PREFACE

Midway through the production run of the K1 keyer, some additional firmware features were added further enhancing the capabilities of the keyer.

A supplement to the manual was printed and supplied with later keyers.

If your keyer is one of the later keyers, you should also download and print out the

"CMOS II Manual Supplement - Version2.pdf"

which is also available on the site, **www.idiompress.com**, where you downloaded this manual.

Here is how to determine if your Logikey K-1 has the version 2 features:

Enter the "inquiry" mode by pressing down buttons 3 and 4 and releasing them together. Answer the question mark sent in CW by sending the letter "V". If you get the raspberry sound, your keyer is Version 1; if you get a digit between 0 and 9 you have version 2, in which case you should print out the supplement and add it to your manual.

If you have a Version 1 keyer, it is possible to upgrade it to Version 2. The conversion requires replacement of the CPU and the replacement of the timing capacitor. It is covered in the supplement referenced above. Contact Idiom Press for a quotation on the components needed.

73

Idiom Press

OPERATING MANUAL

Logikey K-1 KEYER

by KCØQ and NØII

The Logikey K-1 Keyer is a compact, full-featured memory keyer using CMOS microprocessor technology for low cost and high reliability. In contrast to other advanced keyers, its control requires neither a myriad of switches and knobs nor complicated keypad sequences. Instead, commands are simply sent to the keyer in Morse code using your paddles! Some of its features are listed below:

1. Iambic keyer with dot and dash memories.
2. Four messages of 384 bits or 48 characters each.
3. Messages may 'call' others and contain functions.
4. Input queue to store multiple message activations.
5. Contest serial number --- 001 to 9999.
6. Digital and linear analog speed control -- 6 to 60 WPM.
7. Adjustable weight on code elements -- 25% to 75%.
8. Built-in sidetone monitor -- 700 Hz.
9. Tune function for transmitter adjustment.
10. Selectable automatic character spacing.
11. Timed pauses within messages.
12. Message loop capability for continuous replay.
13. Break in message to allow paddle-inserted text.
14. Ultra-low power consumption for battery operation.

RESET STATE

After power is applied, the keyer responds with "OK" and is ready for operation. It is initialized as follows:

Speed:	20 WPM	Load mode:	character
Weight:	50%	Input queue:	on
Monitor:	On	Serial number:	001
Auto-space:	Off	Number option:	0
Function speed:	Equals Paddle speed		

FUNCTION COMMANDS

Commands to the keyer are entered in Morse code using your paddle. To alert the keyer that a function is desired, momentarily press buttons 1 and 2 simultaneously. The keyer enables the monitor, disables output keying, and acknowledges your request by sending "F".

After hearing the "F", simply paddle in the desired function character(s) explained below. When the function is complete, the monitor is automatically restored to its previous state, output keying is enabled, and the previous operating speed is restored.

Input command strings and the function performed are as follows:

A Auto-space -- toggles the automatic character spacing feature on or off. The keyer confirms the new state by sending "ON" or "OFF". When on, auto-spacing aids the operator in forming properly spaced characters within a word. Character crowding is prevented by forcing at least 3 elements of space whenever more than 1 space has been detected. This effect is especially very noticeable at low speeds. Without auto-spacing, character spaces are determined by the operator.

D Decrement -- decrements the serial number by one, effectively canceling the automatic increment applied when last played from a message. The decrement function accommodates resending the last serial number, as might be needed when a repeat of a contest exchange is requested.

F *dd* Function speed -- sets the speed used for function entry to *dd* WPM, where *dd* are two digits in the range 06 to 30. This speed is employed for entering commands and loading messages. It is independent of the operating speed and is unaffected by the analog speed control. Alternatively, the function speed can be made to follow the operating speed by using *dd* = ØØ.

H Hand-key -- causes the keyer to enter the hand-key mode. Keying output will follow closures of the dot or dash paddle levers, allowing hand-sent code. Normal iambic keyer operation is regained by any button closure.

K*dd* Keying Compensation -- increases keying on-time and decreases off-time by *dd* milliseconds, where *dd* are two digits in the range 00 to 25. Though similar to increasing weight, the adjustment is independent of speed.

L Load mode -- toggles the load mode between character and real-time. The keyer confirms the new mode by sending "C" or "R" as appropriate.

M Monitor -- toggles the 700 Hz sidetone monitor on or off. To conserve battery power, the keyer should normally be operated with the monitor off in favor of the rig's sidetone.

N *dddd* Number -- initializes the contest serial number to *dddd*, where *dddd* are four digits in the range 0000 to 9999. Note that 4 digits must be entered, with leading zeros if needed.

Q Queue -- toggles the input queue on or off. The keyer confirms the new mode by sending "ON" or "OFF" as appropriate. When off, message button activations are acted upon immediately, canceling any message in progress. When on, up to 8 button presses are remembered in order and acted upon in succession as each message completes.

RV Reverse -- reverses the dot and dash paddles as may be desired for left-handed operators.

S *dd* Speed -- sets the current operating speed to *dd* WPM, where *dd* are two digits in the range 06 to 60. The speed function establishes the current knob position at the entered speed. The knob can then be employed to vary the speed faster or slower over a total range of 3:1. Control via the knob is linear and increases speed clockwise.

SU *d* Speed Up -- increases the operating speed by "*d*" WPM, where d is a single digit in the range 0 to 9.

SUD *d* Slow Down -- decreases the operating speed by "*d*" WPM, where d is a single digit in the range 0 to 9.

W *dd* Weight -- sets to code weight to *dd* percent, where *dd* are two digits in the range 25 to 75. Weight is the duty cycle of a continuous string of dots, which is 50% for perfect code. A higher weight produces a heavier sound, and a lower weight causes characters to sound lighter. Once set, weight remains constant and independent of speed.

X Xmit (tune) -- Continuously keys the output for purposes of transmitter and amplifier adjustment. Tuning is stopped by simply tapping either the dot or dash paddle lever.

Z *d* Zeros and Nines: controls the way that zeros and nines are sent in the serial number. See SERIAL NUMBER OPTIONS.

INQUIRY FUNCTIONS

Inquiry functions allow the current state of the keyer to be determined. State information is played out in Morse code with the monitor automatically enabled and keying output disabled. Inquiries operate just like command functions: to enter an inquiry, momentarily press the right two buttons (#3 & #4) together. After receiving the "T" reply, enter the desired inquiry command as follows:

A Inquire Auto-space -- the keyer responds by sending "ON" or "OFF" as appropriate.

F Inquire Function Speed -- the keyer sends the current function speed setting in WPM as two digits.

K Inquire Keying Compensation -- the keyer sends the current compensation in milliseconds as two digits.

L Inquire Load mode -- the keyer responds by sending "C" if in character mode or "R" if in real-time mode.

N Inquire Number -- the keyer plays the current contest serial number (but does not increment it).

Q Inquire Queue -- the keyer responds with "ON" or "OFF" as appropriate.

S Inquire Speed -- the keyer sends the current operating speed in WPM as two digits.

W Inquire Weight -- the current weight percentage is sent by the keyer as two digits.

Z Inquire Zeros and Nines -- the keyer responds with the option number (0-9) currently in effect.

1 Inquire Message 1 -- message 1 is played.

2 Inquire Message 2 -- message 2 is played.

3 Inquire Message 3 -- message 3 is played.

4 Inquire Message 4 -- message 4 is played.

EMBEDDED FUNCTIONS

Certain functions can be embedded within character messages. To distinguish them from normal text, the command strings are prefixed by a "/" and are entered as a separate word. When encountered during a a message play, the functions are executed. Note that if "/" is part of a single word, as in W9KNI/ZA2, it is sent as expected and is not interpreted as a command prefix. Embedded command strings and their use are explained below:

/B Break -- message play is suspended to allow insertion of paddle text. The operator may then insert one or more words using the paddle. Once paddle input has begun, the break function is canceled when inactivity exceeding a word space is detected. The interrupted message is then resumed. A break may also be aborted by pressing a button, which will cause the corresponding message to play immediately.

/D Decrement -- decrements contest serial number by one.

/G*d* Gap -- the normal 7-element interword space is modified to 3+*d*, where *d* is a digit in the range 0 - 9. It is used to exaggerate inter-character or inter- word spacing. For example, a call like WØEJ can be entered with slight lengthening of the space between the "E" and the "J" for emphasis, making it easier to copy. Note that /GØ yields a normal character space, while /G4 yields a normal word space.

/N Number -- the current value of the contest serial number is played. Its value is then automatically incremented by one. Also see SERIAL NUMBER OPTIONS.

/P*dd* Pause -- a speed-independent pause of *d.d* seconds is inserted, where *dd* are two digits in the range 00 to 99. For example, /P35 will result in a delay of 3.5 seconds. Pauses longer than 9.9 seconds are obtained by using consecutive commands that total the value desired.

/S*dd* Speed -- the operating speed is set to *dd* WPM, where *dd* are two digits in the range 6 to 60.

/SU*d* Speed Up -- increases the operating speed by *d* WPM, where *d* is a single digit in the range 0 to 9.

/SD*d* Slow Down -- decreases the operating speed by *d* WPM, where *d* is a single digit in the range 0 to 9.

/1 Message 1 - message 1 is played followed by resumption of the current message. If the current message is 1, a continuous loop is created.

/2 Message 2 - message 2 is played followed by resumption of the current message. If the current message is 2, a continuous loop is created.

/3 Message 3 - message 3 is played followed by resumption of the current message. If the current message is 3, a continuous loop is created.

/4 Message 4 - message 4 is played followed by resumption of the current message. If the current message is 4, a continuous loop is created,

MULTIPLE-BUTTON FUNCTIONS

Single-button closures are reserved for activating messages. As already discussed, the 1-2 combination alerts the keyer for paddle-entry functions. Certain functions are duplicated, wholly or in part, by other button combinations:

2-3 Decrement -- the keyer acknowledges with "D" and then decrements the serial number.

3-4 Inquiry -- the keyer acknowledges with "?" and then waits for paddle entry of the desired option.

1-3 Hand-key -- the keyer acknowledges with "H", then enters hand-key mode until another button closure occurs.

2-4 Tune -- the keyer acknowledges with "X" and then keys the output continuously until a paddle closure occurs.

1-4 Reverse -- the keyer acknowledges with "RV" and then reverses the paddles.

1-2-3-4 Reset -- the keyer speed is reset to 20 WPM at the current knob position, the function-entry speed is defaulted to follow the knob, and then the keyer acknowledges with "OK".

LOADING CHARACTER MESSAGES

In character mode, each Morse character uses one byte of message memory. Precise 3-element intercharacter and 7-element interword spaces are employed when the message is played (unless modified using the /G*d* function). To load a message, first confirm that character-mode loading is in effect by using the "L" Inquiry function. Then press and hold the desired message button. After 2 seconds, a tone is emitted and the button may be released. The keyer then sends "C" to confirm character mode and waits for input.

Morse code text and embedded functions can then be entered with the paddles. When each word is complete, simply stop sending. The keyer will detect and insert a word space, and then prompt you for the next word by sending "I". There is no limit to the elapsed time between words, so there is no need to hurry.

The keyer includes a powerful edit feature for immediate correction of errors during message loading. If a mistake occurs, simply send an error indication of 7 or more dots. The keyer will locate the last word and erase it. It will then play the 'new' last word (if any) so that the position in the message is known exactly. As many words as needed can be erased this way. When the desired position is achieved, continue to enter the remainder of the message.

After the keyer responds with "I" following the final word, the message is ended by a momentary press of the button. To completely clear a message, just end the load in this way immediately after the "C" is sent. In the event that message capacity is exhausted during the load, automatic termination will occur.

LOADING REAL-TIME MESSAGES

Some operators prefer messages containing stretched or compressed spacing rather than perfect timing. Real-time mode stores and replays messages exactly as entered. Along with somewhat reduced message capacity, the primary disadvantage of real-time messages is that they cannot contain embedded functions. Use the Inquiry function "L" command to insure that real-time load mode is in effect. Then press and hold the desired message button. After 2 seconds, a continuous tone will emit, at which time loading is enabled and the button may be released. The keyer then responds by sending "R" to confirm real- time mode and waits for paddle input.

Since the keyer waits until the first paddle closure, there is no need to rush the first entry. Once entry begins, however, the keyer loads continuously -- any pauses are stored as spaces in the message. All intercharacter and interword spacing is strictly up to the operator. To end the message, simply press the message button momentarily. Note that the elapsed time from the end of the last character to message termination is stored as space at the end of the message. If room is exhausted during the load, the message is terminated automatically.

PLAYING MESSAGES

It couldn't be simpler: just tap the desired message button. Both character-mode and real-time messages can be played regardless of the current load mode.

If the input queue is enabled, multiple message button closures will be remembered. Each message will be played in succession as the previous completes. As a simple example, suppose that message 1 contains "CQ" and message 4 contains "DE WB8ZRL". Then pressing button 1 three times and button 4 once, in quick succession, will cause "CQ CQ

CQ DE WB8ZRU" to be played. As many as 8 button activations will be remembered in this mode. With the input queue disabled, however, a button closure immediately cancels any current message and starts the commanded one.

If a paddle closure is detected, messages are immediately aborted and the input queue flushed, except during execution of a /B break function when paddle input is expected. Otherwise, paddles always take priority over message playback.

SERIAL NUMBER OPTIONS

The "Z*d*" command selects one of 10 options for sending zeros and nines in the contest serial number. Zeros may be replaced by "O" or "T", nines replaced by "N", and leading zeros suppressed. The options available are:

Option	:	Leading	Other	
(d)	:	Zeros	Zeros	Nines
Ø	:	Ø	Ø	9
1	:	-	Ø	9
2	:	O	Ø	9
3	:	O	O	9
4	:	-	O	9
5	:	T	Ø	9
6	:	T	T	9
7	:	-	T	9
8	:	T	T	N
9	:	-	T	N

Note: For numbers less than 1000, the first zero is always suppressed.

ERROR INDICATION

When an erroneous input or condition is detected, the operator is notified by a distinctive raucous tone burst. Examples include nonexistent functions, invalid numeric parameters, and exhaustion of message capacity during a load.

EMBEDDED FUNCTION EXAMPLES

Perhaps the most powerful feature of the keyer is its ability to store functions within messages. Stored commands are executed as encountered when a message is played. Serial numbers, for example, are needed in some contests. A Sweepstakes exchange could be programmed as exemplified by: "NR /N A KCØQ 8Ø IA BK". A second message might contain a serial number repeat message. This makes use of the decrement function: "/D NR /N BK".

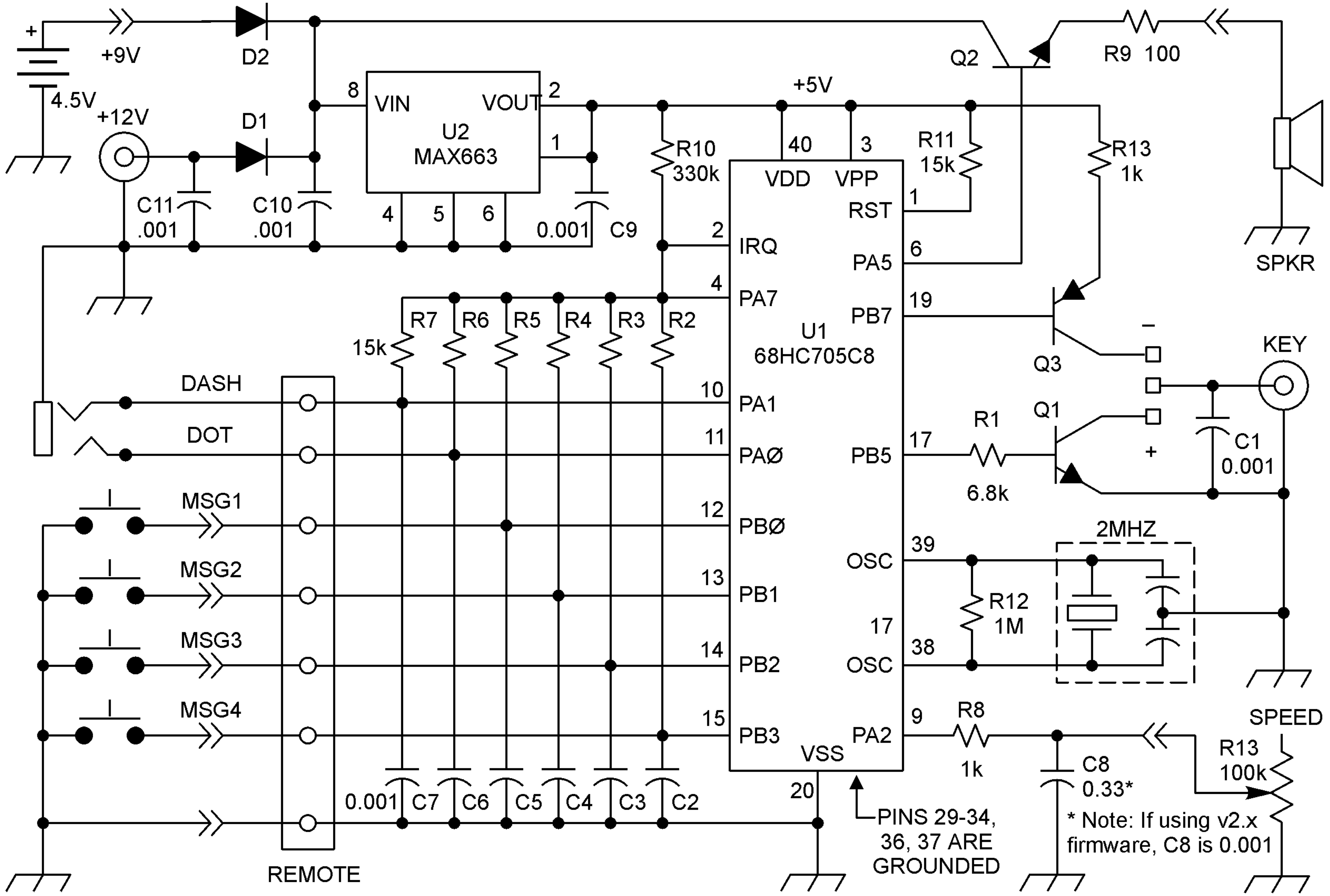
Speed changes within messages are also possible, allowing parts of a message to be played at differing speeds. Using speed changes as in "/SU5 QRZ DE WØSR/7O UP 5 /SD5" plays the message faster, then returns to the previous speed. Or your two or three favorite operating speeds can be programmed into messages for instant changes with a tap of button. For example, a message consisting of simply "/S25" could be used to set the speed to 25 WPM.

Messages may 'call' other messages. Suppose message 4 contains "WA9CNS/KH7". Then the message "CQ CQ CQ DE /4 /4 K" will, when played, yield "CQ CQ CQ DE WA9CNS/KH7 WA9CNS/KH7 K". Call nesting may be as deep as desired. Thus, message 2 can call message 1 which calls message 4 which calls message 3! In fact, continuous loops can be programmed. A loop will result if a message calls itself (directly or via some other message).

Loops can be very useful. A CQ loop in message 1, using a pause, is a good example: "CQ DX CQ DX DE WØWP WØWP K /P35 /1". The message will play continuously, with 3.5 seconds in between. When an answering station is heard, simply tap either paddle to cancel the loop.

If you are fortunate enough to be able to "run 'em" in a contest, the break function and looping can be a real advantage. Message 1 might contain: "QRZ TEST DE WØWP /B 599 A BK /B /1". Here, message 1 first causes "QRZ TEST DE WØWP" to be sent. The /B breaks the message so a responding station's call sign to be copied. The call is then sent via the paddles. The message then automatically continues and sends "599 A BK" followed by another break. The other station's report is then copied while the keyer waits. A simple "TU" or "R" is then sent via the paddles to acknowledge the exchange. This completes the break and causes an automatic loop back, restarting the whole sequence.

If no one responds to the QRZ, simply tap button 1 again to resend the QRZ (remembering that a /B is canceled by a button closure). If the responding station's exchange is missed during the second break, use message 2 to ask for a repeat: "AGN? BK /B /1". After getting the repeat, acknowledge with "TU" or "R" as before, and the QRZ loop is automatically resumed.



* Note: If using v2.x firmware, C8 is 0.001

LOGIKEY K-1

TUTORIAL

Logikey Division
Idiom Press
Box583
Deerfield, IL 60015

Getting Started

Before you can use your Logikey K-1 keyer, you must wire your paddle to a 1/8" (3.5mm) stereo plug. Be sure to use two conductor shielded wire, with the shielding as the ground connection to the paddle. The dot lead of the paddle should be hooked to the tip, the dash lead to the ring, and the shield to the sleeve. You will also need a 9 volt battery, a 2.5 mm power plug, and an external 12 volt DC source. Logikey offers a hook-up kit model HK-1 containing these items except for the 12 volt supply, available from your dealer, or the individual items may be purchased at Radio Shack or a similar outlet.

To install the 9 volt battery, used to save the keyer configuration and programmed messages when external power is removed, open the top by removing the two screws on the sides of the keyer. You will find a catch strip on the floor of the keyer with a mating hook strip attached. Pull off the hook strip, remove the backing and attach it to your nine volt battery. Attach the battery connector to the battery, then fasten the battery to the chassis catch strip.

Check the keying polarity, and change if needed. Looking down on your keyer, you will see the RCA jack at the left rear corner. Moving forward, you will see a disk capacitor, then a three post header with a jumper between the center pin and either the right pin or the left pin. If your transmitter uses positive keying (as do virtually all modern solid state rigs) the jumper must be on the right or inside two pins. To use your keyer with negative line keying (Usually found on older rigs using tube finals), the jumper must be on the left two (or outer) pins. Now, replace the cover on the keyer. Attach the paddle to the keyer through the rear plug, attach a twelve volt source to the power plug with the positive voltage going to the center pin. Once you have familiarized yourself with operating the keyer, connect the keyer to your rig using an RCA plug at the keyer output. You are now ready to begin.

Your K-1 will go from 6-60 words per minute, but not all at once! Rather, the speed control on the front panel covers roughly a 1:3 speed range. So, you can set the keyer for a range of 6 - 20 WPM. Again, you might prefer a range of 20 - 60 WPM. But most users will prefer a speed range somewhere in between.

So you want a range of 6 - 20 WPM? Setting this range could hardly be easier! Turn the speed control knob all the way to the right. Now, press all four control buttons down simultaneously, then release them. The keyer sends a little message to you in Morse - "OK", telling you it has accepted the instruction. The keyer always sets itself at twenty WPM at whatever point the speed control knob is set whenever you push the four control buttons together as you just did.

Send a little code from your paddle. Oops! You say the dot and dash sides are reversed from where you want them? (Most right handed operators prefer the left paddle for dots and the right paddle for dashes.) NO problem! Simply press the two outer buttons (#1 and #4) at the same time, and release. The keyer will send back in Morse a message - "RV", as in ReVerse. Now you will find that what was the dot side is now the dash side, and vice-versa. And it will stay that way until you reverse it again by pushing those two buttons again, or until the 9 volt back-up battery wears out in a few years. Neat, huh?

If you have used keyers before, you will instantly notice the fluid smooth timing as the CW rolls off. Now, increase the speed of the keyer. See how the speed control is linear? Reset the speed range if you like. If you are comfortable sending at 20 WPM, turn the speed control knob all the way to the left, then push the four top buttons again. Now you are set for a range of 20 - 60 WPM. Or set the speed control knob pointing straight up, then press the four buttons. That will give you a range of about 11 - 33 WPM, with 20 WPM in the middle.

OK, the keyer is hooked up and working, the dot side is where we want it, and we've set the speed range so it is comfortable. Now let's learn how to load a message into memory.

Your keyer is capable of storing "The quick brown fox jumped over the lazy dogs back" in each of the four memories, which is far more than is needed for normal operating situations. Here's how to store a message in memory. First, set a speed that is comfortable for you. Now, press the far left button down, (button number 1), and hold it several seconds until you hear a tone. Then release the button. The keyer will send a "C". (This stands for character mode. But don't worry about what that means for now. We'll get into it later.)

Now, begin sending your message, a word at a time. Let's load a message, "the quick brown fox". The first word is "the", so simply send "the". Release the paddle. There. The keyer just sent a Morse "I" to you. That means it has accepted the word, and has injected a word space. Now, send "quick" through the paddle, then stop again. The keyer will send another Morse "I". At this point, the keyer is prepared to wait as long as necessary for you to program in the next word. If you want, you can take time to go get a cup of coffee and return to the shack. It will still be waiting for the next word.

Let's go on and send "brown fax". Whoops. We wanted "brown FOX", didn't we? Must we bail out and start all over again? Heck no. Instead, send the international "I goofed" symbol - a stream of seven or more dots, and release the paddle. The keyer will send "brown", telling you it has erased the incorrectly sent word (In this case "fax") and backed up to the word before, "brown". It sent the word "brown" to remind you where you are in the message, and is ready for you to resume loading the message you want. So now send "fox".

OK, we've loaded the message. To close the message, simply press button 1 down momentarily and release it. To play the message, press button #1 again briefly, and listen to the keyer send the message you programmed, complete with the correction.

Load another message, "jumped over the lazy dogs back" into the second memory, using button #2. Load the message in just like the first message. OK? Play it back to make sure you got it right.

Press button #1, release it, and immediately press button #2 and release. The two messages will play, one after the other. You could load your call on message #1, "AB1CD", and on message #2 "DE AB1CD". On message #3, you could put in "AR K". Then, by pressing buttons #2, #1, and #3 in that order, you would have chained or "queued" together a message "DE AB1CD AB1CD AR K". (You also have the option of NOT having multiple button presses give queued messages, but rather stop one message and begin another. We will get into that later.) You can stop a message being sent at any time by simply touching your paddle, which instantly kills the memory transmission and puts you back in the normal keyer mode.

Once you enter a message into memory, the message is stored even when you remove power from the keyer. The 9 Volt battery saves the memories, and because of the extremely low current drain will last for years.

To kill a message already entered, you can either load a new message in over the top of it or you can erase it. To overwrite it, press down the memory button and hold it down for several seconds until you hear the tone. Release the button, wait for the "C" indicating you are in character mode, and enter away. If you want to simply kill the message, press the button again after you get the "C", and the message will be closed empty until you program a new message later.

How do we shut off the monitor when we go on the air, so we can use the transceiver sidetone instead? Simple. Press the left two buttons (one and two) down together, then release them. The keyer will send back to you the letter "F". (This stands for "Function".) Now, send "M". That's it. The monitor will be off during normal transmissions, and you instead rely on the transceiver's sidetone.

Now let's turn the monitor back on. Press the left two buttons again. You will hear the speaker send an "F", even though the monitor is disabled for normal sending. Now, send another "M", which you will hear through the monitor as you send it. That's it. The monitor is on again. The monitor control function is a toggle command, and you have learned how to switch it on or off. Note that the "F" the keyer sent did not go out over the air, nor did the "M" you sent to toggle the monitor function. In pressing the two function buttons, you took the keyer "Off Line" and off the air until you had completed your command.

Suppose you screwed up and sent an "O" instead of an "M". Since the "O" is meaningless to the keyer, it will send a raspy signal, a Bronx Cheer. That means you goofed. To recover, simply press the two left buttons again, and send the "M" again.

Do you need a key closure to tune up the rig? Press buttons #1 and #3 together, then release them. The keyer will send an "H", as in "Hand-key." Now, any time you press either side of the paddle, instead of sending dots or dashes, you will get continuous output. This lets you hold the "key" down while you tune up that big rig (into the dummy load, of course) in easy stages. When finished, press any button and the keyer will be back to normal.

OK, that's enough for now. You've learned how to set the speed you want, how to program messages, how to load the rig. It's time to hook the keyer up to the rig, get on the air and make a few QSO's, and enjoy how fluid and clean CW can be. And when you are ready, move on to the next section of this tutorial, and we'll go into some of the fancier options. They are easy to learn too, but right now let's use what we have learned.

SECOND SESSION

OK You've been using your K-1 Keyer, and you've discovered how smooth and well behaved it is. Likely your transmitting speed has improved as well. Let's start exploring the next level of features your keyer has to offer. Let's turn the rig off, and go through some more of the commands available to you.

First, let's try the "Inquiry" mode. Push the right two buttons (#3 & #4) together, then release them, which always puts the keyer in the Inquiry mode. The keyer responds with a "?" in CW. Now, simply send the letter "S" through the paddle. The keyer sends back a number - the speed in WPM the keyer is presently set for. Press buttons #3 & #4 again. (The keyer automatically leaves the Inquiry Mode after each question, so for each new inquiry you must re-enter the inquiry mode.) Now send a "Q" through your paddle. The keyer will respond by sending in Morse either "ON" or "OFF". Now, press the two right buttons, and send the "Q" again. You will get the same answer. The point here is that the Inquiry mode only tells you what the keyer is set for, and does not affect the setting. The Function mode, on the other hand, would have reversed the setting. A little later we will actually discuss the "Q" command.

Press the Inquiry buttons again, and respond to the Morse "?" by sending the number "1" through your paddle. The keyer will play back the message presently stored in memory number one. You can read out the other messages by asking for 2, 3 or 4. If no is message stored, there will be no response.

What is the advantage of entering the Inquiry mode to read out a message memory? Isn't it easier to just push the number 1 button and have it read out? Yes, except that in doing so the stored message will go out over the air. Reading the memory contents through the inquiry mode will play the message back over the monitor speaker, even if the monitor is toggled off, and not out over the air.

Now let's work with the "Function" mode, always activated by pressing the left two buttons, (#1 & #2) down. You will recall using the function mode to toggle the monitor speaker on and off earlier. The function mode is a very powerful tool, controlling a number of other keyer functions we will now explore.

First, let's play with the speed setting. Activate the Function mode by pressing the left two buttons, #1 & #2. The keyer responds by sending "F" in Morse. Now, send the letter "S", then send a "1Ø". Now, regardless of where the speed pot is set, the keyer speed is set at 10 words per minute. Turn the speed control knob up, and the speed will increase from ten words a minute; turn it down and the speed will decrease - but will not go below 6 words per minute. (Be sure to note: To enter a speed below 10 WPM, you must enter a leading zero such as Ø7 for 7 WPM.)

Now, enter the function mode again, send "S 4Ø". Bang. Now the keyer speed is set at 40 words per minute! A little fast for you? Turn the speed down to a comfortable level and reset the speed, again through the function setting for speed. But wait. You say the slowest speed is still too fast for you to accurately control the paddle? Not to worry. That's why the keyer also has the four button speed reset, remember? Set the speed control at the point you want to represent 20 words per minute, (all the way clockwise for the 6 - 20 WPM range) then press all four of the top buttons together, and release. You are back at the speed range you are comfortable with.

While we're on the subject of speed, let's take a look at the keyer speed while you are in the function (or inquiry) mode. Normally, the code speed in the function mode is identical to the regular keyer speed. But, if you like, you can set the function speed at a fixed value, with a speed range available of 6 - 30 WPM. This feature could be valuable to you if you like sending at high speeds, for example, but want a slower, more deliberate speed to access anything through the function or inquiry modes.

To set the function mode speed at a fixed value, enter the function mode and send "F 1Ø". This will set your function mode speed at 10 WPM. Immediately after you have finished sending the "1Ø" the keyer returns automatically to normal mode. If the keyer speed was set at 20 WPM before, you are back at 20 WPM. But now enter the function mode again. The "F" prompt will come back to you at 10 WPM. And now, while you are entering function commands, the keyer is set at 10 WPM. As soon as you are finished with the Function mode, the keyer will return to the normal speed you were at, in this case 20 WPM.

If you decide you prefer the function speed to track the normal keyer speed, which is the default setting for the keyer, you may return to it by entering the function mode, then sending "F ØØ". Now the function mode speed will be set the same as the normal mode speed.

In the first tutorial session we briefly discussed the "Q" setting, which determines whether messages can be "queued" by pressing message buttons in sequence. The default setting is ON, which most operators prefer. To turn it off, enter the function mode and answer the "F" prompt by sending a "Q". The keyer will respond by sending "OFF", indicating that it has turned the queuing function off. Now, press button #1 and then button #2. Memory #1 will start playing, but the instant you push button #2 it will terminate, and the message stored in memory #2 will begin transmission. To restore queuing, again enter the function mode and again send the "Q" command. The keyer will respond with "ON" indicating that queuing has been restored.

Now let's explore the weighting function. Your K-1 Keyer has an extremely accurate weighting capability because it uses digital processing in the CPU to modify the lengths of dots and dashes separately. The default weighting is 50%, the supposedly ideal weighting. And of course, for many users it is. But the weighting can be easily and precisely modified. To do so, enter the Function mode. Answer the keyer's "F" response by sending "W 3Ø" from your paddle. Then send your call sign. Sure sounds different, doesn't it? That's 30% weighting. Now, enter the Function mode again, and this time send the command, "W 7Ø". Send your call again. Yup, it sure did change...

25% through 75% is the maximum range of weighting available, and of course in normal operation these extremes would never be used. However, at higher speeds of sending, some operators prefer heavier weighting, using perhaps 55% or even 60%. Again, operators at slower speeds, particularly in the 6 - 10 WPM range, may prefer a weighting of perhaps 45% or 40% for a more pleasant sound to their ears. If you are inexperienced in such matters, the best advice is to restore the weighting to 50% and leave it until you have a specific reason to change it. You might want to get a friend who is a good CW operator to listen to your signal over the air and advise you.

The weighting setting the keyer is using can also be asked for through the Inquiry mode. To check, simply press the right two buttons to enter the Inquiry mode, and answer the "?" prompt by sending a "W" through your paddle. The keyer will send the spacing percentage it is presently set for.

Some transmitters, unfortunately, do not perfectly reflect the keying supplied to them. The usual problem is that they tend to shorten the length of the dot or dash sent to the rig from the keyer. The reason for such delays is usually to allow QSK circuitry time to function. Such keying errors are the same no matter what speed you are sending at. A recent major transceiver, for example, subtracts 18 milliseconds from all keyed dots and dashes. This gives the listener to your rig an impression of light weighting, especially at higher speeds.

Such transmitter induced error can, theoretically, be cured by adjustment of the weighting towards a heavier value. However, such a correction would be correct only at a single speed setting. Your K-1 Keyer offers a specific correction feature for this problem, the "K" function. To zero out an 18 millisecond error, enter the function mode, then answer the "F" response with a "K 18" command sent through your paddle. The "K 18" adds 18 milliseconds of transmit time to every transmitted dot and dash, thereby canceling out the keying error.

But how do you find out what correction is appropriate for your rig? One way is to ask stations on the air about how your weighting sounds, particularly if it is set at 50%. If other operators tell you it is light, try adding say 10 milliseconds (K 1Ø) and ask for further reports, preferably from the same station. Remember, it takes only a few seconds to change the setting for an experiment.

Another way is to check out magazine product reviews. Some reviews, using photographs of a dual trace oscilloscope, show the actual keying against the transmitter output, thus displaying the keying error. Again, in some cases, the manufacturer actually specifies the variation in the manual.

With a dual trace oscilloscope, determining the amount of compensation needed is simple. But if you have only a single trace scope or a station monitor available, you can still enter the appropriate correction with by experimentation. First, set the keyer weighting to 50 percent. Then (using a dummy load!) send a string of dots, and synchronize the monitor for a stable image showing multiple dots on the screen. Measure the space between dots and compare it to the length of the dots themselves. If the dots are shorter than the spaces in between, experiment with the "K" function value until the transmitted dot length matches the space length. Note that having done this, if you have been using a heavier weighting based on "on the air" comments, you almost certainly will want to change the value to a lighter setting, very probably 50 percent!

The "K" function setting can be read from the "Inquiry" mode if desired by responding to the "T" prompt by sending the letter "K" through your paddle.

Another control available through the function mode is to switch auto spacing on or off. Auto-spacing has always been a bit controversial among CW operators; some operators prefer it while others abhor it. What auto spacing does is force the operator to leave at least three space elements between transmitted letters, so that with auto character spacing letters are not run too close together. When you send a letter and then pause before sending the next letter, the keyer senses that more than one space element has gone by in the timing, and will not begun transmission of the next letter until a full three space elements have passed. Without auto spacing, the operator alone is responsible for the timing of transmitted letters.

The reason some operators dislike it is that if they try to force letters through too quickly, the keyer "stutters" and won't start sending the next letter until the mandated three element spaces have gone by. This causes a feeling of loss of control in the mind of some operators. Other operators relish the extremely precise letter spacing that auto-spacing allows. The K-1 Keyer default is to have auto spacing off. To turn it on, enter the function mode, then answer the "F" prompt by sending "A". Since auto-spacing is a toggled function, this will reverse the existing state, and auto-spacing will be turned on. Once it is on, send a little CW for practice, and see what you think. Auto-spacing is much more noticeable at slower speeds, especially for operators going slower than their normal operating speed. To turn auto-spacing off, again enter the function mode (by pressing buttons one and two, in case you forgot) and answer the "F" prompt by again sending the letter "A" through your paddle.

The on or off state of the auto-spacing switch can also be checked by using the Inquiry mode. Respond to the "?" prompt with the letter "A". The keyer will answer with either "ON" or "OFF". Please, one last comment about auto-spacing on your keyer. If you tried auto-spacing on another keyer and disliked it, try it again on the K-1 Keyer. You will almost surely find the implementation far friendlier than that of any other keyer you have ever used, and you might just change your mind. Give it a try!

Another command available through the function is a choice between "Character Mode" or "Real Time Mode" for loading messages. You already know that when you hold down the memory button to initiate the loading of a message, the keyer responds with a Morse "C", telling you the keyer is in "Character" mode. When we discussed this earlier, we were more interested in getting you up and running, rather than covering all the fine points at once. We stated that we would come back to that "C" for Character mode later - and now is the time.

You already know that to program a message you send a word, and that the keyer sends an "I" to tell you the word has been added, as well as a word space increment, and then accepts the next word. You also learned that you could remove a mis-sent word by sending a string of dots, which the keyer would recognize and would then erase the last word loaded. These techniques allow maximum utilization of the memory, allowing the longest possible message to be loaded. They also are more convenient for most operators.

There is an alternate load mode available, however, called "Real-Time". In Real-Time mode, what you load is mirror-imaged back to you, warts and all. This mode can be useful if you have a special characteristic way of sending something, usually your call sign. For example, if your call is WY9IE, you might prefer a little extra spacing between the "I" and the "E" for emphasis, so the E won't get lost in the shuffle. In Character Mode message loading, you could add a complete word space between the "I" and the "E", but that would be excessive. In Real-Time mode, you get the exact spacing you want.

Real-Time mode has several disadvantages. One is that it is wasteful of memory. You will recall that you can program "The Quick Brown Fox Jumped Over The Lazy Dogs Back" message into each of the four memory channels. However, in Real-Time mode, you can not store such a long message due to the relatively inefficient storage pattern that makes the Real-Time mode what it is. Another disadvantage is that since the Real-Time mode is a mirror image of what you send to the memory, you loose the ability to correct mistakes in loading by sending a string of DITs.

The default load mode is for Character Mode. The command to switch between Character mode and Real-Time mode is a toggle command. To switch it, enter the function mode, then answer the "F" prompt by sending "U", which will toggle you into the opposite load mode, and indicate the new mode with a "C", or an "R" as appropriate. Whenever you load a message by pressing the memory channel button several seconds, you will be informed which memory load mode the keyer is in; the keyer will send either a "C" for Character mode, or an "R" for Real-Time mode. Note that some memories can be stored in Character mode while others are stored in Real-Time mode. The K-1 Keyer will accommodate both at the same time.

You can also ask the keyer which Load mode it is in through the Inquiry mode. Simply enter the inquiry mode by pressing down the two right buttons, and answer the "?" prompt by sending "L" with your paddle. The keyer will respond with either a "C" or an "R".

OK, that's enough for this session. Of course, many of the commands and features we have explored in this section will likely be used lightly or not at all. And that is exactly as it should be. The keyer defaults are set to reflect the desires of most CW operators. But by now you surely have a greater understanding about some of the features of the keyer, and have doubtless customized several of the defaults for your own satisfaction. In the final lesson, we will learn how to set up the message memories to do some really neat and useful tricks, like automatic serial number generation, automatic speed changes in mid-message, closed loop messages etc. But take some time off before you get into that. Get on the air and make some contacts with your keyer, and play with what you have already learned.

THIRD SESSION

Although the features you have learned thus far already put your K-1 Keyer far beyond most keyers, there is one more layer of commands and tools that will really make you and your keyer shine. These features include several more function mode commands, and another type of command, the embedded command. Embedded commands are inserted into programmed messages, and allow pauses, closed loops, calling one message from within another, contest serial numbers, speed changes inside the message and more.

What is an embedded command? It is a special command inserted (through the paddle) into a message being loaded into memory. Embedded commands can only be used when a message is being loaded in the Character mode, rather than the Real-Time mode. An embedded command must always follow a word space, where the keyer sends an "I" indicating the word space. The embedded command starts with an '/' followed immediately by the command.

For purposes of this tutorial only, a "_" in the following text is to be considered a WORD SPACE - not a transmitted character. The underline dash is easier to recognize than a simple space. OK? Good.

How do you program your portable call sign in if the "/" is used for entering an embedded command, you ask? Simple. As long as the slash bar is NOT preceded by a word space, the keyer will recognize it as a slash bar. So if you are signing WY9IE/KH7, no problem at all. Just key it into the memory. (Now, on the other hand, if you want to program in WY9IE - /KH7 with a word space pause, that can be done with the Gap command, which we learn shortly.)

So. Let's try out a message with an embedded command. First, let's call one message from inside another. To do so, first load into memory #1 your call sign and a "K" - "DE WY9IE AR K". Now, load the following into message #2: "CQ_CQ_CQ_/1". (Don't forget that the "_" means a word space in your message load.) Got it stored? OK Now, simply press the #2 button to play out its stored message. Hah! See that? It called and played message #1, running it out with a perfect word space between the two messages.

OK, now try loading a different message into message #1: "CQ_CQ_CQ_DE_WY9IE WY9IE_/1". OK, now play it back. See how it keeps playing over and over again? You have created a closed loop, with the message continuously calling itself. Of course, to stop the message, all you have to do is touch your paddle.

Load into message #2: "12345". Then, load this into message #1: "ABC_DEF_GHI_/2_JKL_MNO". Now play message #1 back. See how message #1 picked up the embedded call to message #2, then, when message #2 had run its course, message #1 continued?

Now, let's have a look at the embedded Pause command, called by "/P*dd*", where *dd* stands for an operator selected number between 00 and 99, and where each digit stands for one tenth of a second. So, load this message into memory #1: "CQ_CQ_CQ_DE_WY9IE WY9IE_AR_/P5Ø_/1", and then play it back. Now you should have the CQ message play, sign your call, and then after a five second pause, repeat again. The /P5Ø is your embedded command into the message to pause 5 seconds, then continue, where continuation in this case calls for a repeat of message #1. And note that you embedded two commands in a row, with each doing its thing. This closed loop message with a pause for listening is great for beacons, contest CQing on a relatively dead band, Meteor Shower skeds etc.

Try this message: "ABC_DEF_GHI_/P5Ø_JKL_MNO". See how the pause allows internal breaks in message transmission. But remember, if you touch the paddle during the pause, the message will be killed, and you are restored to normal mode.

But suppose you want a break in the message play-back so you can insert paddle-sent text? For that, embed a "/B" command, the "B" being for "Break." Load this message: "UR_RST_/B_DE WY9IE" and play it. Notice that the message play back gives you the "UR_RST" then quits. Now, manually enter the RST report - say "579", then release the paddle. Almost immediately after you release the paddle, the "DE WY9IE" will play. The "/B" command opens a window for paddle entry. When the message reaches the "/B" it stops and waits for your entry. While you are sending manually, the keyer monitors your sending, and as soon as it detects a space greater than a word space it picks up where it left off and finishes the message. You may find at higher sending speeds you are encouraged not to tarry on your spacing or the message will try to resume before you are finished, then will cease when it senses your continued paddle closures.

Another embedded command is the "/S" command, which allows the operator to specify the speed in the message. This command can be troublesome, in that the speed it sets in a message becomes the speed the keyer is set at when the message has completed transmission. So, let's say you have the keyer going at 20 words per minute, then send a message with an embedded command calling for a speed of 30 words per minute. When the message is finished, you will find your keyer is now set at 30 WPM. You could embed a second command to reset the speed to 20 WPM at the conclusion of the message, but if for any reason you change to a speed other than 20 WPM, as you likely will, you will find the speed control range upset again. None the less, the ability to call speeds inside the message can be useful. Try entering this message, then play it back:

"/SØ6_CQ_CQ_/S12_CQ_CQ_/S24_CQ_CQ_/S48_DE WY9IE_/SØ6".

Note that for speeds under 10 WPM, you must use a leading zero, as in "Ø6".

A pair of embedded speed controls will probably be more useful for many operators, the /SU*d* and /SD*d* commands. These commands increase and decrease the speed of the keyer from its current setting, with an available range of 1 - 9 WPM change. These commands will be especially useful in exchanges with "canned" responses, like "59904" or "599KW".

Try loading this message: "/SU9_599Ø4_73_/SD9", then set your keyer to perhaps 10 WPM and send through the paddle: "ZA1DX DE WY9IE", then press the message button. When you return to the paddle, you will find you are back at 10 WPM or whatever speed you started at. As you can see, thus speeding up an expected message can give you more QSO's per hour in a contest situation, and thus can be extremely valuable. When embedding such commands, remember, any time you use an "/SU*d*" command you will almost certainly want to use a complimentary "/SD*d*" to restore the keyer to the speed it started at.

Another interesting trick is to load a simple message "/SU3" in one memory, and "/SD3" in another. Now, you have push button speed control, raising the keyer speed 3 WPM each time you press one button, and lowering the speed 3 WPM each time you press the other.

One other embedded command can be useful for a number of functions - the "/G*d*", or "Gap" command. Remember when we talked about sending the call "WY9IE" and trying to put extra space between the "I" and the "E". In a simplistic character mode message, that is impossible; the only option being a whole word space. The other simplistic option is to store a message in the "Real-Time" mode.

But with the "/G*d*" command, (The "*d*" stands for "digit") it is possible to set an exact "gap" between the "I" and the "E". Here's how it works. A normal space between two characters is 3 "bits" long. (A bit is the same length as a single dot.) The "/G*d*" embedded command adds as many bit spaces as you want (up to 9) to a standard letter space. Remember, the minimum space is three bits long, and the G*d* command adds to that basic space. So, a /G1 embedded command will lengthen a space between two letters only a small amount. A /G4 command will open up the space from a letter space to a word space. Try programming "WY91_/G2_E" and see how it sounds. Experiment with different values for /G*d*. Once you understand it, you will almost surely use it for one special situation or another.

Now, let's get into serial number generation, primarily useful for contest operation. The K-1 Keyer serial number controls are unmatched by any other keyer design available.

First, you must decide what number format you want to use. Do you send your zeros in contest serial numbers using five dashes, three dashes, or a single dash? Do you use the same zero format for a leading zero as for other zeros? Or do you not send leading zeros? What about nines? Do you spell out a proper nine, with four dashes and a dot, or do you follow the now common format of using an "N", as in "5NN"? The K-1 Keyer "Z*d*" function command lets you specify your choice. Examine the Z Option table in the accompanying instruction manual, and select the serial number format you prefer.

Let's suppose you decide on option #6, with both leading zeros and other zeros sent as a "T". Now, enter the function mode of the keyer and respond to the "F" prompt with the command, "Z6". This will cause generated serial numbers to reflect that format.

Now that that has been set, you also have the option of presetting a serial number. If you do not do so, the first serial number sent will be "TT1 ". If you do want to preset a number, say 1066, enter the function mode again, and answer the "F" prompt with the command, "N 1Ø66". Note that you must use a five-dash zero in "1066" even though the keyer will send it as a "T" under the "Z6" function. Now, enter the inquiry mode by pressing buttons 3 & 4, and respond to the "?" prompt with an "N". The keyer will send back "1T66", CW shorthand for 1066. If you want to reset the serial number at a low number such as "23", you must use leading five-dash zeros to create a four digit number "ØØ23" so that the keyer will accept the new number.

So how do we use the serial number in a contest message? Right - with an embedded command. So let's enter one: "R_TU_5NN_/N_BK". This, when sent, will give you "R TU 5NN 1T66 BK". "But," you say, "There is a word space between the 5NN and the serial number. And that is wasted time in a contest where time is of the essence." Right! So let's get rid of that with the embedded /G*d* command. So, let's reprogram the message: "R_TU_5NN_/GØ_/N_BK". Now we have eliminated the word space between the 5NN and the serial number. It will not take long for the contest operator to appreciate the flexibility these commands give him.

But, you say, "What do I do when some clown calls me for a duplicate QSO and I send the serial number before I realize he is a dupe? Do I have to reset the serial number with the 'N' function command? That would be a pain!"

Indeed it would be. But you don't have to reset the number. All you have to do is press the middle two buttons, (2 & 3) and the keyer sends a "D" and will automatically decrement one number! That's all there is to it.

There is one last button function we have not covered - another tune-up approach. Pushing buttons 2 & 4 together and releasing them "closes" the key contacts to the transmitter for tune up purposes. The rig stays keyed until you touch either side of your paddle, which releases the rig.

If you examine the regular operating manual, you will note that some of the commands discussed above can be entered in several ways. For example, the decrement command can be sent through the function mode. But pressing the two center buttons is so much easier it would be a rare operator that prefers the function mode entry for this command. For this reason, the function mode method of decrementing was not covered in the tutorial. (Till now, anyhow!) None the less, you will find it and similar commands in the Operating Manual.

So. That concludes the tutorial for your K-1 Keyer. We know you will be delighted by this fine product. Enjoy it, and be sure and tell your friends about it as well. 73!