MISCELLANEOUS RADIO
A Better Way to Send CW?

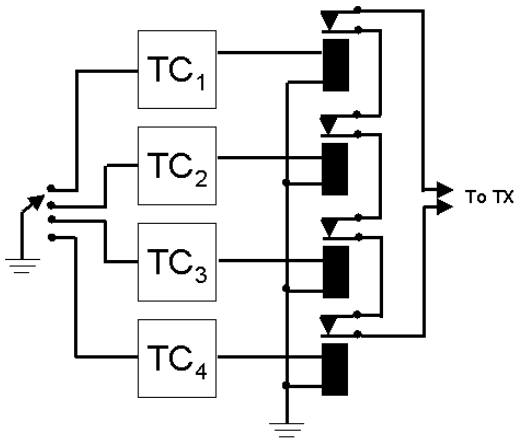
Morse code has been sent via radio in a number of different ways over the years. Most of us probably have sent it using a “straight key” such as a J-38 … up and down, with down closing the contacts and turning on the transmitter. Tap it and you send a dot. Hold it down for 3 times as long and you send a dash. Leave it up for a “tap time” and you space one dot or dash from the next. Leave it up for 3 of those times and you space one letter from the next. And, leave it up for 7 of those times and you space one word from the next. Mechanically, it is very simple, but it puts the entire problem of timing on the operator. You could turn your J-38 on it's side so now it was back and forth rather than up and down [called a “sideswiper”], but the timing problem is still there.

It didn't take very long for the electromechanical bug to arrive. It was back and forth like a sideswiper, one direction just closing the contacts for dashes which you made just like a straight key. The other direction started an armature vibrating on a flat spring which made dots automatically with approximately the right spacing until the armature quit vibrating. Bugs were a lot easier to use once you mastered it, and now the dot timing was fixed at however you set the weights on the armature. Some bugs were machinists' works of art, shiny, and with velvet lined wood boxes to carry them around. Some, like my WW2 J-36, built by Lionel of model train fame were decidedly proletarian in appearance. In fact, mine looked like it had gone ashore with the Marines at Iwo Jima, but it worked fine and only cost $5 at Surplus Sam's in Los Angeles. This was a decided advantage since all of my earnings were going into my college fund right then.

Then came electronic keyers … 10 or so dual triode vacuum tubes which made both dots and dashes with the correct timing, and keyed your transmitter with a relay. We often modified our bugs to bring out separate dot and dash connections and prevented the armature from vibrating to operate our keyers. Soon paddles came along, separate contacts on each side to operate the keyer. And progress brought us iambic keyers with dual paddles that could go right, left, or be squeezed together to create alternating dots and dashes. The electronics migrated from vacuum tubes to solid state transistors and finally to microcircuits.

In all cases however, and despite all of this progress, the timing issue remained ... we “send” the dots and dashes … the spacing between them is an open issue. Theoretically, an electronic iambic keyer could produce perfect code, but the operator still had to operate it correctly to get the right spacing. Is there a better way? Well, the answer is “maybe,” and to find it, we need to think out of the box so to speak, and the key [no pun intended] may be “spacing.”

Since the electronic keyer makes perfect dots and dashes, the real problem we're confronting for perfect Morse sending is the spaces which, except for the one dot-time spaces between consecutive dots and dashes made by the keyer, are still under our control. It turns out there are three different length spaces in Morse, measured in dot-times: one [between dots and dashes], three [between letters], and seven [between words]. More careful study reveals that the one-dot time inter-element space really comes in two flavors … one dot-time space followed by a single dot time key-down [which would be the dots], and one dot-time space separated by 3 dot times key-down… the dashes. And, this is the “Ahh Ha!” moment. It's the spaces that matter, the dots and dashes fall out automatically! It would appear that all this time, we've been “sending” dots and dashes when we should have been “sending” the spaces between them!



As is often the case, this is not a new revelation. A. F. Scotten, W6ZMZ, realized this simple fact and wrote about it in the October 1949 issue of QST. He even came up with a keyer circuit to accomplish it which he somewhat humorously named “The Humbug.” I've drawn it up for you.

His scheme was pretty simple – four normally closed relay contacts put the transmitter into key-down. I'm presuming that he would have some sort of normally open foot switch or the like to keep the transmitter off until ready to send.

Those boxes marked “TCn” are timing circuits, in his day likely free-running multivibrator circuits each using a couple of triodes. Each one produces one of the four spaces we need. One sends by selecting them with some form of a 4 position switch. When none of the four are selected the transmitter is in key down state. Each one “sends” a space of one of the four flavors followed by key-down.

He even offered a test sentence to demonstrate the difficulty of perfect spacing: *"Then after Richard had arrived he and Clarence each kissed beautiful Annabelle and she ceased all resistance because in actual fact she liked it better than ever"*

Ignoring for the moment that this sentence was surprisingly racy for QST in 1949, its dominant feature is that it does not contain two or more consecutive dashes. He maintains that it is very difficult to send this in Morse with any of the various keyers of the day [iambic keyers hadn't been invented in 1949]. [1]

Today? Well, we'd use solid state devices instead of relays, and we'd likely build the timing circuits onto a programmable microcontroller. If I was building this, I'd replace his 4-position switch with four individual SPST switches mechanically linked to 4 paddles that you depressed, similar to the keys on those stenographs that court reporters use. Several configurations come to mind: four keys right next to each other that you just put the fingers of one hand over, or possibly two sets of two using two fingers of each hand. In fact, the court reporters actually press more than one key at a time to make some of their shorthand marks. It's called a "chordal keyboard" since it's like playing chords on a piano. Programming our microcontroller to recognize single, double, and triple depressions, we would need only 3 switches. The code might be:

KEYS			
1	2	3	ACTION
X	X	X	TX on
X		X	TX off
X			Dot Space
	X		Dash Space
		X	Letter Space
	X	X	Word Space

Since the controller will make perfectly timed spaces, you'll automatically get perfectly timed dots and dashes as well. It would have been very difficult in 1949 to have the controllers remember the key combinations in a buffer and send them perfectly even if our keying was a little erratic, but today, adding that sort of memory is trivial.

Back in 1956, only machine-sent CW [such as from a Boehm tape keyer which sent dots and dashes] was perfect, were perfect despite the fact that the solution for perfect CW was at hand and published publicly. There is no indication that W6ZMZ ever attempted to patent his brilliant and novel creation. Why it did not revolutionize CW sending on the ham bands is a mystery. It is a deeper mystery however why his article appeared in the October 1949 issue of QST and not in the April issue.

1 I don't have any problem sending this on my Winkey or K3, but he seemed to think I would.