MorseMonkey

Introduction

Congratulations on your purchase of the MorseMonkey (everyone seems to say congratulations when you buy something… silly isn't it?)

The product is intended to give you a simple tool to practice Morse Code using a straight keyer or an Iambic keyer. The primary ways to practice are:

- Tap the letter or number that is displayed for you and it will display if it was correct (Dictation)
- Tap random letters or numbers and it will display them if it can understand them (Practice)
- Repeat the letter or number you hear and it will display if it was correct (Echo)
- Listen to random call signs and confirm you understood them

It uses digital logic that is powered by 3 volts, so it can run on two AA batteries (not included). Alkaline batteries are recommended.

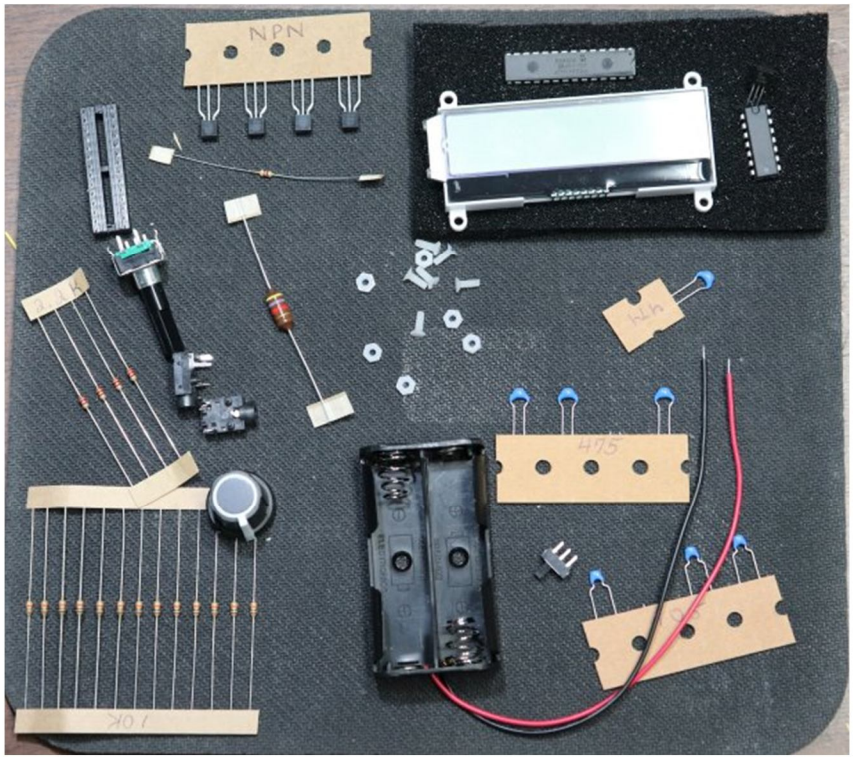
Do NOT attempt to supply it from a 5V power source, as the LCD display and PIC microcontroller are rated for 3.3V typical operation. (The PIC microcontroller has a max rating of 4V, the LCD has a max rating for 3.6V).

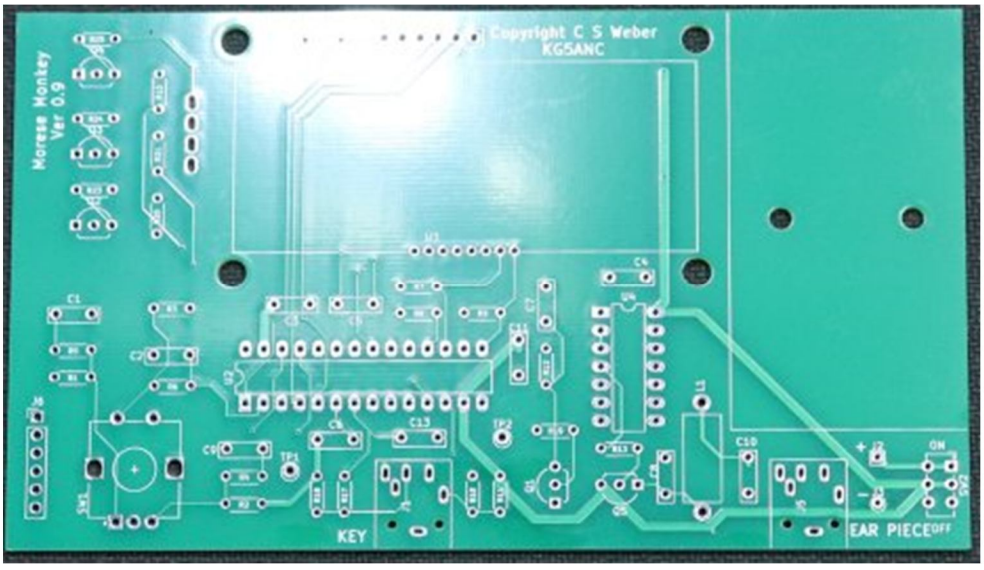
Items which are not supplied, but will be needed for use:

- To listen to the audio generator, you will have to supply a small earbud to plug into the audio jack.

- Your own keyer and a 3.5mm phone jack. The unit can use an Iambic keyer (default) or be set to use a straight keyer. A menu setting controls which type of keying the unit is expecting.

This is a single board you can place on any nonconductive surface when in use. Or you can fabricate your own case for it.





Assembly Instructions

This project has been designed to use as many common parts as possible. Most of the resistors are 10K, with a few exceptions. Most of the capacitors are 1uF, with a few exceptions.

Insure all the parts are present. We hand pack the parts, and try to insure that everything is included. If anything is missing, please contact us.

Several parts have been removed due to a design change. **They are crossed out on the parts list.** Initially a low pass RC circuit was being used to debounce the dial and key contacts, however, debouncing in the software proved to function better. Lesson learned.

Parts List

Ref	Description	Value	Quantity
BT1	Battery Holder		1
~~C1~~, ~~C2~~, C4, C5, ~~C6~~, ~~C9~~, C11, ~~C13~~	Capacitor marked 105	1.0uf	3
C3, C8, C10	Capacitor marked 475	4.7if	3
C7	Capacitor marked 474	0.47uf	1
R1, R2, R3, ~~R4~~, ~~R5~~, ~~R6~~, R7, R8, R9, ~~R10~~, R11, R14, R15, ~~R17~~, R18, R19, R20, R21	Resistor coded brown – black – orange (5 have been removed)	10K	13
R12	Resistor coded yellow – violet -red	4.7K	1
R13, R23, R24, R25	Resistor coded red – red -red	2.2K	4
SW1	Rotary encoder		1
SW2	Power switch		1
J1, J5	3.5mm Audio Jack		2

L1	Inductor marked yellow – violet - red	4.7mH	1
U2	Preprogrammed MCU	PIC 24HJ64GP202	1
U4	Shcmitt trigger inverter	74(A)HC14	1
U3	LCD Display	NHD-C0200	1
Q1	MOSFET transistor	2N7000	1
Q2, Q3, Q4, Q5	NPN Transistor	2N3904	4
--	28 pin DIP socket		1
--	Circuit Board		1
--	Nylon screws		6
--	Nylon nuts		6

Tools Required:

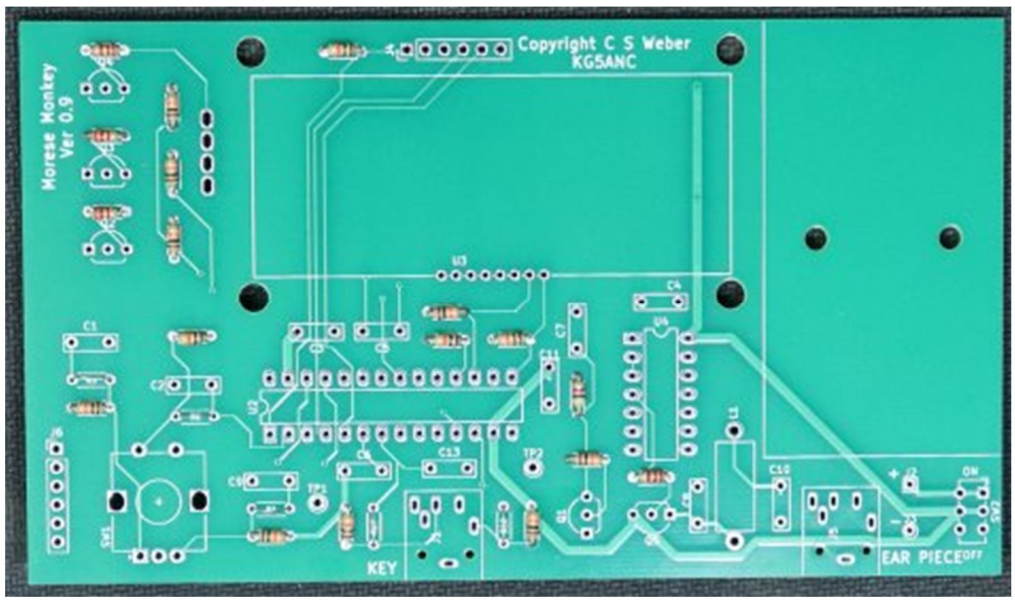
- Soldering iron with small tip
- Solder 0.082 inch or 0.81mm diameter is preferred
- Diagonal cutters to trip leads
- Small screwdriver and needle node pliers to tighten the nylon screws which hold the display and battery
- Digital multimeter to check polarity and troubleshoot if necessary
- Solder wick or other desoldering tool in the event a part needs to be removed
- Small quantity of glue (Elmers© glue all or similar)
- Mouse pad to place the board on to hold it steady while soldering

1. Start by inserting all lowest profile parts first. These would be the resistors. Using the parts list locate all the 10K resistor locations and insert them first. They are R1, R2, R3, R7, R8, R9, R11, R14, R15, R18, R19, R20, and R21.

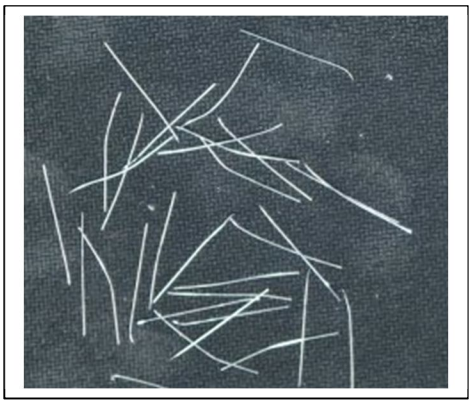
2. Next install the 4.7K resistor R12.

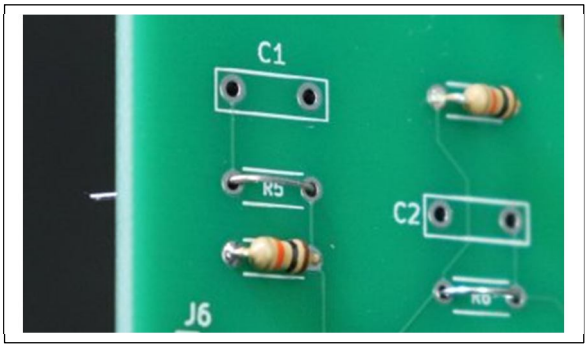
3. Next install the 2.2K resistors R13, R23, R24, and R25.

4. Turn the board over and carefully solder each lead, insuring not to short out to any adjacent traces on the circuit board.



NOTE: There are 5 resistors which are removed. After inserting the existing resistors and trimming the leads, use those leads to in place of resistors R4, R5, R6, R10, and R17.





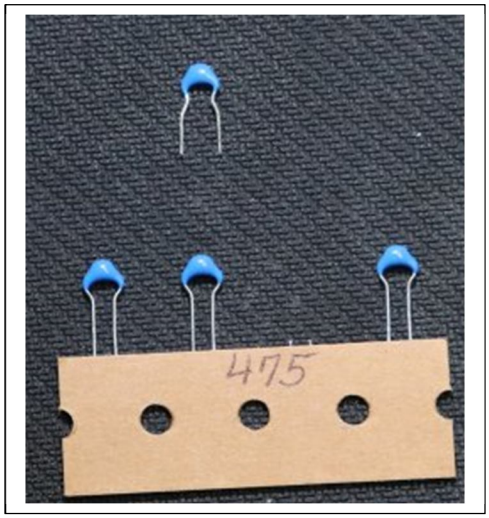
5. The next lowest profile parts is the **socket** for the micro controller U2, and the inverter U4 which is a basic oscillator for producing the audio tone, and U3 the LCD display. Do not insert the

microcontroller yet. The LCD display has 4 holes for the nylon screws. Install the screws through the holes on the display, and the nuts on the underside of the board. After attaching the nut, a small drop of glue can be used to hold the nut in place.

6. To hold the socket and inverters in place, you can slightly bend two of the legs outward from the bottom, so it will not fall out when trying to solder it.

7. Again, turn the board over and solder the components in place. After soldering there is no need to trim the leads.

The next parts to install are the capacitors. Capacitors are marked in numbers like resistors are marked in colors. The code 105 means “one“, “zero”, followed by 5 more zeros, which means 1,000,000 picofarads, which is 1 microfarad. All capacitors are non-polarized



8. Insert the caps in their locations, and solder them in place. First the “105” capacitors which are C4, C5, ad C11.

NOTE: The next capacitors supplied are aligned for a smaller spacing. As shown in the picture, it will be necessary to bend the leads outward slightly. DO NOT bend the leads close to the blue body, and it could cause the capacitor body to fracture. Bend the leads further down from the body of the capacitor.

9. After adjusting the capacitor leads, insert the capacitors marked 475 which are 4.7 microfarards, in C3, C8, and C10.

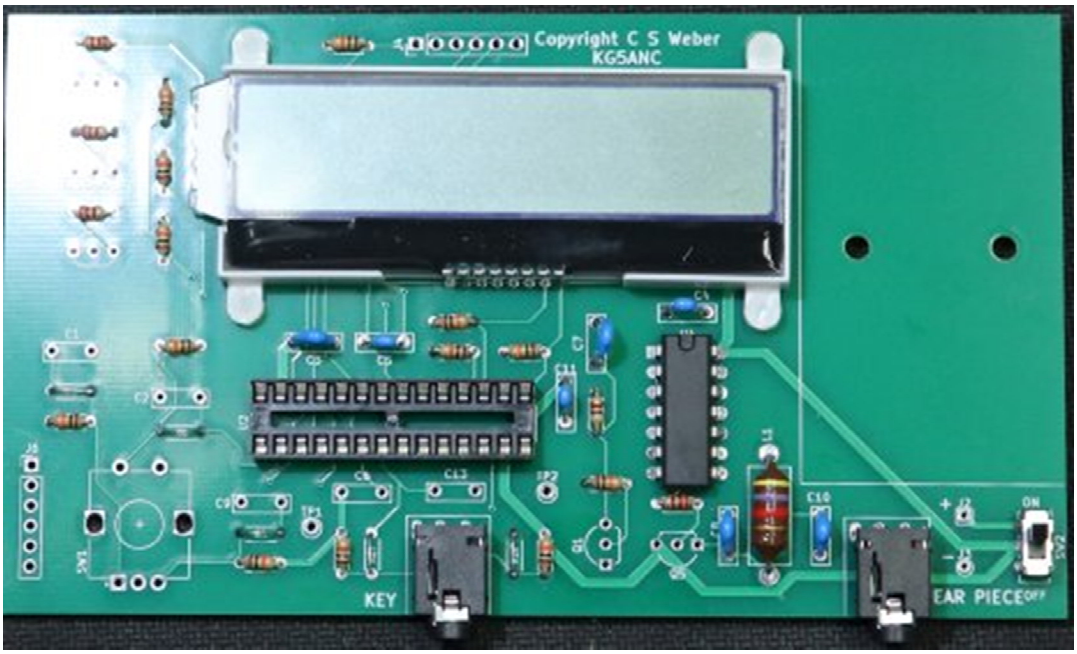
10. Next, install the capacitor marked 474, which is 0.47 microfarads in C7.

11. Again, turn the board over and solder the components in place. Then trim the leads.

12. Next install the audio jacks. J1 is used to attach the keyer. J5 is for the ear piece if used. Solder the jacks in place.

13. Next install the inductor, L1. Solder the inductor in place, and trim the leads.

14. Next, install the power switch SW2, the direction is not important. Solder the switch in place.



15. Next install the transistors Q1 through Q5. Install the transistor with the flat face matching the outline on the circuit board.

Q1 is delivered on the foam to protect it from static. It is installed toward the bottom center.

Q5 is to the right of Q1. Do not get the two transistors confused.

Q2, Q3, and Q4 which are located in the upper left next to the LCD display. Solder the transistors in place. And trim the leads.

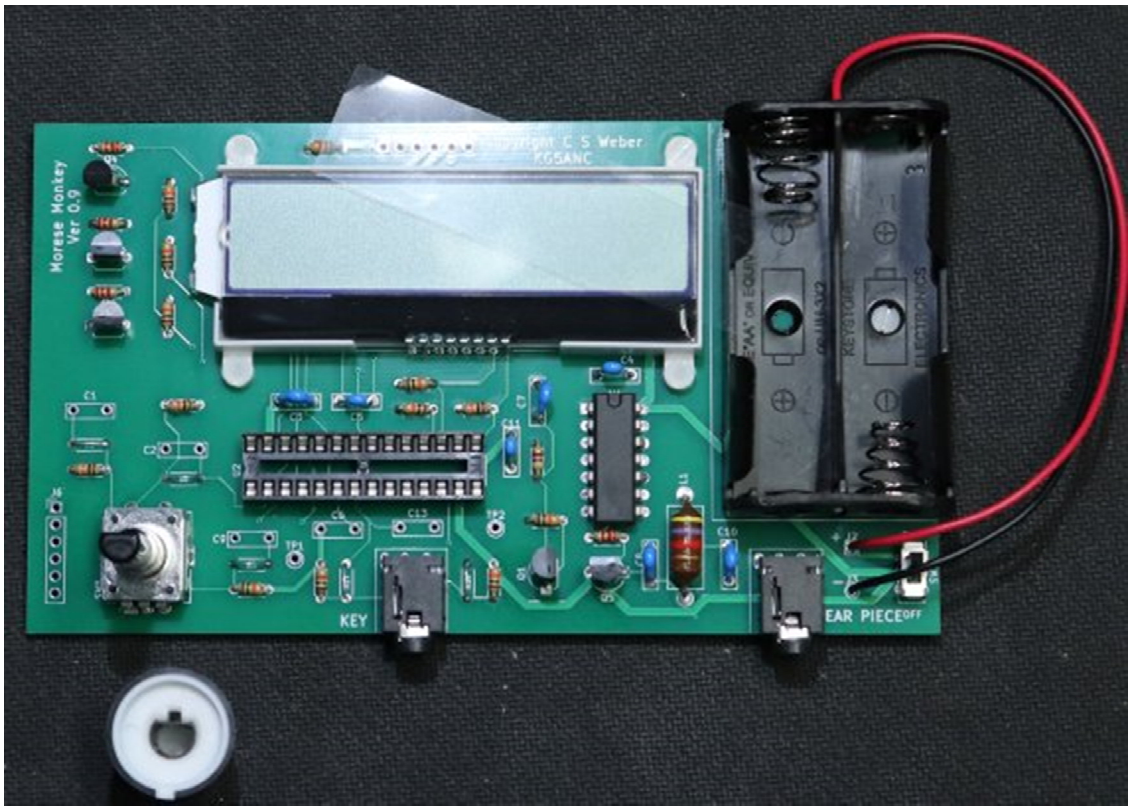
16. The battery holder can be attached next. Use the screws to attach the battery holder to the boards, and insert the red wire (+) into the hole marked J2 below the holder. Insert the black wire (-) in to the hole marked J3.

NOTE: A batch of battery holders have the screw holes different from the circuit board due to an ordering or spec sheet error. The battery holder can be held in place with one screw. It is NOT recommended to

attempt to drill another hole in the circuit board as the board may be damaged. However, drilling small hole through the plastic battery holder is a solution. Or if desired, only use one screw and use a piece of tape to hold it in place, or a small drop of glue can be used.

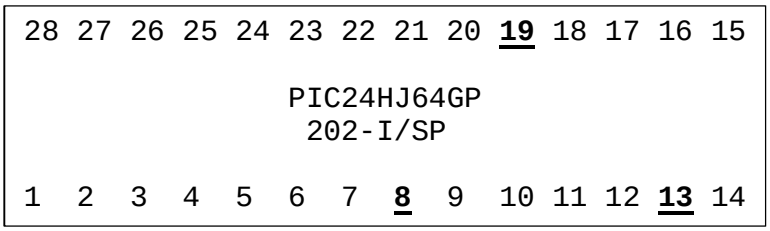
All the nylon screws can be held in place with a small drop of glue on the nut side.

17. Last install the SW1 rotary switch, and solder in place. Then attach the knob by aligning the notch flat side and pressing it on the shaft.



There is a protective film over the LCD display which can be peeled off.

18. Before installing the 28 pin microcontroller, you should use your DVM to test the voltage on the 28 pin socket pins, to insure the voltage is the correct polarity. The pins are numbered below:



Pins 8 and 19 are ground or negative. Pin 13 is the positive supply. Also pin 27 will be ground, and pin 28 will be the positive supply.

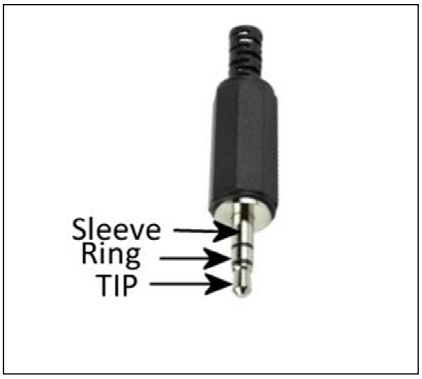
Install the batteries and turn on the power switch. Measure the voltage from pins 19 or 8 to pin 13 to confirm about 3 volts.

NOTE: Due to the 2022 "Chip Shortage" some microcontrollers may be marked PIC24HJ64GP502 rather than PIC24HJ64GP202. These function identical, with the exception that the 502 microcontroller also contains CAN bus peripheral, which is not used.

J4 and J6 are not used at this time, however may be used in the future. J6 is a serial output which can be used for debugging. J4 is for a PICKIT 3 connection to program or reprogram the device.

19. Insuring the notched end on the MCU chip is to the left (placing pin 1 nearest to the rotary dial,) plug the PIC 24HJ64GP202/502 into the 28 pin socket and turn the power on. The

display should show “MorseMonkey” and version number for about ½ a second.



The user will have to supply a keyer with a 3.5mm phone jack, and optional earbud to listen to the tones.

To connect an Iambic keyer, the contacts are connected to the tip and the ring of the phone plug and the common ground is connect to the sleeve. The user can set which of the contacts (tip or ring) will produce the dah or long, and the other contact will produce the dit or short.

To connect a straight keyer, simply connect the keyer to either the tip or ring, and the sleeve as ground. The straight key contact is set the same as the Iambic long contact, as described in the settings section below.

You select which type of keyer to you are using through a menu selection described in the setting section below.

Ensure the speaker or headphone jack is plugged all the way in.

Using the MorseMonkey

The MorseMonkey menu is shown in the display. Rotating the dial will select the menu options, and pressing the knob will select that menu option.

The menu, which is referred to as the "top" menu, will cycle through the following:

Settings – Practice – Dictate – Echo – Copy

Selecting "Settings" will allow the user to select various settings. All settings are reset to the default when the power it turned off and back on.

Selecting "Practice" will display "Press to Start…". Or you can turn the knob to "Back" and press the dial to return to the top menu.

Selecting "Dictation" will display "Press to Start…". Or you can turn the knob to "Back" and press the dial to return to the top menu.

Selecting "Echo" will display "Press to Start". Or you can turn the knob to "Back" and press the dial to return to the top menu.

Selecting "Copy" will display "Press to Start". Or you can turn the knob to "Back" and press the dial to return to the top menu.

Settings

Settings will allow you to select the type of keyer, enable the light, set the WPM and enable playback. To change a setting, turn the dial to the desired setting on the top line, and press the dial until the desired setting appears on the lower line.

The settings options are:

- Keyer can be either Iambic or straight.
- Lights can be either on or off. When using a mode that provides a scored result, the light provides a green or red feedback to the user.
- WPM can be 5, 10, or 18. In some operations, it controls the expected duration of dits and dahs. In other operations, it only controls the time between each character (Farnsworth timing).
- Sidetone can be either on or off. This will playback what you are keying, allowing you to hear the keying.
- Straight/Iambic DAH sets the Keyer Jack J1 contact which will be used for the Straight key or for the long DAH in an Iambic key. This can be either the tip or the ring contact.
- Save will store the settings back into flash memory. To protect writing to the flash memory excessively and wearing it out, it will only save the settings if they are changed.

When the unit is powered up, the settings that were last saved are restored.

Turn the dial to “Back” and press it to return to the top menu.

The operation of the remaining functions is described next.

Practice

Practice will let you tap any characters you wish, and display them. There is no scoring of the characters. In Iambic mode the left and right paddles are used for dit and dah. In straight keying, the amount of time you spend to tap the let depends on the WPM setting. It will also display a series of . and - to show the interpreted dits and dahs. The duration of a dit and a dah is loosely determined by the WPM setting.

When you are done, simply press the dial and the display will return to the top menu.

Dictate

Dictate will display a letter and expect you to tap it on the keyer. In Iambic mode the left and right paddles are used for dit and dah. In straight keying, the amount of time you spend to tap the let depends on the WPM setting. If the unit thinks you keyed the letter correctly, it will display “Correct” and will change the display light to green briefly if the light is enabled in the settings. If the unit thinks you keyed the letter incorrectly, it will display “Incorrect” and the display light will change to red briefly if the light is enabled in the settings.

When you are done, simply press the dial and the display will show a summary of the correct and incorrect characters. Press the dial again to return to the top menu.

Echo

Echo will produce beeps which represent characters. When you hear the character, you must type it back. If you do not start any keying within 10 seconds it will repeat the character. In Iambic mode the left and right

paddles are used for dit and dah. In straight keying, the amount of time you spend to tap the let depends on the WPM setting. If the unit thinks you keyed the letter correctly, it will display “Correct” and will change the display light to green briefly if the light is enabled in the settings. If the unit thinks you keyed the letter incorrectly, it will display “Incorrect” and the display light will change to red briefly if the light is enabled in the settings.

The unit will also display the actual character that was played.

When you are done, simply press the dial and the display will show a summary of the correct and incorrect characters. Press the dial again to return to the top menu.

Copy

This operation does not expect the user to key anything. Copy will generate a random callsign and play it back. The goal is to train the users to listen to the code and correctly copy it down. The callsign is 6 characters:

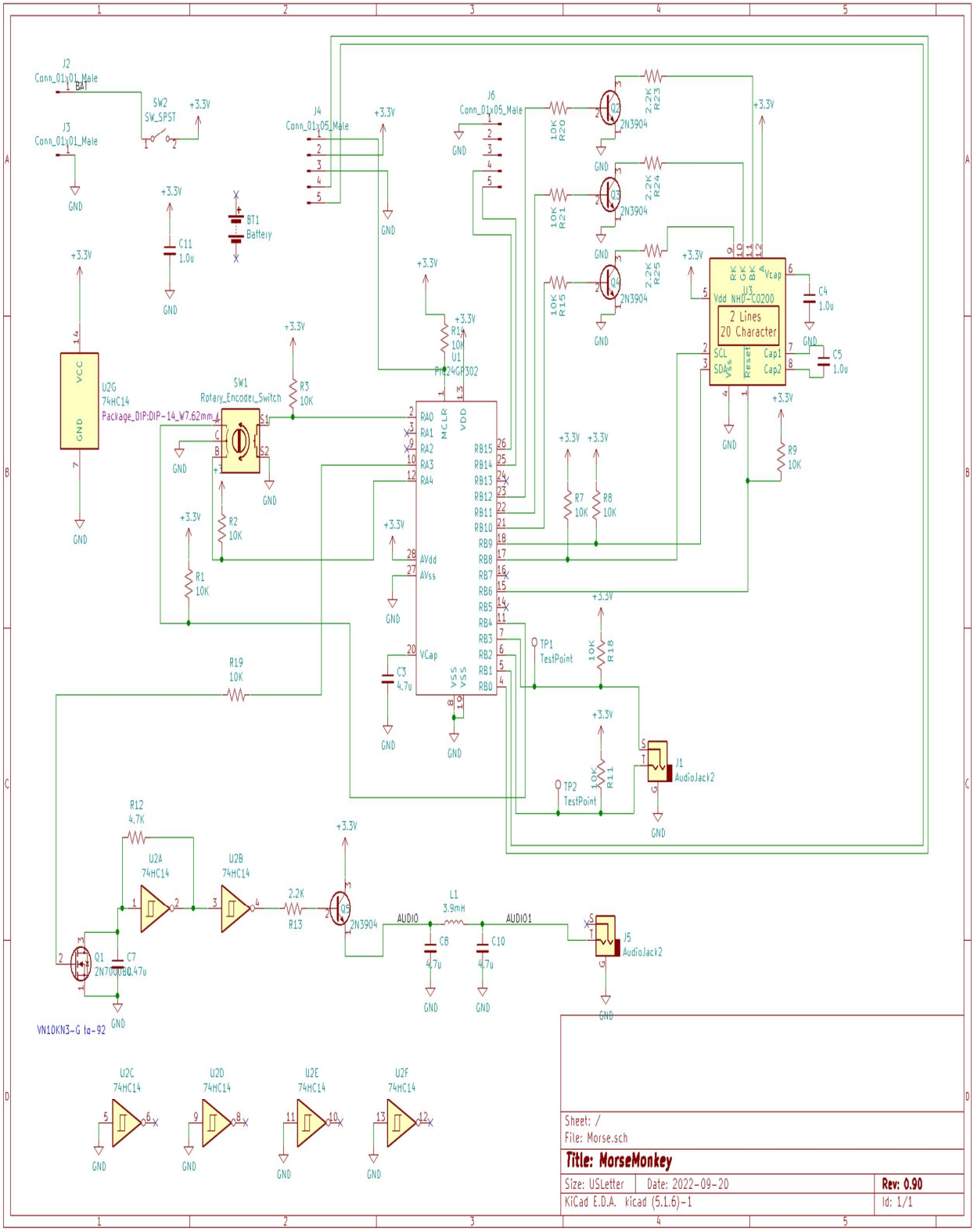
- Begins with either a K or W
- A random letter
- A random number
- and 3 random letters.

The duration of each dit and a dah are not controlled by the WPM, only the pause between each character. This encourages the user to hear the whole “character” rather than a string of longs and shorts. The callsign will repeat every 10 seconds until you stop it.

You do NOT tap the callsign back. Instead, when you are confident of the callsign, press the keyer once (the dah or long when using an Iambic

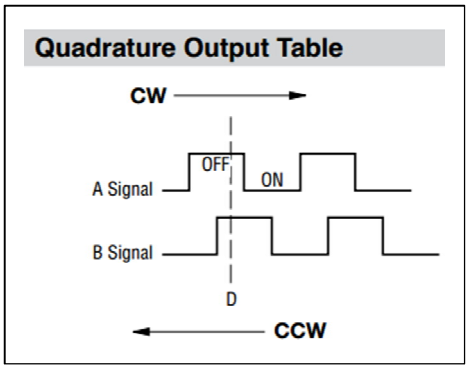
keyer) and the display will show you the generated callsign. Then press the keyer again, and another random callsign will be created and played.

When you are done, simply press the dial and the display will return to the top menu.


Sheet: /
File: Morse.sch
Title: MorseMonkey
Size: USLetter
Date: 2022-09-20
Rev: 0.90
KiCad E.D.A. kicad (5.1.6)-1
Id: 1/1

Circuit Operation

The core of the operation is the PIC24HJ64xxxxx MCU. The MCU runs a 50Mhz instruction clock.



The dial generates a quadrature signal based on the rotation direction. The common contact is tied to ground, and pullup resistors on the two contacts are used to float the inputs to logic high. The MCU is interrupted when signal B on RA4 changes. After a show period (about 2 ms) it test to determine if the signal is low, and whether signal B is high or low. This determine the direction of rotation.

The button press operation is a similar manner on RA0.

The display is driven by a serial peripheral interface (SPI bus) which sends serial commands to the LCD display using RB8 and RB9.

The display contains three LEDs to illuminate the background: Red, Green, Blue with a common anode (+). They are controlled by Q4, Q3, and Q2 respectively. The transistors are driven by MCU outputs RB10, RB11, and RB12. When the outputs are low, the transistor base is virtually ground, and the transistor is cut off. When the output become high, the current is allowed to flow from ground on the emitter to the collector,

and the transistor is in Vce saturation mode, with a typical voltage drop of 0.2V. The LEDs are then powered through current limiting resistors R25, R24, R23 from positive to ground.

The key jack tip input is connected to RB2 and is used to detect the 'dah' in Iambic keying, or the regular key for straight keying. The key jack ring is connected to RB3 and is used to detect the 'dit' in Iambic keying. The ring connection is not monitored in straight keying.

The Schmidt trigger inverted U2A is used to create an approximate 1Khz oscillator. A Schmidt logic device will cleanly change from one state to another when the input voltage reaches a specified point. It will reverse when the input voltage falls below another specified point. This means any voltage between those point will not generate noise, and the level when the change occurs is predictable. The difference is typically 0.9 volts. Starting with an output of high, the input level slowly charges through R12 and C7. When the input reaches a specified voltage, the output will invert. This will cause the capacitor to now discharge through R12 until the input falls to the other specified level. The MOSFET Q1 is used to short out the C7 which turns off the oscillator.

The output from the U2A oscillator is buffered with U2B, and drives transistor Q5 as an emitter follower to deliver more output current. L1, C8 and C10 are used to create a "poor mans" low pass filter, removing the high pitch components from the square wave output from the oscillator. The actual

cutoff of the lowpass filter is dependent on the impedance of the earpiece.

Switch S2 turned the unit on or off.

Personal Note

This is my first attempt at a project for what could be mass consumption. All kits were hand packed and every attempt was made to insure all parts are present. If there are any concerns please contact me.

The ability to update the program is not present at this time. If enough changes and improvements are made, I would consider an update process where the PIC25 is sent in, and re-loaded with updated firmware. Otherwise, for the more daring who may have a PicKit 3 in their possession, I would send out the updated HEX file to be loaded.