Introduction

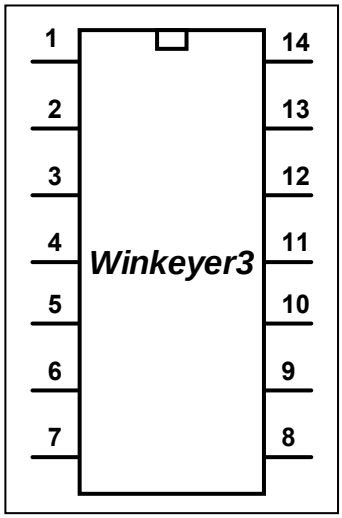
This document will describe the Winkeyer3 (WK3) IC and its' various interfaces. WK3 is a third generation single chip Morse keyer IC. It is designed to attach to a PC's USB port and provide accurate transmitter keying for Windows based logging or other ham radio software packages. Due to timing latency inherent in the multi-threaded Windows operating system, it is difficult for a PC to generate accurately timed Morse. WK3 buffers ASCII characters sent by a Windows based software application. It then translates them to Morse and directly keys a transmitter or transceiver. In addition, WK3 has a paddle input allowing an operator to break in and send directly at any time. WK3 also provides a speed potentiometer interface so that an operator can instantly dial any speed desired.

The host PC communicates to WK3 over a simple serial interface which can be a COM or USB port. Letters to send, along with operational commands, are sent from the host to WK3 over the serial link. A substantial feature list is provided allowing the user to precisely tailor WK3's keying characteristics to a particular transmitter. WK3 has a very low power requirement; in fact, it was originally designed to be powered from a PC's serial port. In standby it draws under a micro amp.

K1EL sells several kits that utilize the WK3 IC, refer to the k1el website: www.k1el.com for further information. One popular kit is WKUSB which includes a USB interface, speed control, enclosure with pushbuttons, and an internal battery pack for standalone use.

Features

- 1200/9600 Baud Serial Rx/Tx Interface
- Iambic CW Paddle Interface
- Two Key Output Ports (high true TTL)
- Two PTT Output Ports: (high true TTL)
- 25 ma output sink/source
- Adjustable PTT lead in and tail delays
- Adjustable Speed 5-99 WPM
- Adjustable Weighting and dit/dah ratio
- Adjustable Farnsworth Character Spacing
- Adjustable Keying Compensation
- Autospace and Letterspace Control
- Sidetone output with adjustable frequency
- Standalone K14 Keyer Emulation
- Paddle only sidetone
- Custom configuration for two users
- Dit/Dah Memory Control
- Single Op Two Radio (SO2R) Support
- Adjustable Paddle Switchpoint
- Iambic A, B, Ultimatic & "Bug" modes
- Speed Pot Interface
- Adjustable speed pot range
- Embedded commands
- 160 character input buffer
- No crystals or oscillators
- 3.3 to 5 volt operation
- Current Draw: < 2 ma in active operation
- HSCW and QRSS Capability
- Automatic power down sleep mode
- Twelve stackable memory slots
- Stored Callsigns



Pin 1 – Vcc (5.0 volts)
Pin 2 – Port 2 Key Output
Pin 3 – Port 2 PTT Output
Pin 4 – USB Connected Sense
Pin 5 – Serial Receive Input
Pin 6 – Serial Transmit Output
Pin 7 – Port 1 Key Output
Pin 8 – Sidetone
Pin 9 – Port 1 PTT Output
Pin 10 – Speed Pot Analog Input
Pin 11 – Right Paddle Input
Pin 12 – Left Paddle Input
Pin 13 – Switch Array Input
Pin 14 – Vss (Ground)

Figure 1 – Winkeyer3 Package & Pinout

Theory of Operation

This section will describe how WK3 works. As shown in Figure 1, the host PC is connected to WK3 over a serial COM port, which can be a USB port supporting a virtual COM port. WK3 is a slave to the PC in that it receives commands and data from the PC. The PC can send commands while WK3 is sending Morse allowing dynamic configuration changes. There are four reasons WK3 will transfer data back to the host:

1) Inform the host of a status change in WK3.
2) Inform the host of a speed pot or pushbutton change.
3) Respond to a request for status from the host.
4) Echo back Morse in ASCII as it's being sent from either the serial port or the paddles.

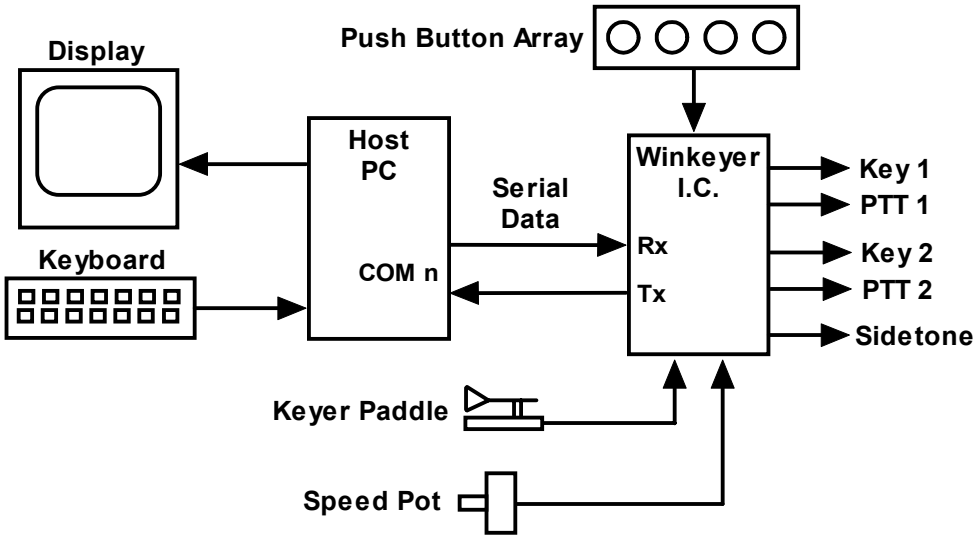


Figure 2 – WK3 to PC Connection

There are two types of serial data transfer from the host to WK3: Command and Data. Commands modify WK3's operation in some way, for example changing operating speed, pausing transmission, or asking for status. Data can be letters, numbers, or prosigns that are to be sent in Morse. Commands and data are processed differently in WK3. Data is put into a serial buffer that allows the host to send data ahead of the Morse being sent. The size of this buffer is 160 characters and is a FIFO which is an acronym for First In First Out. This means that characters are taken out in the order they were put in. Since there can be a considerable delay from host input to Morse output, commands bypass the input FIFO and are acted upon immediately. This allows changes to be made while sending is underway.

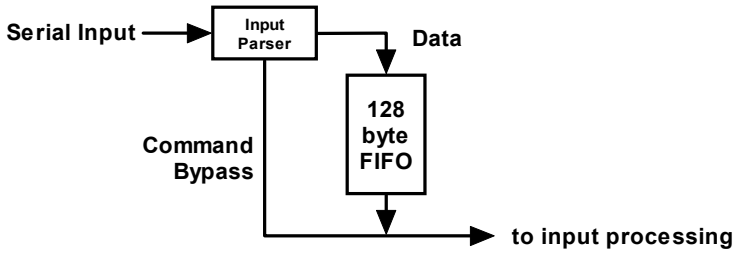


Figure 3 – Data and Command Flow inside WK3

Since there are times when you don't want commands to take effect immediately, WK3 allows commands to be buffered. This means that the command is placed in the serial buffer and won't be acted on until it comes out of the buffer. An example of the use of a buffered command would be to send two words at two different speeds, the first at 15 WPM and the second at 20 WPM. By placing a buffered speed command between the words, the speed will not be changed until the first word is completely sent. Not all, but many of WK3's immediate commands can be entered as buffered commands.

Communication from WK3 to the host operates in a loosely coupled manner. This means that the host never issues a command and waits for a response. Instead, the host sends a request for information and WK3 queues this request and will respond to the host when processing time allows. WK3 processes tasks in parallel, there may be other bytes waiting to be sent back to the host before the latest request can be handled. Rather than wait for a return, the host should divide its WK3 driver interface into two parts, one part that issues command bytes and a second part that checks for returned bytes and processes them when they arrive. Following is a bit of pseudo-code that illustrates this concept. It will make more sense as you learn about the WK3 command set.

```
Serial Comm Thread {
    while (1) {
        if (host has a command to send to WK3) {
                send command to WK3;
        }
        else if (WK3:uart_byte_ready) {
                wkbyte = WK3:uart_read();
                if (( wkbyte & 0xc0) == 0xc0 {
                        it's a status byte.   (Host may or may not have asked for it.)
                        process status change, note that it could be a pushbutton change
                }
                else if ((wkbyte & 0xc0) == 0x80) {
                        it's a speed pot byte (Host may or may not have asked for it.)
                        process speed pot change
                }
                else {
                        it must be an echo back byte
                        if (break-in==1) { it's a paddle echo }
                        else { it's a serial echo }
                }
        }
    }
}
```

Notice that unless WK3 has something for the host to read, the host continues to process outgoing commands and other tasks. Also note that speed pot and status bytes can be unsolicited, in other words WK3 can send these at any time a state change occurs inside WK3. Echo back bytes are also unsolicited as they are based on asynchronous Morse sending. The host has to be able to handle these as they occur. If host processing is not immediate, a serial input buffer on the host side is required to make sure no returned bytes are missed.

Paddle Input Priority

WK3 accepts input from either its serial port or iambic paddle port. Paddle input will always take priority and will interrupt serial data, automatically clearing WK3's serial input buffer. When a paddle break-in occurs, any additional serial data that arrives from the host will be processed, but will be ignored unless it is an immediate command. After paddling ceases, WK3 will pause for one word space time before it resumes serial data transmission.

Standalone Keyer Mode

WK3's primary purpose is to provide accurate Morse keying to a Windows based application. The most often requested feature from users of WK1 was to also allow WK to be run independently, not connected to a computer. In response, all subsequent Winkeyer ICs provide standalone operation.

In most respects the standalone keyer is completely separate from the host mode keyer. This means that the standalone keyer has its own configured state that is overwritten when changing to host driven mode. Configuration changes issued while under host mode will be cancelled when returning to standalone mode. A simple Windows application is available from K1EL called WK3MGR which can be used to set the standalone configuration and message contents from the PC. Alternatively, settings and messages can be entered by paddle commands in standalone mode.

Power Up Default State

On power up, WK3 comes up in standalone mode and stays in that mode until it receives a Host Open command from a PC host. At that time, standalone mode is suspended. When the host takes over, it normally downloads a block of initialization parameters (see Load Defaults command) to set the operating state as desired and to sync WK3 settings with those of the host. When shutting down, a host application that has opened WK3 should always issue an Admin:Close command to return WK3 back to standalone mode. This allows the keyer to be used in standalone mode while still attached to the PC. When WK3 is physically disconnected from the host it automatically goes into standalone mode even if a Close command was not issued.

While WK3 is attached to a PC com port in the closed state, it will accept ADMIN commands. This allows an application such as WK3MGR to access standalone messages and settings.

Winkeyer3 Lockup Recovery

After WK3 is connected to a host it should not be physically disconnected while the host application is active. Accidents do happen and if the USB cable is pulled in the middle of a command or data exchange WK3 can get locked up. It's not very likely but it can happen. A provision is included to easily get WK3 back in operation again. Press and hold the command button until WK3 responds with a restart response. If WK3 was stuck in host open a **C** will be sent in sidetone. If WK3 was locked due to a different problem (very unusual) it will echo an **R**.

Pushbutton Notification

Starting with WK version 2.0, the pushbuttons can be used in host mode. Normally pushbutons are used in standalone mode only. One pushbutton is designated as the Command push button and is used to initiate paddle commands. The other pushbuttons are used to enter or play recorded Morse messages. The host must issue an Admin:Set WK2 Mode command to enable pushbutton notification. Once the mode is sent, pushbutton status will be returned in the WK2 status byte. See page 13 for further information.

USB Sense

WK3 was designed with USB interfacing in mind. A USB sense input is provided which should be asserted high when the USB port is attached and active. If the USB port is switched to standby or disconnected from the PC, USB Sense should be pulled low. When low, WK3 is allowed to go into low power sleep mode. Note that while connected to the host with USB sense high, WK3 will not go into low power standby.

Versions later than 22 were changed to handle the case where the host PC goes into standby while a host application is actively connected to WK. If this happens, WK will not disconnect and will remain in host mode until the PC comes out of standby. This maintains the link between the host application and WK so that WKUSB can be used as soon as the PC wakes up and leaves standby mode.

Firmware Update

The firmware image within WK3 can be upgraded in the field. When K1EL releases a firmware update, it can be uploaded to the WK3 IC through the WK3MGR utility over USB. Updates will be available from www.k1el.com/wkupdates.html

Host Mode Command Descriptions

This section documents the commands that are sent from the host to WK3 over the serial/USB interface. Commands are special hex codes that are sent to WK3. These codes range from 0x01 through 0x1F. In this document a hex value will be presented in angle brackets, for example **<02>**. Some commands have one or more parameters sent immediately after the command code is sent, this will be documented as the command code followed by a value: **<02><nn>** where nn is a single byte decimal value. The notation **[c]** represents a single ASCII character sent to WK3 as a single serial byte.

Immediate Commands

These commands are processed as soon as they are received, they bypass the input buffer.

● **Admin Command <00><nn> nn is a value from 0 to 20**

After power-up the host interface is closed and serial status, echo, or pot change data will be not be sent to the host. The only commands WK3 will accept are Admin commands. Admin commands are received, processed and any return status or data will be sent back immediately. There are many Admin commands andthey serve various purposes. With the exception of the Amin:Close command, all Admin commands should only be issued while the host interface is closed. Following are descriptions of the Admin commands:

0: Calibrate This is an historical command preserved for WK1 compatibility. The command is accepted but not processed by WK3. The command syntax is: <00><00> pause 100 mSec <FF>

1: Reset Resets the WK3 processor to the power up state. Do not send this as part of the initialization sequence. Only send this if you want to do a cold reboot of WK3.

2: Host Open Upon power-up, the host interface is closed. To enable host mode, the PC host must issue the Admin:open <00><02> command. Upon open, WK3 will respond by sending the revision code back to the host. The host must wait for this return code before any other commands or data can be sent to WK3. Upon open, WK1 mode is set.

3: Host Close Use this command to turn off the host interface. WK3 will return to standby mode after this command is issued and standby settings will be restored.

4: Echo Test Used to test the serial interface. The next character sent to WK3 after this command will be echoed back to the host. <00><04><65> echoes A.

5: Paddle A2D Historical command not supported in WK3, always returns 0.

6: Speed A2D Historical command not supported in WK3, always returns 0.

7: Get Values Historical command not supported in WK3, always returns 0.

8: Reserved ***K1EL Debug use only***

9: Get Cal Historical command not supported in WK3, always returns 0.

10: Set WK1 Mode Disables pushbutton reporting

11: Set WK2 Mode Enables pushbutton reporting, alternate WK status mode is selected.

12: Dump EEPROM Dumps all 256 bytes of WK3's internal EEPROM.

13: Load EEPROM Download all 256 bytes of WK3's internal EEPROM.

14: Send Standalone Message Command WK3 to send one of its internal messages. The command syntax is: <00><14><msg number> where msg number is 1 through 6

15: Load X1MODE Load mode extension register 1, WK1 does not support this register. Note that the bit assignments of this register are different between WK2 and WK3 mode

Value nn Bits	Function
7-4 (MSN)	Letterspace Adjustment, upper 4 bits 0 to 15 in 2% increments
3	Standalone cut mode, send 0 as T and 9 as N when set to 1 (ignored in host mode)
2	Unused
1	Enable Paddle Status when = 1
0	Unused

Table 1 – Extension Mode Register 1 (X1MODE) in WK2 Mode

Value nn Bits	Function
7 (MSB)	Select User 0 or 1
6	Select Message Bank 0 or 1
5	Select tune 50% duty when = 1
4-0	Letterspace Adjustment, lower 5 bits 0 to 15 in 2% increments

Table 2 – Extension Mode Register 1 (X1MODE) in WK3 Mode

16: Firmware Update This command initiates an image upload. This feature is protected.

17: Set High Baud Change Baud Rate to 9600 baud

18: Set Low Baud Change Baud Rate to 1200 (default)

Baud rate change must be handled in a specific way. Since most applications expect WK3 to run at 1200 baud, this is always the default and will be reinstated whenever WK3 is closed. If an application wants to run at 9600 baud, it must start out at 1200 baud mode and then issue the Set High Baud command. When the application closes it should issue a WK close command which will reset the baud rate to 1200.

19: Set K1EL Antenna Switch Function not currently documented, reserved for future release.

20: Set WK3 Mode Enables Winkeyer 3 functions; expanded X1MODE and additional X2MODE register

21: Read Back Vcc Return WK IC power supply voltage. This command returns a single byte which can be converted to voltage: 26214/byte value = Voltage*100

22: Load X2MODE Load mode extension register 2. This register is active on WK3 mode only.

Value nn Bits	Function
7 (MSB)	Paddle status On when set
6	Fast command response on when set This applies to standalone PB press.
5	Cut 9 : Substitute N for 9 when set
4	Cut 0: Substitute T for 0 when set
3	Paddle only sidetone when set
2	SO2R mode (PTT1 = radio select)
1	Paddle Mute when set
0	Spare

Table 3 – Extension Mode Register 2 (WK3 Mode Only)

End of ADMIN Commands

● **Sidetone Control <01><nn> nn is a value described below**

WK3 Pin 8 functions as the sidetone square wave output and supports the legacy WK1 and WK2 sidetone schemes. When not in WK3 mode the sidetone selections are governed by table 4 and 5 below. *Note that WK3 frequencies will be slightly different than those shown in the table.*

Value nn Bits	Function
7 (MSB)	Enable Paddle Only Sidetone when = 1
6-4	Unused set to zero
3-0	Sidetone frequency N (See Table 4 below)

Table 4 – WK1 and WK2 Sidetone Control Assignments

N	Frequency		N	Frequency
0x1	4000 Hz		0x6	666 Hz
0x2	2000 Hz		0x7	571 Hz
0x3	1333 Hz		0x8	500 Hz
0x4	1000 Hz		0x9	444 Hz
0x5	800 Hz		0xa	400 Hz

Table 5 – WK1 and WK2 Sidetone Selection Table

The most significant bit of the frequency byte controls the paddle only sidetone feature which was introduced in WK2. You can choose to only use sidetone for paddle entry and mute it for CW sourced from the host port. This is called Paddle Only Sidetone and is selected by setting the MSB of the sidetone control value.

In WK3 mode, the sidetone frequency is continuously adjustable between 500 and 4000Hz. All bits of the sidetone control register are used. The formula for the value is 62500/frequency. For example: nn = 62 for 1000 Hz. Due to integer round off, the setting is approximate and will be within 5% of the frequency specified. In WK3 mode, the paddle only sidetone control bit is relocated to the X2MODE register.

● **Set WPM Speed <02><nn> nn is in the range of 5-99 WPM**

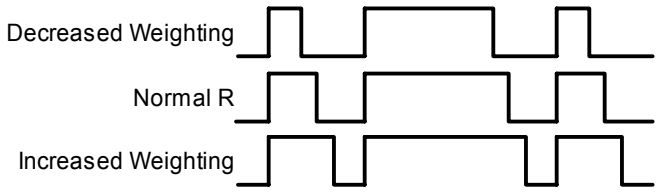
Example: <02><12> set 18 WPM

Set a new Morse operating speed, this command takes effect as soon as WK3 receives it. If the speed is set to zero then WK3 will take its speed setting directly from the speed pot, this is the reset default.

● **Set Weighting <03><nn> nn is in the range of 10-90%**

Example: <03><32> for weight=50

This command allows a proportional amount to be either added or subtracted from the length of all transmitted dits and dahs. A value of 50 (0x32) 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.



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

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

Figure 4 - Weighting Example

● Set PTT Lead/Tail <04><nn1><nn2> nn1=lead in time, nn2=tail time

Example: <04><01><A0> lead-in=1, tail=160
Values can be 0 to 250 in 10 mSecs steps

WK3 provides a transmitter PTT output for each key 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. These delay settings apply to both key ports. The trailing delay is handled differently for CW sent by paddle and CW sent by "machine". Paddle delay is controlled by the Hang Time setting in the PINCFG register (see the *SetPinConfig* command). The Tail setting determines the delay used for CW sent by an internal message or CW sent by a Host application. 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
```

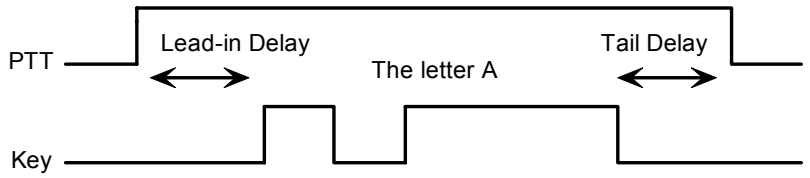


Figure 5 – PTT Lead-in and Tail Example

A detailed description of PTT functionality can be found on page 15.

● Setup Speed Pot <05><nn1><nn2><nn3> nn1 = MIN, nn2 = RANGE, nn3 = 0

This command sets the limits for the speed pot. MINWPM sets the lowest value returned; WPMRANGE indirectly specifies the maximum value returned. For example if MINWPM=10 and WPMRANGE=15, the full pot swing values, min to max, would be 10 to 25 WPM. Note that the max value is MINWPM+WPMRANGE. The value of the third parameter is not used but it must be included to maintain backward compatibility for applications supporting only WK1 keyers. Recommendation is to set this to zero but any value is accepted.

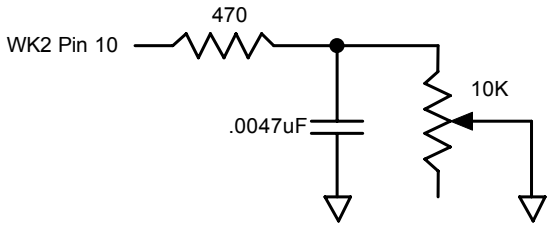


Figure 6 – Speed Pot Circuit

● Set Pause State <06><nn> nn = 01 pause, value = 00 unpause

When WK3 is paused, sending will stop immediately and will not resume until pause is cancelled. The current character being sent in Morse will be completed before pausing. The Clear Buffer command will cancel a pause state.

● Get Speed Pot <07> Return Winkeyer3's current speed pot setting

This command will cause a speed pot command request to be queued which will be acted on as soon as possible. Depending on current processing load, the pot status byte will be sent within 200 milliseconds after command receipt. The application should not wait for a response but process the returned data in an unsolicited status handler. The returned value will range from 0 to 31 and will be the actual speed pot value minus the MIN_WPM setting. The speed pot is windowed into a 32 step range between 5 to 99 WPM (see Setup SpeedPot command). The two MS Bits of a Speed Pot status byte will always be b10:

1	0	6 bit value in WPM

Table 6 - Speed Pot Status Byte Format

● Backspace <08> Backup the input buffer pointer by one character.

This command is only meaningful if there is something in the serial input buffer, otherwise it is ignored.

● Set PinConfig <09><nn> Set the PINCFG Register

Low nibble determines how output pins are mapped
High nibble controls ultimatic mode and hang time

The original WK1 IC was in an 8 pin package in which Pin 5 was a shared resource, it could be assigned as a PTT output, a Sidetone output, or a secondary Key output: If it was assigned as a PTT output, that meant it was not possible to output sidetone. Likewise if it was assigned as a secondary Key output, sidetone or PTT were not allowed. When WK was moved to a 14 pin package, separate outputs were provided.

Bit 7-4	Bit 3	Bit 2	Bit 1	Bit 0
See Below	Pin5 KeyOut Enable	Pin3 KeyOut Enable	Pin 5 Sidetone Enable	Pin5 PTT Enable

Table 7 – WK1 PINCFG Format (for reference)

In the 14 pin package, there is a dedicated sidetone pin, two PTT outputs, and two Key outputs. The pin config assignments are compatible with WK1 for backwards software compatibility. For example Bit 0 specifies whether PTT should be used, Bit 1 specifies whether Sidetone is enabled, and Bits 2 and 3 select which key port is active. If Key Port 1 and PTT is enabled, PTT1 will be asserted in sync with KEY1. If Key Port 2 is enabled, KEY2 and PTT2 are activated.

Bit 7-4	Bit 3	Bit 2	Bit 1	Bit 0
See Below	KeyOut 1 Enable	KeyOut 2 Enable	Sidetone Enable	PTT Enable

Table 8 – WK3/WK2 PINCFG Format

The WK2/WK3 PINCFG register also includes two additional features; Ultimatic Dit/Dah priority and Paddle hang time. These settings are allocated to the upper four bits as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3-0
Ult Priority 1	Ult Priority 0	Hang Time 1	Hang Time 0	See Above

Table 9 – WK1/WK2/WK3 Priority/Hang Time Format

Ultimatic Priority = 00 Normal Ultimatic
Ultimatic Priority = 01 Send dahs when both paddles are pressed in Ultimatic mode
Ultimatic Priority = 10 Send dits when both paddles are pressed in Ultimatic mode
Ultimatic Priority = 11 Undefined
Hang Time = 00 Wait 1 wordspace + 1 dit before ending paddle insertion

Hang Time = 01 Wait 1 wordspace + 2 dits before ending paddle insertion
Hang Time = 10 Wait 1 wordspace + 4 dits before ending paddle insertion
Hang Time = 11 Wait 1 wordspace + 8 dits before ending paddle insertion

Hang Time is similar to tail time in that it holds PTT on between paddle presses, but it is proportional to sending speed since it is measured in word space time while tail time is mostly a fixed delay.

● Clear Buffer <0A> Reset Input Buffer Pointers

This command will reset the input buffer pointers to an empty state. In addition, Tune and Pause are cancelled by this command.

Clear Buffer can be sent at any time to abort a message, abort a command, or to clear the serial buffer. It will cancel any Morse character in progress immediately ending it in midstream if necessary.

● Key Immediate <0B><nn> nn = 01 key down, n = 00 key up

Use this command to implement a tune function. Once asserted, key down will remain in effect until either a key immediate with a zero value is received or the internal tune watchdog timer expires. The tune timer is hard coded to a value of 100 seconds and cannot be disabled. Key down can be aborted either by the paddles or by a clear buffer command.

● Set HSCW <0C><nn> nn = the lpm rate divided by 100

WK3 supports HSCW (High Speed CW) transmit rates through the use of this immediate command. .

For example nn=20 selects 2000 lpm and nn=35 selects 3500 lpm. Any rate from 1000 to 8000 can be picked but only a few are used by radio amateurs. In the US, common rates are 1000, 2000, 4000 and 6000 lpm while in Europe 1000, 1500, 3000, 4000 lpm are common.

● Set Farns WPM <0D><nn> nn is in the range of 10-99

Example: <0D><12> for Farnsworth=18 WPM

Farnsworth spacing is useful for CW practice because it encourages you to learn characters by sound not individual dits and dahs. When Farnsworth is enabled, letters are sent at a fixed rate of **nn** WPM while spacing between characters is determined by the sending WPM rate. When the WPM rate is set above the Farnsworth WPM, Farnsworth is automatically disabled.

● Set Winkeyer Mode <0E><nn> nn = Mode bit field in binary

Example: <0E><13> set bits 4,1,0, clear the rest

The operational mode of WK3 is controlled by the mode register.

Mode Bit	Function
7 (MSB)	Disable Paddle watchdog
6	Paddle Echoback (1=Enabled, 0=Disabled)
5	Key Mode: 00 = Iambic B 01 = Iambic A
4	10 = Ultimatic 11 = Bug Mode
3	Paddle Swap (1=Swap, 0=Normal)
2	Serial Echoback (1=Enabled, 0=Disabled)
1	Autospace (1=Enabled, 0=Disabled)
0 (LSB)	CT Spacing when=1, Normal Wordspace when=0

Table 10 – WK3 Mode Selection Table
The WK3 mode register is cleared at reset.

Bit 7
WK3 has a paddle watchdog counter that disables the key output after 128 consecutive dits or dahs. WK3 assumes that this is an unintended condition and keying outputs are turned off. Sidetone remains on to alert the user. Paddle watchdog is on by default but it can be turned off by setting this mode bit.

Bit 6
When this bit is set to one all characters entered on the paddles will be echoed back to the host. From the host's perspective, paddle echo and serial echo are the same; in both cases a letter sent in Morse by WK3 is echoed back to the host. The echo occurs after the letter has been completely sent. The host can determine the source by the sense of the "break-in" status bit. If this bit is high when the echoed letter comes in, then the letter's source was from the paddles, if break-in is low the source if from the serial port.

Bit 5, 4
WK3 supports iambic A, B, Ultimatic, and Bug keying modes. In iambic mode, WK3 sends dits and dahs automatically based on which paddle you press. In bug mode WK3 sends the dits and you manually send the dahs. You also can use bug mode to operate in straight key mode or if you want to key through WK3 with a different keyer, simply set bug mode and use the dah input to key WK3.

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 Ultimatic mode when both paddles are pressed the keyer will send a continuous stream of whichever paddle was last pressed.

Bit 3
This bit swaps the operation of the dit and dah paddles. It's used mostly to accomodate left handed ops or to correct for a miswire in the keying cable.

Bit 2
Serial Echo Back tells WK3 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 WK3's progress in real time. Note that buffered commands and their parameters are not echoed back to the host.

Bit 1
Autospace works in the following way: If you pause for more than one dit time between a dit or dah, WK3 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. WK3 has a paddle event memory so that you can enter dits or dahs during the inter-letter space and WK3 will send them as they were entered. With a little practice, autospace will help you to send near perfect Morse.

Bit 0
When contest spacing is enabled, wordspace time is reduced by one dit. Instead of 7 dits per wordspace, contest spacing selects six dits per wordspace.

● Load Defaults <0F><value list> value list is a set of 15 binary values

This command is used to load all of WK3's operating parameters to be loaded in one block transfer. The values are binary and must be loaded in order. The values are exactly the same as those loaded for the individual commands. The preferred time to issue this command is at reset just after the interface has been opened. Do not issue this command while WK3 is transmitting.

1) Mode Register	2) Speed in WPM	3) Sidetone Frequency
4) Weight	5) Lead-In Time	6) Tail Time
7) MinWPM	8) WPM Range	9) X2 Mode
10) Key Compensation	11) Farnsworth WPM	12) Paddle Setpoint
13) Dit/Dah Ratio	14) Pin Configuration	15) X1 Mode

Table 11 - Default Value List in order of issuance

● Set 1st Extension <10><nn> nn is in the range of (0 to 250) × 1 mSecs

This command is no longer supported in WK3, it will be accepted but is ignored.

● Set Key Comp <11><nn> nn is in the range of (0 to 250) × 1 mSecs

Example: <11><B4> sets key comp to 180 mSecs

Keying Compensation allows a fixed amount to be 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 with compensation on.

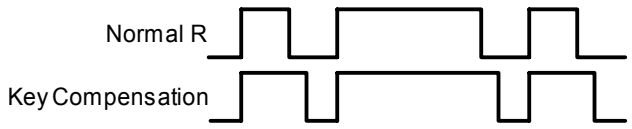


Figure 7 – Keying Compensation Example

● Set Paddle Switchpoint <12><nn> nn is in the range of 10-90%

Example: <03><37> for a switchpoint = 55 decimal

This setting controls when WK3 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. Faster operators report a setting somewhat less than default is optimal. 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.

● Null Command <13> This command is ignored

In some situations there is a need to insert a placeholder in the serial data stream. This command provides that function. It occupies a stream position but is ignored by the Morse processor.

● Software Paddle <14><nn> nn = 00 paddle up, n=01 dit, n=02 dah, n=03 both

This command provides a means to assert the paddle input from the host. The host application can implement a keyboard paddle by converting PC up/down key codes to Software Paddle commands. Due to the slow response time of the keyboard, operating system, and serial communication, it can be a challenge to get this scheme to work above 20 WPM. The paddle watchdog guards this interface in the same way as normal paddle input.

● Request Winkeyer Status <15> Return Winkeyer status byte

This command queues a request to WK3 to send its current operating state. The status byte returned consists of a bit field that is defined by one of the following tables, as determined by the current WK mode. The three MSBs of the status byte are always b110. Note that in WK2 mode mode bit 3 identifies the status byte as a pushbutton status byte. WK2 and/or WK3 mode is set by ADMIN commands (see pages 5 & 6).

Status Bit	Name	Definition
7 (MSB)	Tag	1
6	Tag	1
5	Tag	0
4	WAIT	WK is waiting for an internally timed event to finish
3	KEYDOWN	Key down status (Tune) 1 = key down
2	BUSY	Keyer is busy sending Morse when = 1
1	BREAKIN	Paddle break-in active when = 1
0 (LSB)	XOFF	Buffer is more than 2/3 full when = 1

Table 11 – Winkeyer Status Definition WK1 compatible mode

Status Bit	Name	Definition
7 (MSB)	Tag	1
6	Tag	1
5	Tag	0
4	WAIT	WK is waiting for an internally timed event to finish
3	0	This is a WK Status Byte
2	BUSY	Keyer is busy sending Morse when = 1
1	BREAKIN	Paddle break-in active when = 1
0 (LSB)	XOFF	Buffer is more than 2/3 full when = 1

Table 12 – Winkeyer Status Definitions (WK2 Mode)

Status Bit	Name	Definition
7 (MSB)	Tag	1
6	Tag	1
5	Tag	0
4	PB4STAT	1 when PB4 pressed, 0 when PB4 unpressed
3	1	This is a pushbutton status byte
2	PB3STAT	1 when PB3 pressed, 0 when PB3 unpressed
1	PB2STAT	1 when PB2 pressed, 0 when PB2 unpressed
0 (LSB)	PB1STAT	1 when PB1 pressed, 0 when PB1 unpressed

Table 13 – Winkeyer Pushbutton Status Definitions (WK2 Mode)

A PB status byte is sent whenever the PB state changes, in other words a byte will be sent when a pushbutton is pressed and a byte will be sent when the pushbutton is released. Only one press will be detected at a time.

● Pointer Command <16><nn> Input Buffer Command Set

This command allows the host app to manipulate the input buffer for special situations such as “on the fly” callsign correction. Four commands make up the pointer command set:

nn=00 Reset input buffer pointers to start of buffer, only issue this when buffer is empty.
nn=01 Move input pointer to new position in overwrite mode
nn=02 Move input pointer to new position in append mode
nn=03 Add multiple nulls to the buffer <16><03><number of nulls>

A detailed description of the pointer command is detailed in a separate application note.

● Set Dit/Dah Ratio <17><nn> nn is in the range of 33-66 decimal

Modifies the ratio of dit time to dah time. Standard ratio is 1:3 (dit:dah) when **nn** = 50

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. This causes an intentional 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 !

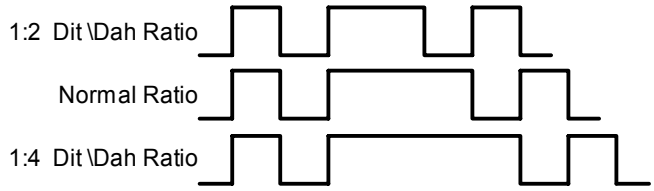


Figure 8 – Three ratio settings for the letter R

Buffered Commands

These commands go into the input buffer maintaining their positional relationship to data.

● PTT On/Off <18><nn>nn = 01 PTT on, n = 00 PTT off

This command allows the PTT output to be used for a custom purpose. The command is operational only when PTT is disabled (*see PINCFG command on page 9*). PTT can be turned on or off at will and is not affected by transmit or any other command including Clear Buffer. Typical applications could be a power level control, antenna selector, or to turn on a cooling fan. Since this is a buffered command, the change in PTT will happen at the command's position in the buffer and remain in effect until the next PTT ON/OFF command is encountered. This command will not stall the output buffer.

● Key Buffered <19><nn> nn = 0 to 99 seconds

Use this command to assert the key output for a specific period of time. Since this is a buffered command, key down will begin at the command's position in the buffer and will stop transmit until the timeout has been satisfied. The key down can be aborted either by the paddles or a Clear Buffer command. The maximum key down time is 99 seconds.

● Wait for nn Seconds <1A><nn> nn = 0 to 99 seconds

This command is used to insert a fixed pause into a message. Since this is a buffered command, the pause will begin at the command's position in the buffer and will halt transmit until the timeout has been satisfied. The wait can be aborted either by the paddles or by a Clear Buffer command. The maximum key down time is 99 seconds.

● Merge Letters <1B>[C][C] Merge Two Letters into a Prosign

You can build "on the fly" prosigns with this command. Issue the merge command followed by two letters or numbers and they will be merged together: <1B> [A] [R] is sent as **`AR.`** Note that nothing will be sent until both letters have been received. Several common prosigns such as AR, SK, BT, and DN are already assigned (see page 16). One application of this feature is to send special European language characters.

● **Buffered Speed Change <1C><nn> nn is in the range of 5-99 WPM**

Example: <02><23> set 35 WPM

This command places a speed change command into the serial buffer that will be acted upon when it is taken out of the buffer. The current speed in force will be stored and will be reinstated when the buffered speed change is cancelled by a Cancel Speed Change command or any of the following: Unbuffered Speed change, Weight change, Farnsworth change, Ratio change, Compensation change, or Mode change.

This command is useful for building messages with embedded speed changes. In this example the first part of the message will be sent at 5 WPM, the second at 25 WPM and the last at the current speed:

```
<1C><05>VVV DE K1EL <1C><19>VVV DE K1EL <1E>END
```

● **HSCW Speed Change <1D><nn> nn = (lpm/100)**

This command acts the same as the immediate HSCW command. This allows you to insert an HSCW burst in a regular CW message or to put HSCW bursts of two different rates into the same message.

● **Cancel Buffered Speed Change <1E> No Parameter**

This command will cancel an active buffered speed change command. The sending speed that was in force before the buffered speed change was encountered will be restored. Several buffered speed changes can be issued within a message but none will alter the original sending speed.

● **Buffered NOP <1F> No Parameter**

This command will occupy a position in the input buffer but will be ignored when it is processed.

PTT Functionality

WKUSB'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 the 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. WK3 provides three independent PTT delays to meet this requirement. (See figures on Page 8)

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 (0 to 250).

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. Tail delay is the sum of two delays, Tail setting times 10 milliseconds plus three dit times.

Tail Delay = Three Dit Times + (Tail Setting *times* 10 milliseconds) *note: one dit time = (1200/WPM) mSec*

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

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.

Unsolicited Status Transmission

WK3 will automatically send two types of unsolicited status to the host: speed pot change, and WK status byte change. Whenever the speed pot is moved, its new value will be sent to the host. Likewise whenever there is a change to Winkeyer's internal status register, a copy of it will be sent to the host. In WK2 mode, the WK status byte is expanded to include pushbutton status.

Refer to the GetStatus command for a description of the status byte format and GetPot command for a description of the speed pot status byte format. Since these bytes can arrive at any time and potentially can be mixed with echo back bytes, they have identifying tags. If the MSB is set that identifies the byte as unsolicited, if bit 6 is clear it's a speed pot byte or if bit 6 is set it's a status byte. In WK2 mode, bit 3 identifies a pushbutton status byte.

The host can request either of these bytes to be returned by using the Get Pot or Get Status commands. Due to the parallel task handling nature of WK3, a response may not happen immediately, there may be another byte in the return queue that needs to be sent first. Worst case latency will be 200 milliseconds. It is not advisable for the host to wait for a response, it is better to handle it as illustrated by the code fragment presented on page three.

Prosign Key Assignments

WK3 has mapped several unused ASCII codes to standard prosigns. Table 14 shows the mappings. Any additional prosigns can easily be generated using the merge character command.

ASCII	Hex		Prosign		ASCII	Hex		Prosign
"	0x22	*Is mapped to*	RR		+	0x2B	*Is mapped to*	AR
#	0x23	*Is mapped to*	EE (null)		-	0x2D	*Is mapped to*	DU
$	0x24	*Is mapped to*	SX		/	0x2F	*Is mapped to*	DN
%	0x25	*Is mapped to*	EE (null)		:	0x3A	*Is mapped to*	KN
&	0x26	*Is mapped to*	EE (null)		;	0x3B	*Is mapped to*	AA
'	0x27	*Is mapped to*	WG		<	0x3C	*Is mapped to*	AR
(	0x28	*Is mapped to*	KN		=	0x3D	*Is mapped to*	BT
)	0x29	*Is mapped to*	KK		>	0x3E	*Is mapped to*	SK
*	0x2A	*Is mapped to*	EE (null)		@	0x40	*Is mapped to*	AC

Table 14 – Prosign/Abbreviations Assignments

Serial Baud Rate

WK3's standard baud rate is 1200 baud with an alternate baud rate of 9600 selected by command. The communication settings are eight data bits, 2 stop bits, with no parity. Some thought has to be given to using 9600 baud. WK3 will always power up at 1200 baud so the host interface must always be prepared to start the initialization at 1200 and then switch over to 9600 after the baud change command is issued. The baud rate will revert back to 1200 baud if WK3 is reset.

Gap (Extra Space) Insertion

WK3 interprets the | character (hex 0x7C) as a ½ dit delay time. The | character can be included in a text string to add extra emphasis to similar sounding sequences. An example is W1OMO, sending it as W1|O|M|O makes it easier to copy.

Reset WK3/Restore Factory Defaults

As mentioned previously, the command pushbutton can restore WK3 operation if a lockup has occurred. Lockup can happen if the USB cable is pulled in the middle of a host transmission to WK3. Even though it's a very rare occurance, a means is provided to recover.

Normally, you press the command button and wait for an **R** before entering a command. If you continue to press the command button after the **R** is sent, in about five seconds WK3 will send 6 dits in sidetone and then reboot back to standalone mode and reload the standalone settings. You can then replug it back into the PC and reopen it from you app or just use it in standalone mode. Note that the standalone message contents are not erased but all other settings will be restored back to the last standalone settings you had preserved using the **P** command.

If you have a case where you want to return WK3 back to factory defaults, a special command sequence is provided. Press the command button, wait for the **R** and then enter di-dah-di-dah. When WK3 responds with an **R,** re-enter di-dah-di-dah and WK3 will clear all settings including messages and return to factory defaults. After the initialization is complete WK3 will send dah-dah-dah-dit. Note that this will also erase stored messages.

Sleep Mode

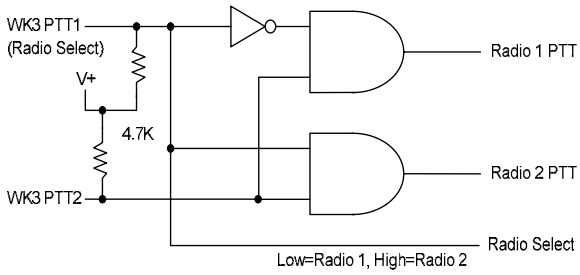
WK3 utilizes the low power sleep mode of the PIC CPU. WK3 normally rests in sleep mode and draws about 1 μA of DC current. When either of the paddles or a push-button is pressed, the chip wakes up and goes into active mode drawing less than 1 ma idle and about 10ma while actively sending and driving sidetone. After the paddle or push-button is serviced WK3 goes back to sleep after a few seconds. Note that running WKUSB without sidetone and instead utilizing your radio's sidetone will extend battery life.

Keyer Lock

A lock feature is provided to disable paddle input and message pushbuttons. This is useful when you want to pack up the keyer, leave the batteries in, and disable it. It's also nice to be able to lock the keyer paddles to keep little hands from sending "messages". While the keyer is locked it is held in low power shutdown mode. To lock the keyer press the command pushbutton, wait for the R, and then enter a period (di-dah-di-dah-di-dah). To unlock the keyer press and hold the command pushbutton for about 5 seconds and WK3 will wake up and send an R.

SO2R Mode

WK3 provides a basic SO2R capablity. When SO2R mode is enabled, PTT operation is modified. PTT1 becomes a radio select output while PTT2 becomes a common PTT output for both ports. To utilize PTT for two radio setups requires additional gating as illustrated below:



The PTT outputs on WKUSB are open collector so pull up resistors are required to generate logic high and low conditions. PTT1 will be logic zero when radio 1 is selected and logic one when radio 2 is selected.

WK3 Host Mode Command Table

Command Name	Code	Type	Description	Syntax	Pg
Admin	00	Imm	Administative Commands	<00><type>…	5
Sidetone Freq	01	Imm	Set sidetone frequency	<01><freq>	6
Speed	02	Imm	Set Morse sending speed	<02><WPM>	6
Weighting	03	Imm	Set key weighting	<03><weight>	6
PTT Lead-in/Tail	04	Imm	Set up PTT delays	<04><leadin><tail>	7
Speed Pot Setup	05	Imm	Set up speed pot range	<05><m><wr><pr>	7
Pause	06	Imm	Pause Morse output	<06><0 or 1>	7
Get Speed Pot	07	Imm	Request speed pot value	<07>	7
Backspace	08	Imm	Backup input pointer	<08>	8
Pin Configuration	09	Imm	Set output pin configuration	<09><config>	8
Clear Buffer	0A	Imm	Clear input buffer	<0A>	9
Key Immediate	0B	Imm	Direct control of key output	<0B><0 or 1>	9
HSCW Speed	0C	Imm	Set HSCW speed	<0C><lpm/100>	9
Farnsworth	0D	Imm	Set Farnsworth speed	<0D><WPM>	9
Winkeyer3 Mode	0E	Imm	Load Winkeyer3 mode byte	<0E><mode>	9
Load Defaults	0F	Imm	Download WK state block	<0F><…15 values…>	10
First Extension	10	Imm	No longer supported	<10><msec>	10
Key Compensation	11	Imm	Set Keying Compensation	<11><comp>	11
Paddle Switchpoint	12	Imm	Setup paddle sensitivity	<12><sens>	11
Null	13	Imm	Null Command, NOP	<13>	11
S/W Paddle Input	14	Imm	Software Paddle Control	<14><paddle select>	11
Winkeyer3 Status	15	Imm	Request Winkeyer3 status	<15>	12
Buffer Pointer	16	Imm	Buffer pointer commands	<16><cmd>…	12
Dit/Dah Ratio	17	Imm	Set ratio of dit/dah	<17><ratio>	13
PTT Control	18	Buff	Turn PTT on/off	<18><0 or 1>	13
Timed Key Down	19	Buff	Turn KeyOut on for an interval	<19><secs>	13
Wait	1A	Buff	Wait for N seconds	<1A><secs>	13
Merge Letters	1B	Buff	Merge chars into prosign	<1B>[c][c]	13
Speed Change	1C	Buff	Change Morse speed	<1C><WPM>	14
HSCW Speed	1D	Buff	Set HSCW speed	<1D><lpm/100>	14
Port Select	1D	Buff	Select output port	<1D><n> n=0 or n=1	14
Cancel Buff Speed	1E	Buff	Cancel Buff Speed Change	<1E>	14
Buffered NOP	1F	Buff	Null Command (buffered)	<1F>	14

Table 15 – WK3 Host Mode Command Table

Winkeyer3 (WK3) Standalone Mode

FEATURES

- Keyer speed range: 5 - 99 WPM
- HSCW: 1000, 1500, 2000, 3000, 4000 or 6000 lpm
- QRSS: 3, 6, 10, 12, 30, 60 second dits
- Non-Volatile Message Memory: 240 letters in 12 slots with embedded commands.
- Dynamically allocated message memory
- Backspace correction on message entry
- Keying Modes: Bug, Ultimatic, Iambic A or B
- Automatic Serial Number Generation
- Audio Frequency keying mode (PTT)
- Adjustable Weight 25 to 75 %
- Automatic letterspace mode (Autospace)
- Unlimited Message Stacking
- Supply voltage monitor
- SO2R Support
- Adjustable Letterspace
- Adjustable Keying Compensation 0 to 31 mSec
- Paddle swap command
- Beacon: Programmable interval: 1 to 99 seconds
- Continuously Adjustable Sidetone Frequency
- Two separate Keying/PTT outputs
- Speed control potentiometer support
- Push-button user interface
- 39 easy to use commands
- Downloadable messages and settings
- Tx and Rx code practice utility
- Two user setups each with callsign

WK3 will operate in standalone mode when not connected to a host and will closely emulate the K14 keyer IC in functionality. The most noticeable difference between host and standalone mode is that when in standalone mode the user can enter commands on the paddles. This is initiated by pressing the command pushbutton. In host mode the pushbuttons are ignored. Low power mode is activated in standalone mode, this means that WK3 will go into a low power sleep state when idle. This makes it very battery friendly.

Standalone Pushbutton Functionality

WK3 standalone requires at least one push-button control, this switch is referred to as the command push-button and is connected to pin 13. It serves two functions, command control and message record/playback control. Up to five additional message push-buttons can be added to provide a total of six message slots. Be sure to use normally open switches for the push-buttons. Pin 13 is an analog input which senses the switch network shown in Fig 22. Message push-buttons 2 through 6 are connected as shown. Use 5% tolerance resistors for the switching network.

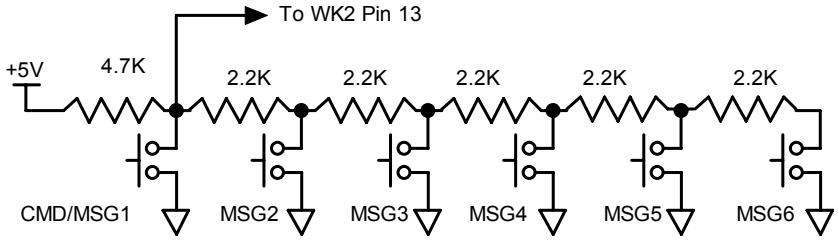


Figure 9 – Pushbutton Matrix

Standalone Command Mode

If the command push-button is pressed and held, the WK3 will respond after about two seconds with the letter ***R*** in sidetone only. This means WK3 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 WK3 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 an ***R***. If an illegal command is entered WK3 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.

WK3 Standalone Command List

A - Select sidetone on or off
B - Start Easy Beacon
C - Set Command Speed in WPM
D - Decrement Serial Number
E - Swap message banks
F - Set Farnsworth Speed
G - Set 50% Duty Cycle Tune
H - Set Paddle Hang Time
I - Set Letterspace Adjust
J - Set Paddle Sensitivity
K - Select Keyer Mode
L - Set PTT Lead-In Time
M - Mute Transmit (CPO mode)
N - Load 4 Digit Serial Number
O - Select output key port
P - Start Practice Mode
Q - Query current settings
R - Review message without transmitting
S - Set preferred WPM speed
T - Load PTT Tail Time
U - Select Autospacing on/off
V - Set Keying compensation in mSec
W - Set Key Weight
X - Select Extended Command Mode
Y - Set Dit/Dah Ratio
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 WK3 responds with. A [pb] means that WK3 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. WK3 will acknowledge this command by responding with an ***R***. Note: If sidetone is disabled it will be re-enabled while in 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: The WK3 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, WK3 responds with an ***R*** otherwise a ***?*** *See the* **S** *command for details on setting transmit Morse speed.*

D - Decrement Serial Number by 1: WK3 responds with a ***D*** after the decrement.

E - Swap message banks: The WK3 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 WK3 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 WK3: 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.

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 Lead/Tail description on page 7 for more information.

M - Toggle transmit mute: Mute transmit when you want to use the WK3 as a Code Practice Oscillator (CPO) When muted, WK3 will send CW in sidetone only. In response to this command, the WK3 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 play message token /N into a message. The serial number is automatically incremented after playing. 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 WK3 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 WK3.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 mode 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 diffculty 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: WK3 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 WK3 will repeat the first character followed by a new character. Now you must echo back both characters. WK3 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 WK3 will respond with an ***X*** and start over with a new 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 key port, enable transmit mute (**M** command) to inhibit this.

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 WK3 Current Settings: WK3 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 extension (this parameter described in host mode section)*
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*
V	*followed by firmware revision, major.minor; 30.01 for 3.0 version 3.1… etc.*

R [pb] - Review a message without transmitting: After **R** is entered, WK3 will respond with an ***E***. Press the message button of the message you wish to play. The message will be sent in sidetone only. If you try to play an empty slot, WK3 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: One value is entered between 5 and 99. This speed is activated when the speed pot is turned fully counterclockwise. The speed pot will act normally above this setting.

The minimum WPM is 5 WPM, the maximum speed is 99 WPM. If the speed is valid, WK3 responds with an ***R***, otherwise a ***?***

T [nn] - Set PTT Tail Time: WK3 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

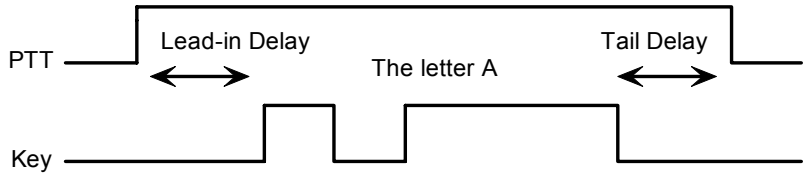


Figure 10 – PTT Lead-in and Tail Example

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 WK3 will automatically insert proper inter-letter space between letters. Each time the **U** command is issued WK3 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 WK3 will interpret this as a letter-space and will not send the next dit or dah until the letter-space time has been met. The normal letter-space is 3 dit spaces but this can be increased by using the **I** command. WK3 has a paddle event memory so that you can enter dits or dahs during the inter-letter space and WK3 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. WK3 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.

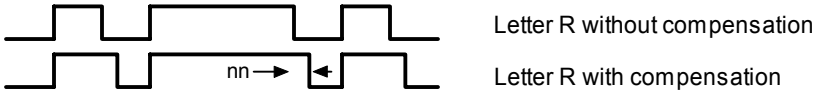


Figure 11 – Key Compensation

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.

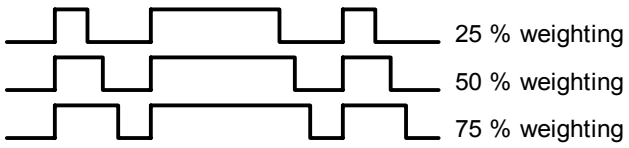


Figure 12 – Key Weighting

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 WK3 will respond with an ***R***
Enter an **X** and WK3 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 continuosly 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 WK3 input pin 7. This design allows four switches and a potentiometer to share a single pin on WK3. The state of the network will produce a unique numeric value which WK3 uses to identify a pushbutton press or speed pot change.

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

Command PB	0 to 11	PB 4	58 to 89
PB 2	12 to 31	Speed Pot	90 to 205 (approx)
PB 3	32 to 57		

WK3 will remain in this diagnostic state until power is cycled.

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 WK3 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.

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- Toggle SO2R Mode: When SO2R is enabled normal PTT is turned off and instead PTT1 is used to signal which output port is active. When port 1 is active, PTT1 is asserted (radio 1 selected) when port 2 is active, PTT1 is not asserted (radio 2 selected) Note that the PTT1 output is open collector and is on when asserted. Command response: an ***A*** is sent when enabled, an ***N*** is sent when disabled.

P - Toggle PTT Mode: The user can choose whether to use automatic PTT while keying. This is a toggle command and after it is entered, WK3 will respond with an ***A*** for PTT enabled, or ***N*** for PTT disabled. Automatic PTT is diabled if you wanted to control PTT manually by embedded command.

R [nn nn] - Set Speed Pot minimum and maximum: After the command is entered WK3 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, if the minimum value is less than 5 WPM. Maximum WPM allowed is 99. If the speeds are valid, WK3 responds with an ***R***, otherwise a ***?***

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

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: WK3 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: WK3 will measure the current supply voltage and send it in Morse sidetone. For example a voltage of 4.52 will be sent as 4r52 with the r indicating the decimal point.

X - Toggle paddle swap (Exchange dit/dah inputs): WK3 will respond with a letter ***A*** to signify when paddle swap is enabled and an ***N*** when turned off.

Standalone Message Functionality

Messages are loaded by pressing the command button until WK3 responds with an ***R*** and then pressing the message pushbutton of the memory slot you wish to enter. When WK3 is ready to accept a new message it will respond with an ***E***. If you wait too long WK3 will respond with a ***?*** and you have to start over. In some keyers, such as WKUSB, only four pushbuttons are provided, messages 5 and 6 are accessed by pressing two pushbuttons in the following sequence after the ***R*** is echoed:

Press and hold either message button 2 or 3. (this specifies message 5 or 6 respectively)
Press the command pushbutton.
Release both push buttons.

(Message 5 and 6 can be played in a similar manner by first pressing and holding pushbutton 2 or 3, then pressing the command pushbutton, and finally releasing both)

After WK3 responds with an ***E***, enter the 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 WK3 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 WK3 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 WK3 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, WK3 will stop further loading, respond with an ***F***, and return WK3 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 each word spaces, as well as embedded commands, occupy memory locations.

When you stop at the end of a message, WK3 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 at the end, especially if you are

using PTT to key an amplifier. This is because WK3 will hold PTT during that added wordspace which will delay an exchange turnaround. The best way around this is to end a message with the di-dah-di-dah (**AA**) character inserted immediately after the last letter in the message. This ends message entry mode immediately.

If you are having problems loading messages into WK3, 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 ***N***, you are not allowing enough space between letters. It's a fine line though because if you allow too much space WK3 will interpret that as a word space. Temporarily lowering the command speed (see command **C**) can help while you learn how the process works.

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 will expand the play ½ letterspace pad character as di-di-dah-dah-dit and expand embedded commands.

Standalone 'Two Press' Message Button Functionality

You can trigger message 5 and 6 directly with the following sequence:
1) Press and hold PB2 for message 5 or PB3 for message 6
2) Press the command PB (you now have two PBs pressed)
3) Now release both pushbuttons and selected message will play.

Standalone Quick Tune Command

This follows a Two Press sequence:
1) Press and hold PB4
2) Press the command PB (you now have two PBs pressed)
3) Now release both pushbuttons and the key and PTT outputs will be enabled.

Pressing the command pushbutton will end tune. You can toggle between 50% and 100% tune duty cycle by hitting the paddles during tune. Tune will be automatically turned off after 99 seconds.

Standalone Quick Serial Number Decrement

Sometimes during contest operation, a serial number has to be reissued. Since the serial number is automatically incremented when it's played, we need a way to decrement the serial number. This can be done 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, WK3 echoes a single dit.

There is a third way to decrement a serial number, that is with a /**D** command embedded in a message. A message can be built that predecrements the serial number before sending it. For example the following two messages can be created:

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

Message 1 is played as the initial exchange and if the serial number needs to be resent, message 2 would be used. The second message pre-decrements the serial number before sending it. Both of these messages leave the serial number incremented 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**

Embedded message command table

/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.
/Cn	Call message in slot n (1-6), return and resume current message.
/D	Decrement serial number.
/E	Toggle message banks
/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	Play user callsign.
/N	Play Serial Number with auto post increment.
/On	Select key output port; 1 for port 1 or 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 a msg PB (2-6) 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)
/Un	Turn PTT on or off. PTT is turned on when n = 1 and off when n=0.
/V	Send the current supply voltage in Morse, useful for beacons.
/Wnn	Wait for nn seconds. (nn=00 to 99)
/Xn	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
AA	End message load immediately	**(DI-DAH-DI-DAH)**	
IG	Insert ½ letterspace pad	**(DI-DI-DAH-DAH-DIT)**	
IM	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 will send BCON DE K1EL BEDFORD NH every 60 seconds
UR RST IS /P QSL will pause to allow the user to enter the RST then resume automatically
/K05 /W10 VVV DE K1EL will key down for 5 secs, wait 10 secs, and then send VVV DE K1EL
CQ CQ CQ DE /M /M /M will send a 3x3 CQ using the user callsign
/H2CQ CQ DE K1EL K1EL K1EL/S15 DE K1EL will send 1st part at 1500 lpm and the 2nd at 15 WPM
CQ CQ CQ DE K1/I10TMT/I00 K will 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 Will play the QTH from message bank 1 and then return for the name.
/Q2EL /1 will 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 sends K1EL BCON, wait 2 secs, send XrXX VOLTS repeat every 10 secs
SOM/E/C3/E/U1/W5/U0/S50K1EL//1/XEOM **Send SOM, swap to msg bank 2, call msg 3, swap back to msg bank 1, turn PTT on for 5 seconds, change speed to 50 WPM and send K1EL/1, cancel 50WPM send EOM.**

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 Command+PB4 pushbutton press. Upon abort normal keying speed is resumed.

PTT Functionality

PTT in standalone mode works the same way as it does in host mode and is described in detail on page 15. 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 WK3_TOOLS 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 with the di-dah-di-dah (**AA**) character to prevent WK3 from automatically appending a wordspace. If there is a wordspace at the end of a message WK3 will hold PTT till the end of the wordspace.

Preserve Settings

WK3 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 **P** command. Press and hold the command PB until the ***R*** and enter an **X** followed by a **P**. This will save current settings in nonvolatile memory so that all settings will be retained on power cycling. Note that messages are saved directly into nonvolatile memory so the **P** 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 WK3 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. Press and hold the command pushbutton. First an ***R*** will be sent followed by a ***W*** about 5 seconds later which lets you know user defaults have been reloaded.

Restore Factory Defaults

If you want to restore WK3 to original "factory" settings hold the pushbutton down as you would for a warm restart and at the same time hold both paddles closed. First an ***R*** will be sent followed by a ***W*** about 5 seconds later. At that time, all messages have been erased and factory settings have been loaded.

Factory Defaults are:

Fixed WPM: 15	Command WPM: 15	Sidetone: 800Hz	Weight: No adjustment
KeyComp: 0	Interchar Spacing: Normal	SampleAdjust: None	KeyMode: Iambic B
Sidetone: On	Autospace: Off	OutputMode: KEY/PTT	Serial Number: 0001
TX Mute: Off	User: 1	Message Bank: 1	Messages: All Erased
Speed Pot Min: 5	Speed Pot Max: 35	First Extension: 0	Farnsworth: Off
Dit/Dah Ratio: 1:3	Sidetone: On	Tune Duty Cycle: 100%	Extra Letterspace: None

Keyer Lock

A lock feature is provided to disable paddle input and message pushbuttons. This is useful when you want to pack up a battery powered keyer and insure that it stays off with batteries still connected. It is also handy to lock the keyer paddles to keep little hands from sending "messages". While the keyer is locked it is held in low power shutdown mode. To lock the keyer, press the command pushbutton, wait for the ***R***, and then enter a period (di-dah-di-dah-di-dah). WK3 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 WK3 will wake up and send an ***R***. This is a very reliable lock mechanism because once the paddles are removed there is no possible way to accidentally unlock WK3.

Shortcuts and Speed Ups

A goal for WK3 design was to streamiline command entry, here are some notable improvements:

Fast Command responses – WK3 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 WK3: 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 WK3 will figure out that there is only one. For example instead of entering **T 0 7** you can simply enter **T 7**.

WK3 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 4 and then press the command button 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. WK3 echoes a single dit to signify that the command was accepted.

Fast Command Response – If you think WK3 takes to 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 – WK3 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**, WK3 will respond with an ***E*** to let you know it is wating for the value.

Standalone Speed Potentiometer Functionality

WK3 uses a 10K potentiometer connected to pin 10 to set sending speed. Turning the speed control will change the speed and update the WPM rate immediately with minimal lag. The entire sweep of the speed pot is called the speed window and it can be modified with two commands. The **B** command sets the bottom, or lowest point of the sweep. The **S** command sets the WPM range. Generally an operator sets the speed bottom first to set the starting point and then enters a range value which determines the upper WPM. Entering 10 WPM for the bottom with a range of 40 WPM will specify a window that starts at 10 WPM and sweeps up to 50 WPM. This allows you to tailor the speed control to an area that you prefer, but it can be changed at any time. The minimum acceptable values for bottom and range are 5 WPM. A 5 WPM range is rather useless, better ranges are in the vicinity of 20 to 40 WPM. If it is desired to set the speed to an exact speed, the easiest way to do this is to turn the speed control fully counter clockwise and set the speed bottom to the desired speed. Otherwise you can determine the current speed control setting with the **Q** (query) command.

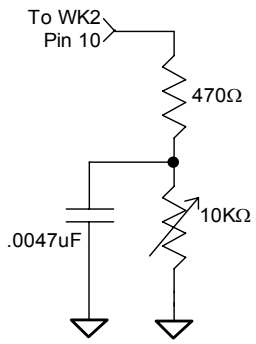


Figure 13

WK3 Standalone Tutorial

On 1st time power up, or by the restore factory defaults command, WK3 will be reset to these settings:

Operating WPM:15	Command WPM:15	Sidetone: 800 Hz	Weight:50%
KeyComp:0	Letterspace Adjust:0	SampleAdjust:50%	KeyMode:Iambic B
Sidetone:On	Autospace:Off	Key/PTT: Port 1	Serial Number:0001
Min Speed Pot: 5 WPM	Max Speed Pot: 35 WPM	SO2R Mode: Off	No 0 or 9 Cuts
User: 1	Message Bank: 1	Tune Duty Cycle: 100%	Farnsworth: Disabled

After WK3 has been powered up, pressing the paddle keys will generate dits and dahs both in sidetone and keyed output. Let's enter a simple command to change the sidetone frequency. Press and hold the command pushbutton (CMD PB) until WK3 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, WK3 will respond with a question mark. After a successful command entry, WK3 will respond by turning on a constant sidetone with keyed output disabled. By pressing the paddles you can raise or lower the sidetone frequency. When you are satisfied with the pitch, end the command by pressing the command PB. With some keyers, such as WKUSB which uses a small speaker, you'll find that higher frequencies produce the loudest sidetone.

Now we will enter an extended command, one that will swap the paddles. Press and hold the command pushbutton (CMD PB) until WK3 answers with an **R**. and then, without hesitation enter an **X** on the paddles. WK3 will answer with an **E** to let you know it is waiting for an extended command entry. In this case, enter another **X** which is the swap command. WK3 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 WK3 will respond with an **N** for **N**ot enabled.

Now try changing the keying mode. Enter the **K** command followed by a letter signifying a keying mode (see page 21). 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. Yopu might want to experiment with a couple of the modes just to get familiar with 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 and slower counterclockwise. When the pot is turned all the way counterclockwise, favorite speed is selected. Favorite speed is set by the **S** comand. To change this, press the CMD PB, wait for the **R**, and then enter **S**. WK3 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 get 10 WPM fully counterclockwise.

The default range for the speed pot is 5 to 35 WPM. Let's change the range using the extended **R** command (see page 21). 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. WK3 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**. WK3 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, WK3 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. If you accidentally set the command rate faster than you can copy, use the warm reboot command on page 28 to get it back to 15 WPM.

The Weight, Keying Compensation, Letterspace, and Dit/Dah Ratio commands adjust the way Morse is generated. Read each command descriptions on pages 21, 23, and 24 to understand how they work.

You can save your settings in WK3's internal EEPROM with the extended command **S** for **S**ave settings.

You can change the keyer's output port 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 separate keying cable from WKUSB to each keyer.

Now try entering some messages. Review the procedure for message loading on page 25. WK3 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 play at two different speeds. Note that after playing 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 WK3 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 messsage 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 played, 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, WK3 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 WK3 to send 0s and 9s in a serial number. Use the extended **T** and **N** commands for this. To play a serial number simply insert a /N command into a message. It's tricky though since the serial number is automatically incremented everytime it is played. 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, WK3 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 play 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 WK3 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 play 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.

WK3 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

WK3 Standalone Commands

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

Extended Command Table

A - Push button diagnostic
C - Contest spacing on/off
F - Fast command response on/off
M - Load user callsign
N - Cut 9 on/off
P - PTT enable on/off
R - Set speed pot range (Min/Max)
S - Save settings in EEPROM
T - Cut 0 on/off
U - Select user 1 or 2
V - Report power supply voltage
X - Swap paddles

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.
/E Swap message banks
/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 Play User Callsign
/N Play Serial Number with auto increment.
/O Keying Port Toggle command
/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 CMD+PB4 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 command (n=0 or 1)
/V Play 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

Introduction 1

Features 1

Theory of Operation 2

Paddle Input Priority 3

Standalone Keyer Mode 3

Power Up Default State 4

Winkeyer3 Lockup Recovery 4

Pushbutton Notification 4

USB Sense 4

Firmware Update 4

Host Mode Command Descriptions 5

Immediate Commands 5

- Admin Command <00><nn> nn is a value from 0 to 20 5
  - 0: Calibrate 5
  - 1: Reset 5
  - 2: Host Open 5
  - 3: Host Close 5
  - 4: Echo Test 5
  - 5: Paddle A2D 5
  - 6: Speed A2D 5
  - 7: Get Values 5
  - 8: Reserved 5
  - 9: Get Cal 5
  - 10: Set WK1 Mode 5
  - 11: Set WK2 Mode 5
  - 12: Dump EEPROM 5
  - 13: Load EEPROM 5
  - 14: Send Standalone Message 5
  - 15: Load X1MODE 5
  - 16: Firmware Update 6
  - 17: Set High Baud 6
  - 18: Set Low Baud 6
  - 19: Set K1EL Antenna Switch 6
  - 20: Set WK3 Mode 6
  - 21: Read Back Vcc 6
  - 22: Load X2MODE 6
- Sidetone Control <01><nn> nn is a value described below 7
- Set WPM Speed <02><nn> nn is in the range of 5-99 WPM 7
- Set Weighting <03><nn> nn is in the range of 10-90% 7
- Set PTT Lead/Tail <04><nn1><nn2> nn1=lead in time, nn2=tail time 8
- Setup Speed Pot <05><nn1><nn2><nn3> nn1 = MIN, nn2 = RANGE, nn3 = 0 8
- Set Pause State <06><nn> nn = 01 pause, value = 00 unpause 8
- Get Speed Pot <07> Return Winkeyer3's current speed pot setting 9
- Backspace <08> Backup the input buffer pointer by one character. 9
- Set PinConfig <09><nn> Set the PINCFG Register 9
- Clear Buffer <0A> Reset Input Buffer Pointers 10
- Key Immediate <0B><nn> nn = 01 key down, n = 00 key up 10

- Set HSCW <0C><nn> nn = the lpm rate divided by 100 ____ 10
- Set Farns WPM <0D><nn> nn is in the range of 10-99 ____ 10
- Set Winkeyer Mode <0E><nn> nn = Mode bit field in binary ____ 10
- Load Defaults <0F><value list> value list is a set of 15 binary values ____ 11
- Set 1st Extension <10><nn> nn is in the range of (0 to 250) × 1 mSecs ____ 11
- Set Key Comp <11><nn> nn is in the range of (0 to 250) × 1 mSecs ____ 11
- Set Paddle Switchpoint <12><nn> nn is in the range of 10-90% ____ 12
- Null Command <13> This command is ignored ____ 12
- Software Paddle <14><nn> nn = 00 paddle up, n=01 dit, n=02 dah, n=03 both ____ 12
- Request Winkeyer Status <15> Return Winkeyer status byte ____ 13
- Pointer Command <16><nn> Input Buffer Command Set ____ 13
- Set Dit/Dah Ratio <17><nn> nn is in the range of 33-66 decimal ____ 14

Buffered Commands ____ 14

- PTT On/Off <18><nn> nn = 01 PTT on, n = 00 PTT off ____ 14
- Key Buffered <19><nn> nn = 0 to 99 seconds ____ 14
- Wait for nn Seconds <1A><nn> nn = 0 to 99 seconds ____ 14
- Merge Letters <1B>[C][C] Merge Two Letters into a Prosign ____ 14
- Buffered Speed Change <1C><nn> nn is in the range of 5-99 WPM ____ 15
- HSCW Speed Change <1D><nn> nn = (lpm/100) ____ 15
- Cancel Buffered Speed Change <1E> No Parameter ____ 15
- Buffered NOP <1F> No Parameter ____ 15

PTT Functionality ____ 15

Unsolicited Status Transmission ____ 16

Prosign Key Assignments ____ 16

Serial Baud Rate ____ 16

Gap (Extra Space) Insertion ____ 17

Reset WK3/Restore Factory Defaults ____ 17

Sleep Mode ____ 17

Keyer Lock ____ 17

WK3 Host Mode Command Table ____ 18

Winkeyer3 (WK3) Standalone Mode ____ 19

Standalone Pushbutton Functionality ____ 19

Standalone Command Mode ____ 19

A - Sidetone enable ____ 20

B [m,dd] - Easy Beacon ____ 20

C [nn] – Command WPM ____ 20

D - Decrement Serial Number by 1 ____ 20

E - Swap message banks ____ 20

F [nn] - Set Farnsworth Speed ____ 20

G - Toggle 50% tune duty cycle 20
H [n] - Set Transmit PTT Hang Delay Time 20
I [nn] - Set Letterspace Adjustment 21
J [nn] - Paddle Sample Point 21
K - Set Keying Mode 21
L [nn] - PTT Lead In Time 21
M - Toggle transmit mute 21
N [nnnn] - Load 4 Digit Serial Number 21
O - Swap Key Output Port 21
P [m,d] - Start Practice Mode 21
Q - Query WK3 Current Settings 22
R [pb] - Review a message without transmitting 22
S [nn] - Set Favorite Speed in WPM 22
T [nn] - Set PTT Tail Time 22
U - Turn Autospace Mode Off and On 23
V [nn] - Keying Compensation 23
W [nn] - Keying Weight 23
X – Enter Extended Command Mode 24
Y [nn] - Set Dit/Dah Ratio 24
Z - Change Sidetone Frequency 24

Extended Command List 24

A – Pushbutton Input Diagnostic: 24
C - Toggle Contest Spacing 24
E[nn] - First Element Extension: 24
F - Toggle Fast Command Response Time: 24
M - Load Callsign 24
N - Select Cut Number for number 9 24
P - Toggle PTT Mode 24
R [nn nn] - Set Speed Pot minimum and maximum 25
S - Preserve Settings in EEPROM: 25
T - Select Cut Number for Zero 25
U - Swap Users: 25
V - Voltage Readout 25
X - Toggle paddle swap (Exchange dit/dah inputs) 25

Standalone Message Functionality 25

Standalone 'Two Press' Message Button Functionality 26

Standalone Quick Tune Command 26

Standalone Quick Serial Number Decrement 26

Embedded Message Command List 26

Embedded message command table 27
Embedded Command Examples 27
Standalone QRSS/HSCW Operation 27
PTT Functionality 28
Preserve Settings 28
Restore User Defaults (Warm Restart) 28
Restore Factory Defaults 28
Keyer Lock 28
Shortcuts and Speed Ups 28
Standalone Speed Potentiometer Functionality 29
WK3 Standalone Tutorial 30
WK3 Standalone Commands 32
Extended Command Table 32
Contact Information 36
Creative Commons Licensing 37
Document Change History: 37
Version 3.0 Software and User Interface Considerations 37
Section One: Standalone Mode Changes (No host mode implications) 37
Section Two: Changes that affect Host Mode 38
Versions Compared 39
Previous Version Reference 40
Description of Version 23 changes 41
Description of Version 22 changes 41
Description of Version 21 changes 42

Contact Information

Winkeyer3 is fully guaranteed and if you are not satisfied please return the chip for a full refund.

Please post questions on the K1EL Message Board:

http://groups.yahoo.com/group/k1el_keyers/

You can contact K1EL directly at:

Steven T. Elliott K1EL
43 Meadowcrest Drive
Bedford, NH 03110 USA

e-mail: K1EL@k1el.com

website: www.winkeyer.com

Creative Commons Licensing

This document and the information within are released under protection of Creative Commons Licensing. The particular licensing chosen allows others to use this information to create an interface for their own personal use but not for profit. This does not apply to software application developers who are writing an interface to WKUSB, there are no restrictions in that regard. This only applies to those who design a WK like device that uses the K1EL Winkeyer command language. Please notify K1EL if you are interested in using the Winkeyer language for your product and we will discuss terms and limitations.

Document Change History:

```
01.16.2006   ste   Initial version
03.31.2006   ste   Minor Updates
12.18.2006   ste   Version 21 changes
04.14.2008   ste   Version 22 changes
10.04.2010   ste   Version 23 changes
08.09.2013   ste   Version 30 changes
```

Version 3.0 Software and User Interface Considerations

Winkeyer Release version 30 (3.0) has a considerable number of changes and improvements which are covered in this section. There are two sections; the first will cover functional changes to standalone keyer mode. The second section will cover changes that affect host operation.

Section One: Standalone Mode Changes (No host mode implications)

1) **Added Extended Commands**: Commands that are not used very often were moved to an extended command set. This allowed several new direct command to be added as well as many new commands in the extended list.

2) **Supply Voltage Monitoring: WK3** has a unique feature that is especially useful in battery powered applications. It is able to accurately monitor its supply voltage and provide an indication when battery voltage is low. Normally the K14 will respond with an ***R*** when entering command mode. If the battery voltage is approaching the minimum operating range, it will respond with an ***L*** instead. This tells the operator that the batteries need to be replaced. The actual supply voltage can be read out in Morse by using the **V** command in the extended command set. See page 11.

3) **Dual User Configuration:** WK3 provides space for two complete configuration setups. For example, one setup could be used for contesting while the other for casual operating. Or, when two operators share the same keyer, each one can tailor the keyer to their taste and store separate profiles. The extended command **U** is provided to select one configuration or the other. All WK3 settings are included in each profile including a stored callsign

4) **Stored Callsigns**: A special memory slot is provided to store the operator's callsign. In fact there are two callsign slots provided, one for each user. The callsign is programmed by using a special callsign load command in the extended command set. The callsign can be embedded in a message with the **/M** buffered command. The callsign slot works like any other message slot, you can call other messages, embed commands, and there is no practical limit to the length of the string.

5) **Dual Message Banks:**
WK3 has two message banks of six slots each for a total of 12 slots not including two callsign slots. While this does not seem like a lot of messages, due to WK3's dynamic memory allocation scheme, it turns out to be more than adequate for most users. It is very easy to swap message banks with the **E** command. This is the shortest command sequence and allows you to swap banks quickly. There is also a buffered message command that swaps message banks.

6) **Sidetone Frequency: I**n the WK2, only a handful of sidetone frequencies could be selected. WK3 allows practically any frequency between 500 Hz and 2000Hz.

7) **Fast Message Interruption:** WK3 waited until a letter was complete before checking for an abort request. The K14 will stop a message immediately upon paddle press, stopping in mid-letter if need be.

8) **Practice Mode:** Both send and receive practice are included. The user can select practice content by letter group so that easy letters can be mastered first followed by progressively more difficult groups. A very good random letter generator is provided which generates varied, ever changing letter order.

9) **Message Stacking:** Up to 10 messages can be queued to be sent in the order requested. The number of nested called messages has been increased as well.

10) **Simplified Beacon Formatting:** In WK3, a **/Jn** command had to be placed at the end of a beacon string in order for the beacon to repeat. WK3 does not require this extra jump command as it is inferred. A WK3 style beacon message looks like this:

/B15 K1EL BCON/J1 can now be formatted as: **/B15 K1EL BCON**

11) **Improved Cut Number Selection: WK3** allows serial number cuts to be used for 0, 9, both, or neither.

12) **Tuning Duty Cycle Selection:** Tune can be set to generate either a 50% or 100% key down duty cycle.

13) **Fixed Speed Setting:** WK3 supports both a variable speed control and a fixed “favorite” speed setting. It is very easy to switch between them. The fixed speed setting is set by the **S** command.

14) **Paddle Serial Number Decrement:** A fast way to decrement the serial number is very useful during contests. The K14 allows this with a simultaneous command pushbutton and paddle press.

15) **Command Response Time Adjustment:** The K14 will enter command mode when the command pushbutton is pressed for about 2 seconds. This may be too long for some operators. A new extended command **F** has been added that allows the delay to be shortened to about 1.3 seconds.

Section Two: Changes that affect Host Mode

This section describes changes that have an effect on host mode operation. All of these commands also apply to standalone mode.

1) **Added Voltage Readback command**: The host can now read back the operating voltage of WKUSB

2) **Added In Field Firmware Update:** Users can now install the latest WK firmware without having to change the WK IC.

3) **Continuously Variable Sidetone**: Set frequency between 500 and 2000 Hz

4) **Changes to X1MODE register, New X2MODE Register**: New bit locations, new functions.

5) WK 3.0 accepts the

6) **New ADMIN commands summary**:

ADMIN CMD	Function	Parameter
0x20	Set WK3 Mode	none
0x21	Return Voltage	none
0x22	Load X2MODE Reg	X2MODE byte value

Versions Compared

Winkey 1 v10

```
0 MODEREG
1 OPRRATE (0 = potlock)
2 STCONST
3 WEIGHT
4 LEAD
5 TAIL
6 MINWPM
7 WPMRANGE
8 1stXTND
9 KCOMP
A FARNS
B SAMPADJ
C RATIO
D PINCFG
E POTRANGE
F N/A

MODEREG
7 NONSTICK: Paddle WDOG
6 MORSEBACK:  Paddle Echo
5 NIAMBIC: Not Iambic
4 KEYERMODE: N0:B/A 1:U/BUG
3 SWAP: Swap paddles
2 ECHO: Serial Echo
1 ASP: Autospace
0 CTSPACE Contest space

PINCFG
7 DITPRI: Ultim Dit priority
6 DAHPRI: Ultim Dah priority
5 HANG1: PTT hang
4 HANG0:   adjustment
3 KEYPORT1: Use Key port 1
2 KEYPORT0: Use Key port 0
1 USETONE: Sidetone on
0 USEPTT: PTT on

STCONST
15 values 0x1 to 0xA

SAMPADJ
Set = 0 to disable latches
```

Winkey 2 v23

```
0 MODEREG
1 OPRRATE (0 = potlock)
2 STCONST
3 WEIGHT
4 LEAD
5 TAIL
6 MINWPM
7 WPMRANGE
8 1stXTND
9 KCOMP
A FARNS
B SAMPADJ
C RATIO
D PINCFG
E X1MODE
F CMDWPM

MODEREG
7 NONSTICK: Paddle WDOG
6 MORSEBACK:  Paddle Echo
5 NIAMBIC: Not Iambic
4 KEYERMODE: NI=0:B/A 1:U/BUG
3 SWAP: Swap paddles
2 ECHO: Serial Echo
1 ASP: Autospace
0 CTSPACE Contest space

PINCFG
7 DITPRI: Ultim Dit priority
6 DAHPRI: Ultim Dah priority
5 HANG1: PTT hang
4 HANG0:   adjustment
3 KEYPORT0: Use Key port 0
2 KEYPORT1: Use Key port 1
1 USETONE: Sidetone on
0 USEPTT: PTT on

X1MODE
7 LSADJ3: Letterspace
6 LSADJ2:
5 LSADJ1:
4 LSADJ0:    Adjustment
3 SN_0_9: 0 and 9 cut enable
2 XMSARE1
1 PDLSTAT: Paddle Status On
0 XMSPARE0

STCONST
15 values 0x1 to 0xA
MSB = PDLST (paddle only ST)

SAMPADJ
Set = 0 to disable latches
```

Winkey 3 v30

```
0 MODEREG
1 FAVEWPM (0 = potlock)
2 STCONST
3 WEIGHT
4 LEAD
5 TAIL
6 MINWPM
7 WPMRANGE
8 X2MODE
9 KCOMP
A FARNS
B SAMPADJ
C RATIO
D PINCFG
E X1MODE
F CMDWPM

MODEREG
7 NONSTICK: Paddle WDOG
6 MORSEBACK:  Paddle Echo
5 NIAMBIC: Not Iambic
4 KEYERMODE: N0:B/A 1:U/BUG
3 SWAP: Swap paddles
2 ECHO: Serial Echo
1 ASP: Autospace
0 CTSPACE Contest space

PINCFG
7 DITPRI: Ultim Dit priority
6 DAHPRI: Ultim Dah priority
5 HANG1: PTT hang
4 HANG0:   adjustment
3 KEYPORT0: Use Key port 0
2 KEYPORT1: Use Key port 1
1 USETONE: Sidetone on
0 USEPTT: PTT on

X1MODE
7 USER: User 0 or 1
6 MSGBANK: Message Bank 0/1
5 TUNE_50: 50/100% Tune Duty
4 LSADJ4: Letterspace
3 LSADJ3:
2 LSADJ2:
1 LSADJ1:
0 LSADJ0:    Adjustment

X2MODE
7 PDLSTAT: Paddle Status On
6 FASTCMD: Fast cmd response
5 CUT_9: Substitute N for 9
4 CUT_0: Substitute T for 0
3 PDLSTONE: Paddle sidetone
2 SO2RMODE: PTT1 = select
1 PDLMUTE: Paddle mute
0 SPARE

STCONST
Value ranges from 30 to 165

SAMPADJ
Set = 0 to disable latches
```

Winkey 1 v10

```
WKSTAT  MSBits = 110ssssss
4 TIMEWAIT: Timer wait
3 WKTUNE: Tune Status
2 KBUSY: Keyer busy
1 BREAKIN: Paddle break in
0 XOFF: Serial XOFF
```

Winkey 2 v23

```
WKSTAT  MSBits = 110ssssss
4 TIMEWAIT: Timer wait
3 WKTUNE: Tune WK1 mode
2 KBUSY: Keyer busy
1 BREAKIN: Pdl break in
0 XOFF: Serial XOFF

if (WK2MODE && !PDLSTAT &&
        WKPUSHB)
4 PB4STAT: PB 4 status
3 WKPUSHB (see above)
2 PB3STAT: PB 3 status
1 PB2STAT: PB 2 status
0 PB1STAT: PB 1 status

if (WK2MODE && PDLSTAT &&
    PB4STAT && WKPUSHB &&
    PB3STAT)
4 PB4STAT (see above)
3 WKPUSHB (see above)
2 PB3STAT (see above)
1 Set to 1 if Dah pressed
0 Set to 1 if Dit pressed
```

Winkey 3 v30

```
WKSTAT  MSBits = 110ssssss
4 TIMEWAIT: Timer wait
3 WKTUNE: Tune WK1 mode
2 KBUSY: Keyer busy
1 BREAKIN: Pdl break in
0 XOFF: Serial XOFF

if (WK2MODE && !PDLSTAT &&
        WKPUSHB)
4 PB4STAT: PB 4 status
3 WKPUSHB (see above)
2 PB3STAT: PB 3 status
1 PB2STAT: PB 2 status
0 PB1STAT: PB 1 status

if (WK2MODE && PDLSTAT &&
    PB4STAT && WKPUSHB &&
    PB3STAT)
4 PB4STAT (see above)
3 WKPUSHB (see above)
2 PB3STAT (see above)
1 Set to 1 if Dah pressed
0 Set to 1 if Dit pressed
```

Previous Version Reference

Following are lists of changes for the previous three versions of the Winkeyer IC. This may be helpful in understnding the progression of changes through the versions of Winkeyer2. Winkeyer1 changes are not included.

Description of Version 23 changes

1) A message will be stopped immediately when either paddle is pressed.
2) Message stacking: A second message can now be queued while a message is being sent.
3) Message is aborted by pressing either the left or right paddle or by pressing command and PB4 in dual-press fashion.
4) Dual-press functionality refined: To play or load message 5 or 6, first press and hold either pushbutton 2 or 3 and then momentarily press the command pushbutton.
5) Quick tune: Dual press PB4 and command pushbutton.
6) Tune is ended by paddle press or when PB4 and CMD are dual-pressed.
7) PTT is now held during space and pad characters.
8) Hang delays have been changed:
   Hang = 0: PTT is held for 1 wordspace plus 1 dit time after letter is finished.
   Hang = 1: PTT is held for 1 wordspace plus 2 dit times after letter is finished
   Hang = 2: PTT is held for 1 wordspace plus 4 dit times after letter is finished
   Hang = 3: PTT is held for 1 wordspace plus 8 dit times after letter is finished
9) Tune and Load message standalone commands have been replaced with Load Lead In time (L) and Load Tail time (T) commands. .
10) Lead in and Farnsworth are now disabled during command mode.
11) Standalone embedded port change command **/X** is now a toggle from 0 > 1 or 1 > 0.
12) QRSS and HSCW embedded command lockup on zero parameter is now fixed.
13) Analog standalone embedded commands removed.
14) Command letters Y and Z are reassigned to relative WPM change commands.
15) Greatly Improved battery life.
16) Letterspace adjustment command added for host and standalone command.
17) Relative WPM change commands added. Y is speed up and Z is speed down..
18) Speed control response improved, now updates in the midst of sending a letter
19) Changed **Q** command value ID letters to match command letters.
20) Changed operation of **B** and **S** standalone commands to adjust speed control window.
21) Set Hang command now accepts a value directly instead of cycling through settings.
22) Removed message break setting, no longer needed.
23) Dual-press CMD+PB4 needed to escape from **/P** embedded command.
24) Quick serial number decrement: Hold command pushbutton and press either paddle.

Description of Version 22 changes

1) In pre-V22 the POT RANGE parameter was not validated, if an illegal value was entered erratic behavior could result. In V22 an illegal value is ignored
2) The Speed Pot value is now averaged to reduce random speed pot changes due to noise.
3) Changes in Speed Pot are ignored during transmit. Again this reduces random speed pot changes due to induced RF. This was never proven to be a problem in V21 but it was easy to implement. The speed pot will be monitored in between active dits and dahs so that a user can still change speed while transmitting.
4) PTT release is now measured one dit time after the last element sent. In addition a tail setting of 1 means no additional delay time is inserted beyond the dit time. In other words delay equals setting minus 1. This greatly improves the QSK performance. Hang time operation has not been changed.
5) Direct (software) paddle input now operates for all modes. In v21 and earlier Iambic A and B were the only modes supported for direct paddle input.
6) V22 now operates properly when it is connected to a host application and the host PC goes into standby. This places all USB ports into a standby sleep mode. This would cause V21 to disconnect from the host and go into standalone operation. V22 will go into sleep mode in sync with the host and will wake up when the PC does. It will remain connected so that the user can continue to use WKUSB from the host application as if nothing happened.

7) Two-press pushbutton sequences are supported for standalone operation. This allows messages 5 and 6 to be accessed directly and also maps the serial number decrement command to a push button press sequence. This is described in detail on page 20 of this manual.
8) A minor bug in autospace has been fixed. In some cases an inserted autospace was a wordspace instead of a letterspace

Description of Version 21 changes

1) The PTT_ENABLE bit in the PINCFG register was ignored in version 20, PTT always cycled with keying. In version 21 the bug was corrected and PTT will assert automatically with keying only when PTT_ENABLE is asserted. If PTT_ENABLE is cleared the buffered PTT command can be used to cycle the PTT outputs.
2) The Buffered PTT serial command (0x18) has been re-instated. When PTT_ENABLE is clear, this command is used to control the two PTT outputs. Either PTT output can be controlled, the selection is based on which KEY bit in the PINCFG register is set. Here are some examples:
To set PTT output 0, first set the KEYPORT0 bit in PINCFG. Then issue a Buffered PTT enable command with a parameter of zero: <0x18><0x01>
To clear PTT output, issue : <0x18><0x00>.
To set and clear PTT output 1, repeat the above sequence with the KEYPORT1 bit set.
Both examples assume that PTT_ENABLE is cleared.
3) A buffered port control command has been added. The command is overloaded on the HSCW speed command (0x1D). To select Key port 0, issue the serial command: <0x1D><0x00> or to select port 1 issue <0x1D><0x01>. By combining the buffered PTT and PORT commands the PTT lines can be synchronously controlled in the midst of keying.
4) Standalone versions of the above two serial commands have been added:
The port command is an **R** followed by either a 0 or 1 to select the desired port
The PTT command is a **U** followed by either a 0 or 1 to either turn PTT off or on.
Message example: **/R0TEST1 /R1TEST2 /U1TEST3 /R0/U1TEST4 /R1/U0 TEST5**

   Here is a breakdown of what the above message does:
   Select key port 0 send TEST1
   Select key port 1 and send TEST2
   Turn on PTT1 and send TEST3
   Select key port 0, turn on PTT0 and send TEST4
   Select key port 1, turn off PTT1 and send TEST5

5) A problem with saving parameters in standalone using the **P** command is fixed.