Quick and Easy CW

With Your PC

See, hear and send CW using a simple interface and your PC.

By Ralph E. Taggart, WB8DQT
602 S Jefferson
Mason, MI 48854

Most multimode communication processors (MCPs) on the market include CW along with RTTY, PacTOR, AMTOR and other modes. Their CW functions allow you to send perfect Morse code from your computer keyboard. The MCPs do a fair job of decoding received CW, too.

But what if you're not interested in the other modes? What if you want CW *only*? Buying an expensive MCP seems a waste if you only want one mode.

Here's an excellent alternative. With a $50 hardware interface and *free* software, you can transform your computer into a Morse machine that's a full-function CW keyboard *and* a receive display terminal.

My interface works with any IBM-compatible PC and uses the printer port to communicate with the computer (see Figure 1). The interface should set you back less than $50 in parts, and a printed-circuit board is available to make construction easier. An experienced home brewer can have the project finished in a single evening while a rank beginner might need a few evenings. Either way, it's easy to build and simple to align.

Construction

A PC-board pattern and parts overlay for this board are available from the ARRL.[1] You can use them to make your own PC board, or as a guide for wiring the circuit using perf-board. An etched and drilled PC board, with a silk-screened parts layout, is available from FAR Circuits.[2] Assuming you use a PC board or the perf-board equivalent, the required mainframe wiring is shown in Figure 2. The circuit board has all the part outlines and values silk-screened, so it's easy to get everything properly positioned.

The most economical approach to obtaining the needed parts is to order them from one of the larger mail-order suppliers. Most parts are also available at well-stocked Radio Shack outlets or local parts distributors, although the cost will tend to be a bit higher.

Any cabinet that can accommodate the circuit board can be used. The **POWER** switch and **POWER** and **CW LED** indicators are the only front-panel items. J2 (**KEYED LINE**), J1 (**AUDIO IN**), J3 (**+12V DC POWER**), and P1 (**COMPUTER**) are all on the rear apron.

Power Supply Options

The interface requires +12 V, +5 V, and –9 V, all at relatively low current. The simplest solution is to use a wall-mount power transformer/supply (200 mA minimum) to provide the +12 V. A 7805 voltage regulator chip produces +5 V from the +12 V bus for the 74LS TTL ICs. Since the –9 V current requirements are very low, I use a 9 V alkaline transistor battery. The battery is switched in and out using one set of contacts on the **POWER** switch and will last a long time—unless you forget to turn the unit off between operating sessions! Alternatively, you can use a multivoltage power supply module or a dc-dc converter to produce the negative voltage for the op amps (anything from –8 to –12 V will do). The suggested power supply option is simple and inexpensive and also assures that there are no hazardous voltages present in the interface unit.

The Computer Connection

This project is designed to connect to the PC parallel printer port, normally implemented with a DB-25F connector on the rear of the computer. You would expect to see a similar DB-25F connector on the interface rear apron with a standard parallel data cable (DB-25M connectors at each end) to connect the interface and computer.

That's what you'd *expect* to see, but DB-25 connectors need an odd-shaped mounting hole, which is difficult to make with standard shop tools. We can simplify construction since we only need the four conductors normally found in a parallel cable: ground, printer data bits 0 and 1, and the strobe data bit.

Connections are made at the interface end using a common 4-pin microphone chassis mounting connector. This connector mounts in a standard ⅝-inch round hole that doesn't take long to make, even if you have to file it by hand. To interface to the computer, we need a cable with a 4-pin microphone plug at one end and a DB-25M connector at the other. The DB-25M connector is actually fabricated from a standard male connector and a plastic DB-25 shell, both of which are widely available. The microphone plug and chassis connectors can usually be found at the same distributors who sell the DB-25 hardware.

For our cable, any lightweight cable with at least four conductors will do just fine. Jacketed 5-conductor TV rotator cable will work very well, but so will a range of available intercom or speaker cables. The following interconnections between the microphone plug and DB-25M connector are required:

Function	Microphone Plug	DB-25M Connector
Ground	1	25
Printer data 0	2	2
Printer data 1	3	3
Printer strobe	4	1

If you've wired the microphone connector on the interface rear apron properly, there are no voltages or signals that can harm your parallel port. You should take reasonable care in wiring the cable, however. If you make an error, the interface simply won't function when you run the software.

If you have the facilities (or the patience) to use a DB-25 connector on your interface, use a DB-25F connector and wire I/O points 1-4 to the connector, following the pattern indicated for the cable. Connection to the computer would then be made using a standard "parallel cable" with DB-25M connectors at each end.

Keying Options

If your equipment uses a positive, low-voltage keying line, point **K** on the board can be connected directly to the keying jack. In this case, you will not need the reed relay and one of the 1N4004 diodes. Since all my CW work is done using low-voltage QRP gear, I used this option for the operational version of my interface. If you might use a wider range of transmitting equipment, use the keying relay. The circuit board does not accommodate the relay (a wide range of relays can be used), but you can mount the relay anywhere in the cabinet using a dab of silicone adhesive or a piece of double-sided foam mounting tape. If you will be operating at relatively high power, I would also suggest the use of the ferrite beads at all wires connected to the rear-apron connectors.

Alignment

There are only three alignment adjustments, all of which are associated with the receive interface. Most selective CW receivers have a very pronounced peak for the beatnote, somewhere in the 400 to 1000-Hz range. In the case of my ARK-30 transceiver, this peak occurs at 700 Hz. I will use this as an example, but you would use a frequency appropriate to your receiver.

Switch the receiver to a dummy antenna to eliminate any interfering signals and tune the receiver to a strong signal from your frequency calibrator or other stable signal source. Carefully adjust the receiver for peak

audio output. The audio input to the interface can be a data output port from the receiver, or you can interface to the external speaker line using a **Y** connector. Connect a pair of headphones or other audio-monitoring option to the junction of the 0.1-μF capacitor and 10-kΩ resistor at the output of U3. Adjust the **TUNE** (R1) control on the PC board for the loudest signal. The filter is sharp, so make the adjustment carefully.

Set the PC board **LEVEL** pot (R2) to midrange and adjust the **VCO** pot (R3) until the **CW** LED comes on. Decrease the **LEVEL** setting slightly (adjust the control in a counterclockwise direction) and readjust the **VCO**

pot, if required, to cause the **CW** LED to light. Continue to reduce the **LEVEL** setting in small steps, refining the **VCO** setting until you reach the point where operation of the **CW** indicator becomes erratic.

Now advance the **LEVEL** control clockwise to just past the point where the LED comes on with no sign of erratic operation. The **LEVEL** threshold setting is critical for best operation of the receive demodulator. If the control is advanced too far, the LED will trigger on background noise and copy will be difficult. If you reduce the setting too far, the interface will trigger erratically, even with a clean beat note. If you have a reason-

ably good CW receiver (CW bandwidth crystal filters and/or good audio filtering), you can back down the **LEVEL** control until the LED stops flickering on all but the strongest noise pulses, but where it will still key reliably on a properly tuned CW signal.
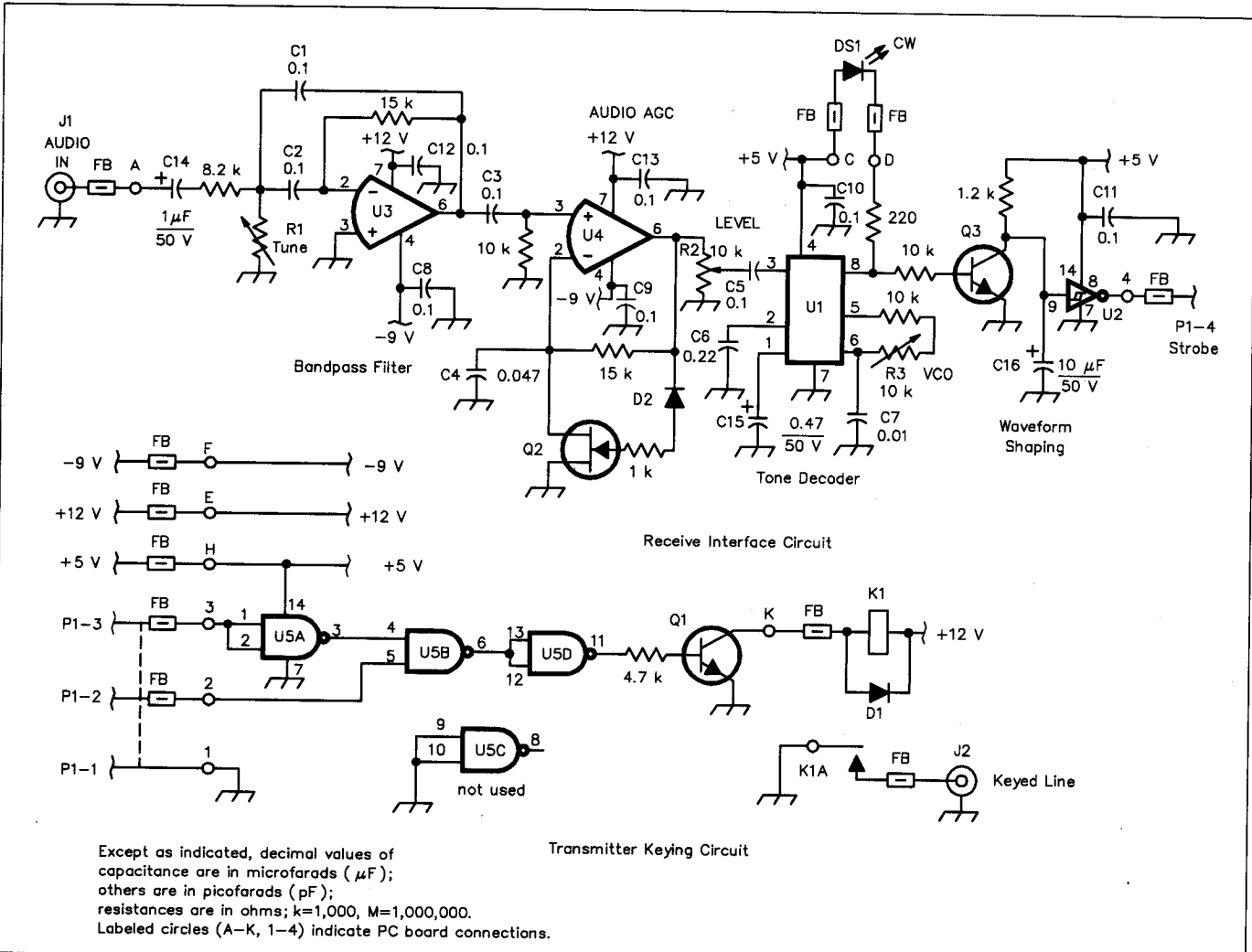
Figure 1—Schematic of the CW interface. All fixed value resistors are ¼ W, 5%-tolerance carbon film. Capacitance values are in microfarads (μF). IC sections not shown are not used.

C1-C3, C5, C7-C13—0.1 μF monolithic
  or disc ceramic, 50 V
C4—0.047 μF Polypropylene (dipped
  Mylar), 50 V
C6—0.22 μF Polypropylene (dipped
  Mylar), 50 V
C14—1 μF Tantalum or electrolytic, 50 V
C15—0.47 μF Tantalum or electrolytic, 50 V
C16—10 μF Tantalum or electrolytic, 50 V
D1,D3—1N4004
D2—1N270 germanium
DS1—Green panel-mount LED (Radio
  Shack 276-069)
DS2—Red panel-mount LED

FB—Ferrite beads (11 total)
K1—12 V dc SPST reed relay (Radio
  Shack 275-233)
J1, J2—RCA phono jacks
P1—4-pin microphone jack (Radio
  Shack 274-002)
Q1, Q3—2N4401
Q2—MPF102
R1—1 kΩ (TUNE)
R2, R3—10 kΩ (LEVEL and VCO)
U1—NE567CN PLL tone decoder (8 pin)
U2—74LS14N hex Schmitt trigger
  (14 pin)
U3, U4—LM741CN op amp (8 pin)

U5—74LS00N quad NAND gate (14 pin)
Miscellaneous
4-pin microphone plug (Radio Shack
  274-001)
DB-25M Connector (Radio Shack
  276-1547B)
DB-25 Shell (Radio Shack 276-1549)
Coaxial power connector (Radio Shack
  274-1563)
8-pin DIP IC sockets
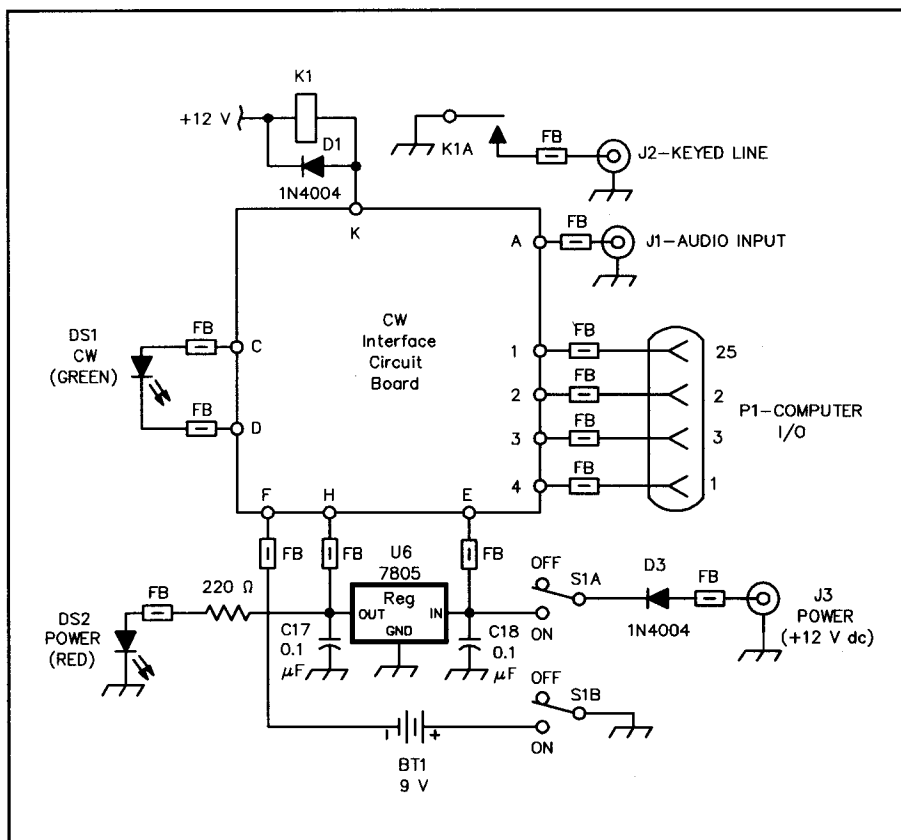14-pin DIP IC sockets
DPDT miniature toggle switch

Figure 2—Mainframe wiring, assuming the use of a printed circuit board or perf-board equivalent. J3 is a panel-mounting coaxial power jack to match your wall-mount/transformer power supply. BT1 is a 9-V alkaline battery. See text. C17 and C18 are 0.1 µF, 50 V monolithic or disc ceramic bypass capacitors. The +5 V regulator chip should be mounted to the grounded wall of the cabinet. Off-board components are duplicated in this diagram (J1, J2, P1, the CW LED indicator, and K1). The CW and POWER indicators are panel-mounting LED indicators. I used red for POWER and green for CW. FB indicates optional ferrite beads used to prevent RF interference with the interface circuits. As noted in the text, if you only intend to use the interface with transmitters with a positive, low-voltage keying line, point K on the board can be connected directly to J2, eliminating K1 and the 1N4004 spike-suppression diode. The break-out points for connections to the circuit board are positioned for clear illustration and do not correspond to the actual locations of these points on the PC board.

Software Installation

The MORSE.EXE program is available at no cost from either the ARRL BBS, 203-666-0578, or my own WSH BBS, 517-676-0368. You'll also find it in CompuServe's *HamNET* forum, library 7.

The distribution file (MORSE.ZIP) includes MORSE.EXE, a sample set-up file (CW.DAT), a sample logging file (LOG.DAT), the HELP text file (CWHELP.DAT), and the program Quick-BASIC source code (MORSE.BAS).
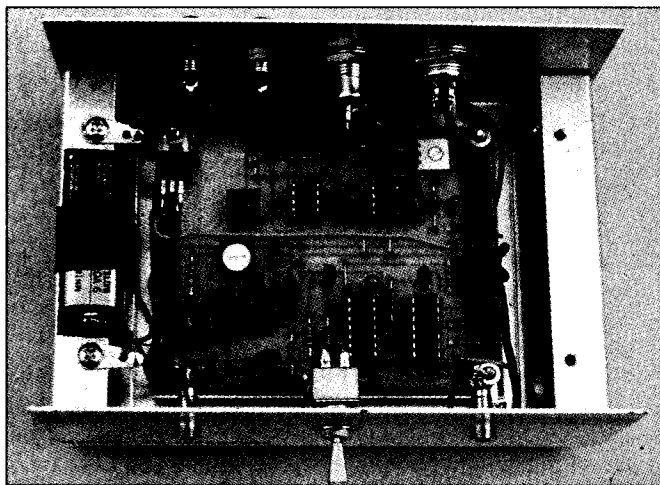
An internal view of the CW interface.

Most of the program structure is built around the many CW keyboard functions, most of which are initiated with single-keystrokes:

❑ *SPEED*—Select your transmitting speed from 5 to 60 wpm. The program autocalibrates to your computer clock speed and transmitting speeds are accurate to within 1%. On receive, the system automatically tracks the speed of the station you are copying from *very* slow up to somewhere between 50 and 60 wpm.

❑ *YOUR CALL*—The program stores your call sign so you never have to type it in routine exchanges. The call can be changed at any time if you want to use the program for contests, special events, or any other situation where you will be using another call.

❑ *OTHER CALL*—If you enter the call of the station you are working (or would like to work), all standard call exchanges at the beginning and end of a transmission can be accomplished with a single keystroke.

❑ *CQ OPTIONS*—The program supports two CQ formats. The "standard" format is a 3×3 call using your call sign. The program also lets you store a custom CQ format which is useful for contests and other nonstandard activities.

❑ *MESSAGE BUFFERS*—The software supports two message buffers that can be implemented with a single keystroke.

❑ *SIDETONE*—If your transmitter doesn't have its own sidetone circuit, you can toggle a sidetone function on or off, letting the computer generate the tone for you. The frequency of the tone can be set between 400 and 1200 Hz.

❑ *WEIGHTING*—Weighting can be adjusted from 0.50 through 1.50.

❑ *DEFAULT SETUP*—All the information discussed up to this point can be saved into a default disk file (CW.DAT) so that you preset all functions whenever you boot the program. The setup can be saved at anytime with a single keystroke.

❑ *LOGGING*—The program supports a range of logging functions that are too numerous to outline here, but one feature is worthy of note. When you first key in the call of a station, the computer will check the log and let you know if you have worked that station before. If you have fully implemented the logging options, it will even tell you the operator's name and QTH.

❑ *HELP FILES*—The program is simple and completely menu-driven. If you forget how to use a function or are using the program for the first time, you can call up on-screen HELP files that explain every function.

Notes

[1] A PC-board template package is available free from the ARRL. Address your request for the TAGGART CW INTERFACE TEMPLATE to: Technical Department Secretary, ARRL, 225 Main St, Newington, CT 06111. Please enclose a business-size SASE.

[2] A circuit board is available for $5 (plus $1.50) from FAR Circuits, 18N640 Field Court, Dundee, IL 60118.