The W9TO vacuum-tube keyer (never published in amateur literature but manufactured by Hallicrafters) was one of the most popular keyers ever designed. Its successor, Chet Opal's Micro-TO keyer, was an all-solid-state version with ICs which appeared in QST *for August, 1967, and gained equal popularity. Now Chet goes one step further, to an all-CMOS keyer with message memory. With its fine ancestral line, this latest entrant in the "most-memory-for-the-least-parts" competition should prove to be a winner.*

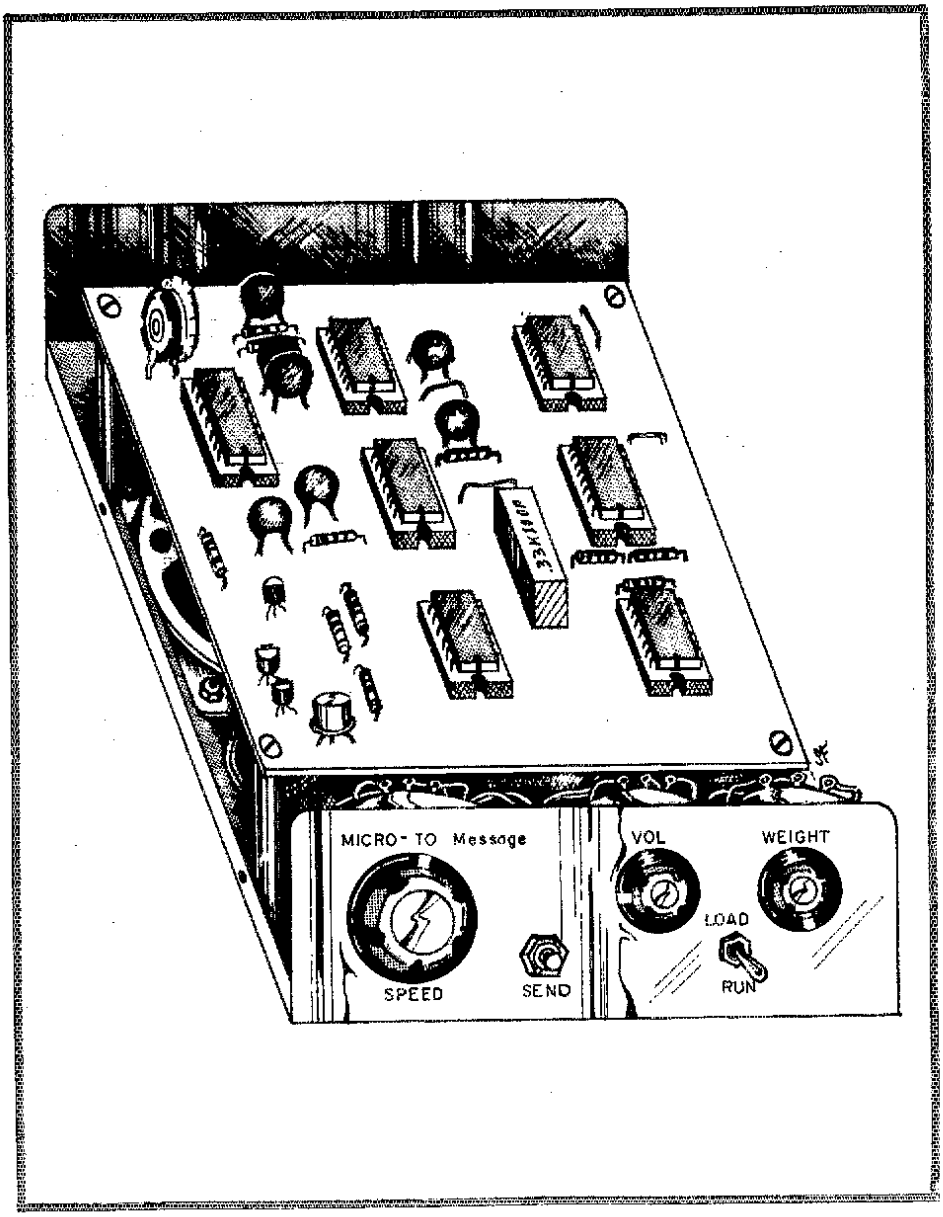
The Micro-TO Message Keyer

By Chet B. Opal,* K3CU (ex-K3CUW)

Since the advent of semiconductor read/write or random-access memories (RAMs), it has been possible to design cw keyers with convenient message-sending capabilities (for example, the Accu-Memory keyer[1]). To operate such a keyer all you do is throw some switches and send the message you want to record. Afterward, the message or messages can be called at the push of a button.

The circuit described here has two advantages over previous designs: It is very simple (only seven ICs, fewer than many nonmessage keyers), and it uses all-CMOS circuitry. Because the CMOS circuits draw so little power, battery operation is practical, which means that the recorded message can be stored indefinitely, independent of the power line (great for Field Day!).

The keyer can be hooked up in parallel with your favorite keyer to provide message capability, or some of the ideas can be applied to existing designs. However, the unit described here has been designed so that it can function as a stand-alone unit. It includes a monitor, a simple weight control, and both positive- and negative-keying outputs. Message loading is very convenient. On playback it can be interrupted by closing either paddle contact. The memory can hold two runs of the alphabet, two sets of numbers, and a punctuation mark or two. The keyer is readily expandable to include multiple-message capability.



Both front-panel and interior simplicity of the Micro-TO Message Keyer are evident in this illustration. A complete keyer with monitor, weight control, and 1024 bits of memory!

The Circuit

The basic idea is quite simple and similar to some previous designs: The message is recorded as a series of 1s and 0s in consecutive locations in a memory. (A dot is a 1 followed by a 0; a dash is three 1s followed by a 0.) On readout, the message is fed back in parallel with the normal keyer output. The design can be adapted to almost any keyer; one requirement is that a free-running clock must be used so that spaces will be recorded and so that the message can be clocked out on playback. In some keyers a free-running clock will generate endless dots which must be inhibited.

Fig. 1 is the schematic of the keyer described here. Circuits U1, U2 and U3 form the basic keyer. U4, U5 and U6 provide memory functions, while U7 and the transistors are associated with the monitor and keying circuits. The keyer portion is essentially a Micro-TO MK II circuit[2] with a few changes. The clock has been modified to include an external "run" input, and a J-K flip-flop has been substituted for the original D flip-flop so

*5414 Old Branch Ave., Camp Springs, MD 20031
[1]Footnotes appear on page 14.

that dots can be inhibited. The message (actually, its complement) goes to the memory from U2C and is fed back and merged in U1C. Basically, by grounding pin 1 of U2A and connecting pin 8 of U1C to the positive supply, one would have an ordinary TO keyer.

The heart of the memory circuit is U6, a 1024-bit CMOS RAM chip. The memory location is addressed by lines A0-A9. The address is latched onto the chip during the falling portion of $\overline{STR}$ ("not strobe"). On the rising edge of $\overline{STR}$, the data at D_I will be stored at that location if $\overline{WE}$ ("not write enable") is low or, if $\overline{WE}$ is high, read out from that location and placed on the D_O output line. Since D_O is often in an undefined open state, a pull-up resistor (R9) is used to define it to "1," i.e., a blank. (Remember we store the complement of the message.) Also, the $\overline{MSG}$ latch flip-flop is used as a buffer between D_O and the output gate, U1C, in order to eliminate glitches while the memory output is changing.

Addresses to the memory chip are generated by U5, a 12-bit binary counter. This chip counts upward by one on each negative transition at the C input whenever it is not held in the reset mode. The clock pulse is derived out of phase with that applied to U6 and delayed by R6 and C4 so that the addresses are set up at the correct times to be accepted by U6. The counter is normally held in the reset mode, with all outputs 0 by the Q output of the $\overline{MSG}$ flip-flop U4B, which is set in the standby mode (this flip-flop is sort of upside down).

The counting circuitry and memory are activated when the $\overline{MSG}$ flip-flop is reset. That can happen in two ways. In the recording mode (S1 thrown to LOAD), a "0" is applied to the D input of U4B. When either paddle is depressed, the output of U1D changes to a 1; this signal is applied to the clock input of U4B, causing the 0 at the D input to be transferred to its Q output. This starts the clock and also takes the binary counter out of the reset mode, so that it starts counting upward. It also enables U6, which records the signal at D_I at successive memory locations following each clock pulse. When the counter gets to the 1024th clock pulse, output 2^{10} changes state; this feeds the S input of U4B, setting it again and terminating the process.

In the playback mode, U4B can be reset by tapping the SEND button. In this case, since $\overline{WE}$ is high, the input is not recorded, but instead played back via D_O. The message normally ends as above, when 2^{10} changes state. In addition, the D input of U4B now has a 1 on it, so that if either paddle is tapped this 1 is transferred to the Q input of U4B, resetting everything. In this way it is possible to terminate a CQ when, as often happens, you hear someone come back to a previous CQ just as you have started the current one.

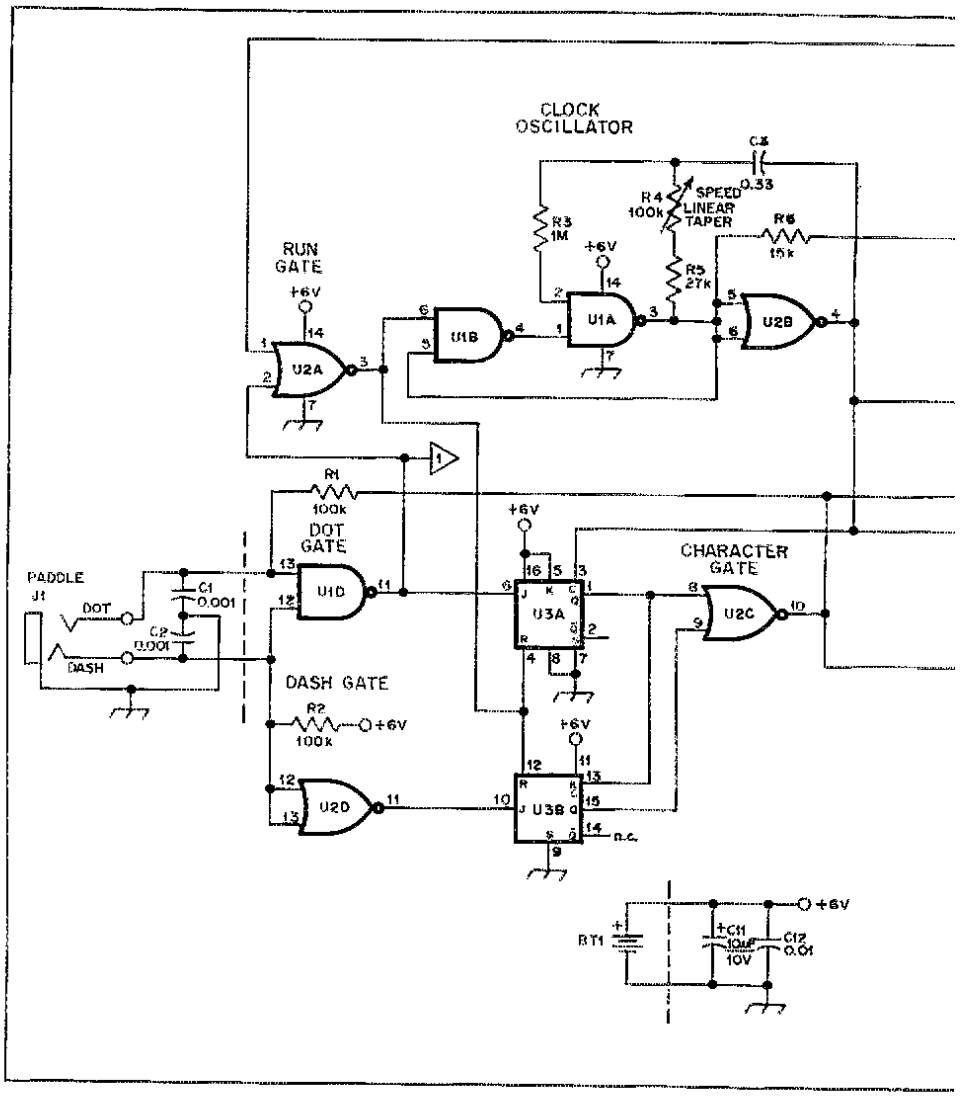


Fig. 1 — Schematic diagram of the Micro-TO Message Keyer. Unless noted, resistors are 1/4 W, 10 percent; capacitors are ceramic, 50 V. C1, C2 and C5 are mounted where their associated signal leads enter the cabinet.

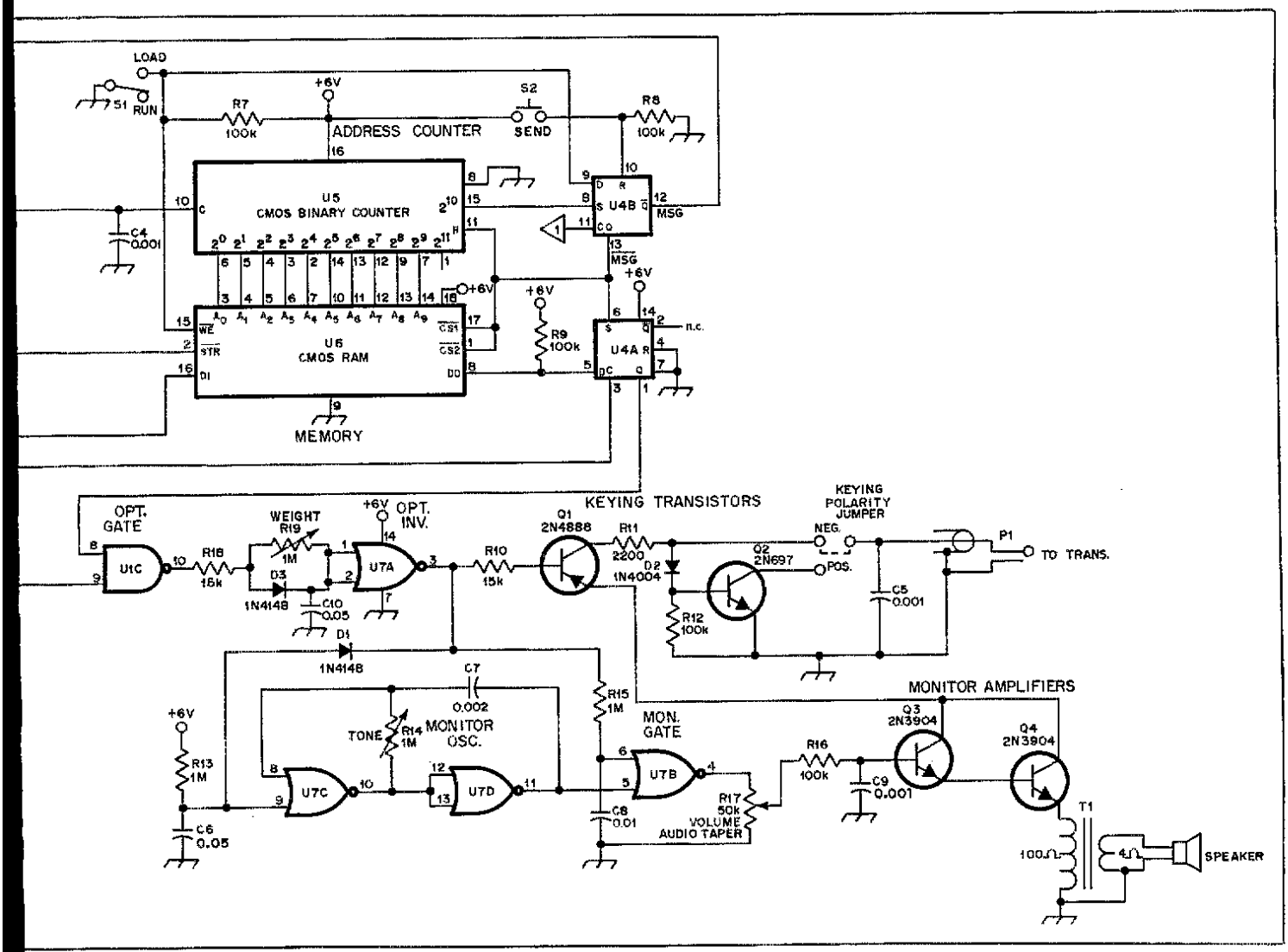
BT1 — 4 to 7-volt battery, four AA cells of any type in series.
C3 — Polyester (Mylar) dielectric capacitor.
C11 — Low-leakage electrolytic, 10 V or higher (tantalum recommended).
D1, D3 — Silicon, signal diode (1N914I, 1N4148, etc.).
D2 — 200-PIV, silicon diode (1N4003, 1N4004, etc.).
J1 — Two-circuit, 1/4-in., phone jack.

The output of U1C contains the message coming from the paddle or the memory. At this point the dot "mark" duration is exactly equal to the "space." In order to compensate for transmitter delays (or to add a little "meat" to the dots, which some operators like) a simple network consisting of R18, R19, D3 and C10, has been installed at the output of U1C. At the beginning of each character C10 charges rapidly through R18 and D3 but at the end it discharges slowly, depending on the setting of weight control R19, extending the "mark" period. The time added is independent of keying speed, so if the weight control is used mainly to make the dots heavier, it will have to be changed whenever the speed is changed.

The monitor is the same circuit as used in the Micro-TO MK II. An additional monitor amplifier is included to provide sufficient drive to the speaker at the lower voltage used here. Also, an npn transistor has been added to provide positive-to-ground keying, which is needed by some of the newer solid-state rigs. The transistor shown will sink about 50 mA of keying current.

Construction

I built my unit using a pc board, the layout for which is shown in Fig. 2.[3] If you use this pattern, be sure to install all the jumpers (18 of them). IC sockets were used everywhere for checkout purposes; this isn't really necessary although it might be worthwhile for U6. I used a Radio Shack no. 270-253 cabinet (5-1/4 × 3 × 6 in. or 133 × 76 × 152 mm), which is attractive and quite inexpensive.



LS1 — 2-1/2-in.(62.5 mm), permanent-magnet speaker,4 to 10 ohms.
P1 — 1/4-in. phone (or other suitable) plug.
Q1 — 200-volt, silicon, pnp transistor (2N5414, 2N4888, MM4002).
Q2 — Medium-current, medium-voltage, silicon switching transistor (2N697, etc.).
Q3, Q4 — Npn, general-purpose, silicon transistor (2N2222, 2N3904, etc.).
R4 — 100-kΩ, linear-taper, 2-W, carbon pot.
R14 — 1-MΩ Trimpot.
R17 — 50-kΩ, audio-taper, 2-W, carbon pot.
R19 — 1-MΩ, linear-taper, 2-W, carbon pot.
U1 — Quad, CMOS, two-input, NAND gate (CD4011AE, SCL4011AD, etc.).
U2, U7 — Quad, CMOS, two-input, NOR gate (CD4001AE, etc.).
U3 — Dual-CMOS, J-K flip-flop (CD4027AE, etc.).
U4 — Dual-CMOS, D flip-flop (CD4013AE, etc.).
U5 — CMOS, 12-bit binary counter (CD4040AE, etc.).
U6 — 1024 X 1-bit CMOS RAM (Intersil IM6518CJN).

Everything will fit into the next smaller size, but I wanted room for future expansion. The speaker is mounted on the floor of the cabinet over an array of holes and the circuit board is mounted on standoffs above it. The battery holder could also be mounted under the board, but I glued it to the cover for easy access.

Most of the parts can be obtained from stores like Radio Shack or Lafayette, which are proliferating everywhere as a consequence of the CB boom. I plan to make available the memory chip, the high-voltage pnp transistor and perhaps other parts, depending on demand. The choice of the push button is important: It must have a soft touch so that the cabinet doesn't walk across the table every time it is pushed. I used an inexpensive, miniature, unenclosed unit and bent the contacts for a feather touch.

Checkout and Operation

A warning: Never install an IC with power connected. Applying voltage to a signal pin before the chip is powered can cause an internal breakdown which persists after power is applied. The CMOS RAM in particular, with thousands of transistors in it, can get "toasty." This doesn't lead to permanent damage if caught in time, but it can be rough on batteries.

Set the WEIGHT control to minimum resistance (fully counterclockwise), the other controls to midrange, and the LOAD/RUN switch to RUN. Hook up the batteries and keying paddle, and close the thumb contact on the paddle. This should lead to a series of dots; the index finger should produce dashes. Adjust the PITCH, VOLUME and SPEED controls to suit. Practice sending for awhile. Now throw the switch to LOAD and send a CQ. You may notice that it is a little harder to send in this mode, since the clock is running continuously, and your timing must be more accurate. After waiting to load blanks into the remainder of the memory, throw the switch to RUN and hit the SEND push button. Out should come an exact replica of what you sent. Determine whether you have a positive or negative voltage at the transmitter key terminals, and patch the keyer output accordingly. Put the top on, plug the key in, and have at it.

Power Considerations

The keyer draws virtually no current in

the standby mode. The actual drain is highly variable, determined primarily by the temperature, the characteristics of the particular memory chip, and potentially leaky components such as the transistors and C11. Actually, C11 is used only to hold power to the keyer while changing batteries. A good test of leakage is to disconnect the batteries, wait a few minutes, then reconnect them. If the message remains intact, there are no serious leakage problems.

The main power drain during operation is from the keying and monitor circuits. Turning the volume control all the way down shuts the monitor off. Another power-saving possibility is to increase the value of R11 until the transmitter barely keys, then substitute about half this value.

I would be pleased to correspond with anyone having difficulties and would also like to hear ideas on what features would be desirable in future versions of the keyer (a stamped reply envelope would be much appreciated). On the latter point, the emphasis here has been on simplicity. Obviously desirable additions would be multiple messages, auto-repeat, iambic operation with dot and dash memories, Morse-to-Baudot conversion, automatic logging . . . . QST

Footnotes

[1]Garrett and Contini, "The Accu-Memory," *QST*, August, 1975.
[2]Opal, "The Micro-TO MK II Keyer," *QST*, September, 1975.
[3]As a convenience to those wishing to avail themselves, ready-made circuit boards may be obtained from the author, as may some of the harder-to-find components. The boards are the same size as the one pictured but have been altered to provide an optional dual message capability. Since prices will depend on demand, the author requests you send a self-addressed stamped envelope for the latest list of prices. (The memory ICs are $7.50 each at this printing date.)

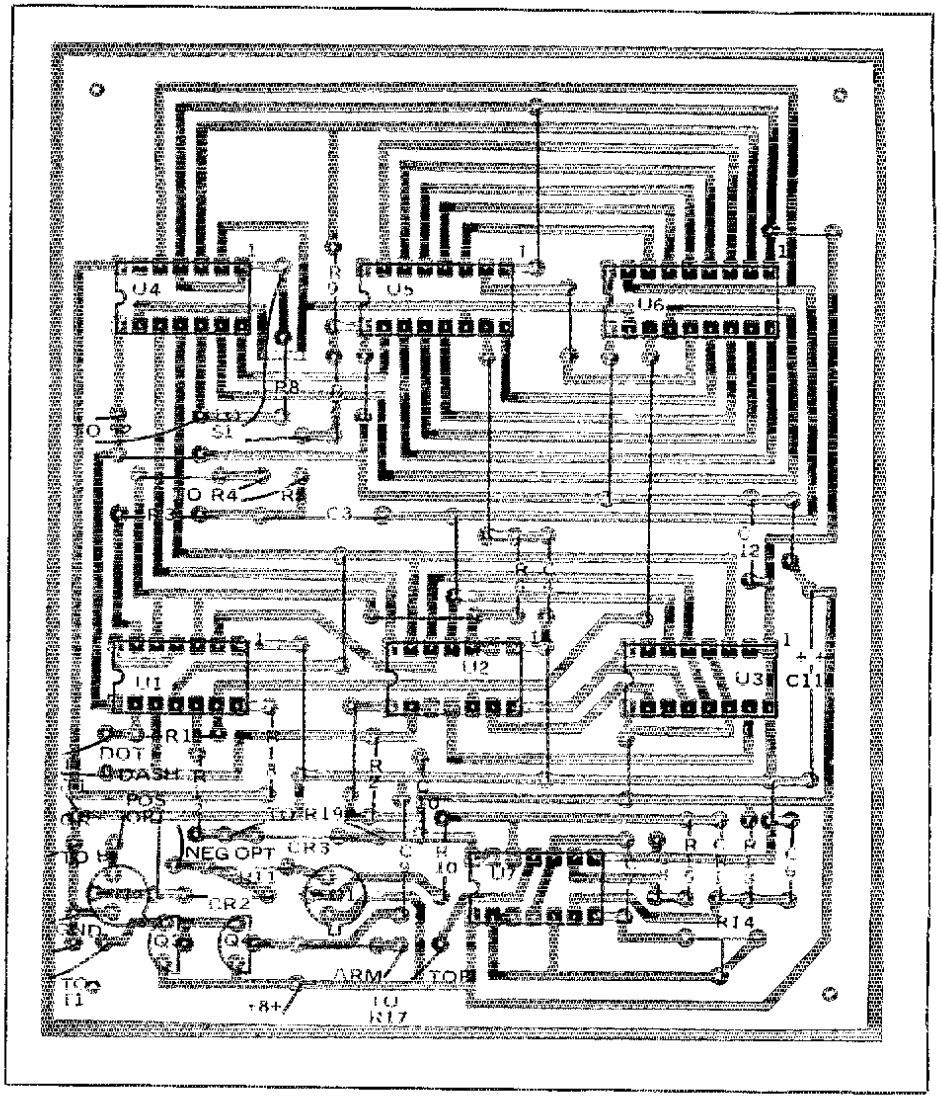


Fig. 2 — Circuit-board etching pattern for the Micro-TO Message Keyer. The pattern is shown at actual size from the foil side of the board with gray representing copper. Refer to schematic and accompanying parts list for identification of components; all components are mounted on the nonfoil side of the board.

Strays

PRACTICE FIRST ON P76-5

☐ OSCAR 8 fans can practice tracking and tuning up their 435-MHz converters with the P76-5 satellite. This Stanford Research Institute spacecraft transmits continuously on 435.974 MHz, with a power of about 200 mW. Its 1000-km (600-mile) altitude corresponds well with the projected orbit of OSCAR 8. So you can debug and Murphy-proof your ground station, and even check for possible transmitter-receiver interaction problems, with this bird. (Warning: The third harmonic of the uplink frequency is close to the downlink frequency and may desense your receiver or converter.)

The accompanying table contains reference orbit information for February. To determine equator crossings of future orbits, add the period (105.729 minutes) and the orbital progression (26.432 degrees per orbit) to the given figures. The orbit crossing marks on the OSCAR 8 OSCARLOCATOR will introduce an error of about 10 degrees per day. For more precise tracking of the P76-5 satellite, see Thompson, *QST* for November, 1975, and the data below.

By the way, the P76-5 satellite transmits on many different frequencies, all in phase, as part of an ionospheric propagation study. The satellite is often off on Sunday evenings, local time. — *WB2CHO*

The P76-5 Satellite

Period: 105.729 minutes
Inclination: 99.655°
Apogee: 1025.968 km
Eccentricity: 0.045°
Progression: 26.432° per orbit
Reference orbits for February, 1978:

Day	UTC	°W Long.	Day	UTC	°W Long.
Feb. 1	0020	188	Feb. 15	0036	196
Feb. 2	0042	198	Feb. 16	0116	206
Feb. 3	0122	208	Feb. 17	0011	190
Feb. 4	0017	192	Feb. 18	0051	200
Feb. 5	0057	202	Feb. 19	0131	210
Feb. 6	0137	212	Feb. 20	0026	194
Feb. 7	0032	195	Feb. 21	0106	204
Feb. 8	0112	205	Feb. 22	0001	188
Feb. 9	0006	189	Feb. 23	0041	198
Feb. 10	0047	199	Feb. 24	0121	208
Feb. 11	0127	209	Feb. 25	0015	191
Feb. 12	0021	193	Feb. 26	0056	201
Feb. 13	0102	203	Feb. 27	0136	211
Feb. 14	0142	213	Feb. 28	0030	195