some messages. Review the procedure for message loading on page 53. WK has two great features associated with messages. The first is backspace, if you make a mistake while entering a message you can backspace to fix the error. The second is that the size of the message slots is not fixed. If you only use two bytes in slot one, only two bytes of message memory are used up, not an entire slot. Once you have mastered message loading you can tackle some embedded commands. An easy command to start with is the speed change command. In slot one enter: **/S10SLOW /S25FAST**. Note that the DN prosign (/) is used as the command identifier. This message will be sent at two different speeds. Note that after sending this message the operating speed will be returned to the original speed, in other words the speed change is not permanent. If you need to insert / into a message simply enter it twice, this tells WK it is not a command: **K1EL//1**

Another set of speed related embedded commands are **/Y** and **/Z**. These invoke a relative increase or decrease in sending rate. They are different than the **/S** command since they add or subtract from the current speed. That means you can move you speed pot around to different speeds but the relationship between the current speed and the **/Y** and **/Z** commands remains constant. For example, look at the message:

 /Y5CQ TEST CQ TEST DE /Z5 WB6JJB.

The CQ TEST CQ TEST portion will be sent 5 WPM faster than the current WPM rate and then slowed back down for the callsign portion. If the current WPM rate was 10 WPM the accelerated rate would be 10+5=15 WPM. If the current WPM rate was 23 WPM, the accelerated rate would be 28 WPM. The message would have

the same affect in both cases; to send the CQ portion of the message 5 WPM faster than the callsign portion. This feature can be helpful to emphasize a portion of a message by slowing it down.

Now let's compose a beacon message. In message slot 2 enter**: /B60/K05 BCON DE K1EL NH** When this message is sent, the keyer will key down for 5 seconds and then send BCON DE K1EL NH. The **/B60** command tells the keyer to repeat the message every 60 seconds no matter how long the message itself is. To cancel a beacon, press the CMD PB or the paddle. WK will stop the loop and respond with an **I** to let you know something was cancelled.

Serial numbering is easy to use. First enter a starting serial number with the **N** command. You need to enter all four digits including leading zeroes. Next select the way you want WK to send 0s and 9s in a serial number. Use the extended **T** and **N** commands for this. To send a serial number simply insert a /N command into a message. It's tricky though since the serial number is automatically incremented every time it is sent. If you want to send a serial number twice in the same message you have to decrement it after you send it like this: **UR NR /N/D /N QSL ?**

Let's look at the embedded Pause command. The **/P** command can be thought of as a three way branch. When it is encountered, WK will wait for one of three things to occur:

1st branch: User paddles something and the message will continue
2nd branch: User presses a message button to send a message
3rd branch: User dual-presses CMD+PB4 to cancel the message entirely.

Here's a set of two messages that will illustrate one way the **/P** pause command can be used.

In slot 2 enter: **CQ TEST DE K1EL K /P UR NR/N QSL?/P**
In slot 3 enter: **UR NR /D/N QSL?**

When you press PB 2, the message in slot 2 will send CQ and then pause to let you listen for a reply. If there is no reply, hit message PB 2 again to repeat the CQ. If there is a reply, enter the station's callsign and WK will automatically continue, send the serial number, and then pause again. If the station needs a repeat of the serial number press PB 3 to send the message in slot 3. Since the serial number is incremented after the **/N** command in message 2, you need to pre-decrement it in order to send the correct number. Message 3 can be repeated until the station acknowledges the exchange. At that time press PB2 to start the whole process again.

WK supports two alternate sending formats. They are selected by embedded commands in a message. QRSS is extremely slow CW for VLF work, and HSCW is extremely fast CW typically used for meteor scatter QSOs.

Here are examples of each:

QRSS: **/K10/ Q2EL/2** Key down for 10 seconds followed by EL at QRSS6 rate then repeat.

HSCW: **/H3K1EL K1EL K1EL K1EL K1EL K1EL K1EL /1**   Callsign in slot 1 is repeated at 3000 lpm

There are two direct pushbutton commands in CW mode:

In CW mode, pressing and holding PB2 down for 2 seconds will enable CPO sidetone. Then the pitch can be adjusted by turning the speed pot. When you are happy with the new pitch, press the command pushbutton and it will be stored permanently (a save command is not necessary). If you press and hold PB3 for 2 seconds, the keyer will be put in tune mode. The active output port will be asserted along with PTT if it is enabled. Pressing the paddles will toggle between 100% and 50% duty cycle.

WK Standalone RTTY Tutorial (using paddle entry to send RTTY)

RTTY capability was introduced in WK version 3.1. In addition to the ability to send RTTY in host mode , WK can also operate RTTY in standalone mode with input from a paddleset or message pushbuttons.

Put WK into RTTY mode with the extended command **Y**. Reverse (swapping mark and space) is toggled by the extended command **K**. Usually, WK's KEY output is fed to your transceiver's KEY input to key the transmitter and PTT is fed to the transceiver's FSK input to generate FSK RTTY. The assignment of KEY and PTT can be reversed by the FSKMAP toggle (extended command **Z**) to get your KEY and PTT outputs configured correctly. It's very important to set an appropriate PTT lead in and tail delay. This prevents FSK from being chopped at the

start and end of transmission. A good starting value is 50 for both lead in and tail. Once the transceiver is configured to operate in RTTY mode you are ready to go. The only other configuration decision is whether diddle should be on or off. The recommendation is to use diddle to fill in the empty space between paddle input letters. Be sure that WKUSB-AF internal sidetone speaker is enabled (the **A** paddle command) so that you can monitor paddle dit/dah entry.

To start a RTTY transmission, enter the Morse code di-dah-di-dah. This will key your transceiver and a CRLF sequence will be sent followed by diddle if enabled. . As you enter letters on the paddle you will hear them through the internal sidetone speaker. After each character has been completed, it will be sent in RTTY FSK Baudot. FSK audio will also be output through the audio port. If you don't want to listen to that, just turn WKUSB-AF's audio volume down and monitor it through your transceiver. It can be confusing but some ops like to hear both. It is possible to overrun the RTTY input buffer if you send Morse very fast, about 20 - 25 WPM works well. To insert a CRLF, enter the Morse code DAH-DAH-DAH-DAH. For tuning or testing, an RYRYRY string can be generated by entering the Morse code DAH-DAH-DAH-DIT. When you are done sending enter di-dah-di-dah which will unkey the transceiver and return you to receive mode.

Enter a test message into one of WK's message slots. If you want to send a message, simply press the message button and the transmit process will proceed automatically. You do not need to insert the di-dah-di-dah codes into a message, WK will automatically do that for you. There are two banks of six messages to work with.

Some ops like to set up one User for RTTY and the other for CW. That way you can switch back and forth with a user toggle command (**XU**). If you use some thought you can load messages that will work for both modes.

As in host RTTY mode, a lead and tail time should be set to help the receiving station quickly sync to your transmission. Recommended values are 50 for both lead in and tail delay.

While paddle RTTY is not as easy as keyboard, it does provide a way of operating RTTY with minimal equipment, especially if you have a transceiver that will decode and display received RTTY.

When RTTY is enabled, pressing and holding PB2 for two seconds will put the keyer in Mark tune mode. The CPO will send out the Mark tone and Key and FSK will be sent out the output ports. To cancel, press the CMD pushbutton. If PB3 is pressed for 2 seconds, the space tone will be heard along with Key and PTT assertion. How the outputs are formatted will depend on the FSKMAP and Reverse settings.

WK3 Standalone Command List

```
A  -  Select sidetone on or off        N  -  Load 4 digit serial number
B  -  Start Easy Beacon                 O  -  Select output key port
C  -  Set command speed in WPM          P  -  Start Practice Mode
D  -  Decrement serial number           Q  -  Query current settings
E  -  Swap active message bank          R  -  Review message without transmitting
F  -  Set Farnsworth Speed              S  -  Set favorite WPM
G  -  Toggle 50% tune duty cycle        T  -  Load PTT Tail Time
H  -  Set Paddle Hang Time              U  -  Select Autospacing on/off
I  -  Set Letterspace Adjustment        V  -  Set Keying compensation in mSec
J  -  Set Paddle sensitivity            W  -  Set Key Weight
K  -  Select keyer mode                 X  -  Enter an Extended Command
L  -  Set PTT Lead In Time              Y  -  Set Dit/Dah Ratio
M  -  Mute Transmit, Tx sidetone only   Z  -  Set sidetone frequency
```

Extended Command Table

```
A  -  Push button diagnostic           P  -  PTT enable on/off
C  -  Contest spacing on/off           Q  -  Report Firmware revisions
D  -  Favorite Speed mode on/off       R  -  Set speed pot range (Min/Max)
E  -  Set first extent                 S  -  Save settings in EEPROM
F  -  Fast command response on/off     T  -  Cut 0 on/off
G  -  Report Speed Pot Value           U  -  Select user 1 or 2
K  -  Toggle RTTY Reverse Enable       V  -  Report power supply voltage
L  -  Toggle RTTY Diddle Enable        W  -  Modify Sidetone volume
M  -  Load user callsign               X  -  Swap paddles
N  -  Cut 9 on/off                     Y  -  Toggle RTTY Mode Enable
O  -  Unassigned                       Z  -  Toggle RTTY FSKMAP
```

Standalone Embedded Command Table

```
/Bnn    Set a beacon cycle time of nn seconds (nn=00 to 99)
/Cn     Call message and then return
/D      Decrement serial number.
/En     Select message bank (n=1 for bank 1, n=2 for bank 2)
/Hn     Set HSCW speed.  See table below for determining n
/Inn    Set Letterspace increase (nn=00 to 31)
/Knn    Key transmitter for nn seconds (nn=00 to 99)
/M      Send User Callsign (set user callsign with the XM command
/N      Send Serial Number with auto increment.
/On     Select Keying Port (n=1 for port 1, n=2 for port 2)
/P      Pause and wait for paddle entry and then continue after one word space time
           The pause can be ended in three ways:
           1) Paddle Press,
           2) Press a message PB
           3) Dual-press PB3+CMD to cancel
/Qn     Set QRSS speed. See table below for determining n
/R      Merge next two letters into a prosign
/Snn    Set a new sending speed (nn=WPM, 5 to 59)
/Un     PTT Off/On (n=1 for PTT off or n=2 for PTT on)
/V      Send supply voltage
/Wnn    Wait for nn seconds (nn=00 to 99)
/X      Cancel /S, /H, or /Q command
/Yn     Relative WPM Change Up (n=0 to 9) current WPM changed to WPM+n
/Zn     Relative WPM Change Down (n=0 to 9) current WPM changed to WPM-n
/1      Jump to message 1   /2   Jump to message 2
/3      Jump to message 3   /4   Jump to message 4
/5      Jump to message 5   /6   Jump to message 6
```

n	HSCW Rate	QRSS Rate
0	1000 lpm (200 wpm)	3 sec dit
1	1500 lpm (300 wpm)	6 sec dit
2	2000 lpm (400 wpm)	10 sec dit
3	3000 lpm (600 wpm)	12 sec dit
4	4000 lpm (800 wpm)	30 sec dit
5	6000 lpm (1200 wpm)	60 sec dit

Rate Table for /Hn or /Qn commands

Topical Index