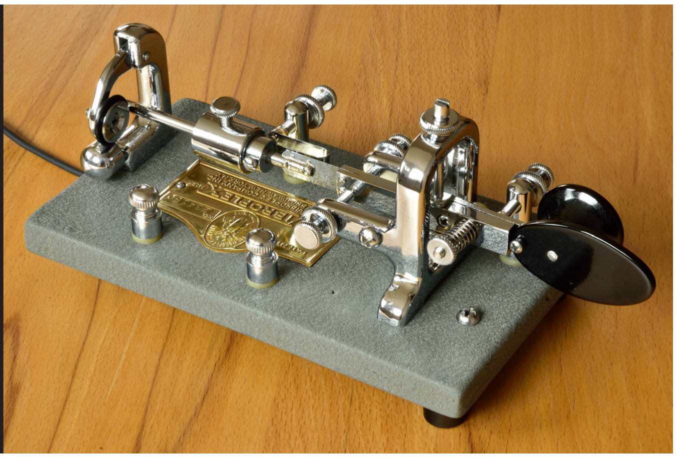

DeBugs

A PIC-based Debouncer for Bugs, Sideswipers and Straight Keys

If you are a serious semi-automatic key operator, you will always keep the contacts of your "bug" shiny clean and perfectly adjusted. But even if you do so, chances are that your transmissions sound filthy. The reason is *contact bouncing*, a common problem with all unwetted mechanical contacts, even those of high quality Morse code keys or keyer paddles. Because moving contacts have mass and springiness with low damping, they will be bouncy as they make and break. That is, when an open pair of contacts is closed the contacts will come together and bounce off each other again and again before coming to a rest in a closed position. Note that contact bouncing is common on closing and uncommon but also possible on opening. Because of their contact spring, particularly the dot contacts of semi-automatic keys are prone to heavy and sustained bouncing, which results in the *"scratchy dot syndrome"* that can sometimes be heard on the amateur radio bands.

Back in the 1930's the manufacturer of telegraph keys T. R. McElroy came up with a little mechanical gadget called *"dot stabilizer"*, which preloaded the vibrating dot contact spring by a very small amount in order to reduce bouncing. By the way: McElroy began his career at Western Union as a messenger boy when he was 14, became a telegraph operator at 15, received his commercial radio operator's licence at 19, and in 1939 he set the all-time speed record for copying morse code with 75.2 words (376 characters) per minute. Another simple and popular electrical approach to reduce not the mechanical bouncing itself but its effect on transmitter keying is a capacitor across the key contacts, which together with the Ohmic resistance of the keyed stage forms an RC low-pass filter.

However, there is not one single possible mechanical or analog electrical debouncing method which could reliably prevent this effect, but on the other hand all of them will more or less affect the keying characteristic. Contrary to that, **DeBugs** is a *digital* debouncer for bugs, sideswipers and straight keys and able to reliably prevent the effects of contact bouncing without affecting the keying characteristic. In the following chapters I will explain its simple construction and theory of operation.

construction

The schematic is shown in fig. 1, programmed PICs are no longer available from the author but the firmware hex-file can be downloaded [1] for non-commercial use. The transistor is any general-purpose NPN type. For voltage supply I recommend a CR2032 3 V coin cell, the very low current drain of approx. 30 µA in sleep-mode makes a switch unnecessary and permits an estimated service life of about one year.

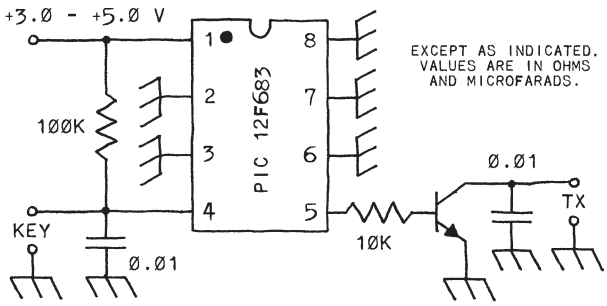

fig. 1 - *DeBugs* schematic diagram.

Because the circuit is so simple, no circuit board layout is presented here. Instead of an etched board I prepared a 45 x 25 mm piece of single-sided PCB (printed circuit board) material, sketched the outlines of the copper traces and milled them out with a dremel tool. This is my preferred construction method for simple circuits and I also build sturdy cabinets from solder-joined plates of this material. At first I prepared the programmed PIC by clipping off the thin parts of its pins, then I soldered all parts in place in the following order: first the 20 mm SMD coin cell holder, then the PIC, two resistors, two 10 nF capacitors and finally the NPN transistor with the flat side of its case pushed firmly down onto the board. The assembled board is shown in fig. 2.

Karl Fischer, DJ5IL, Friedenstr. 42, 75173 Pforzheim, Germany, DJ5IL@cq-cq.eu, www.cq-cq.eu

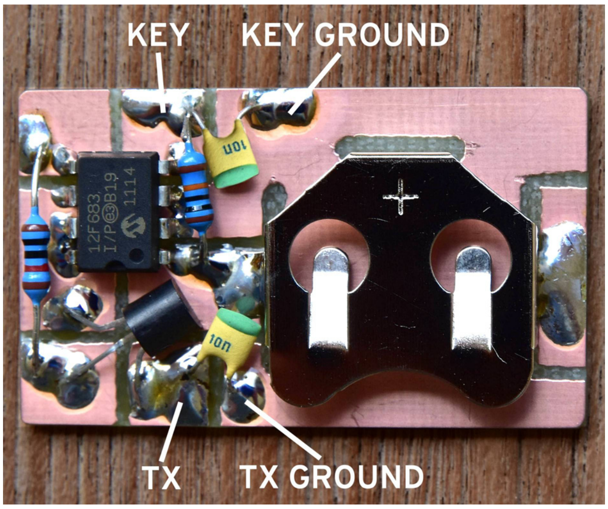

fig. 2 - The assembled board.

It is only 6 mm thick and I used double-sided sticky tape to attach it to the underside of my left-handed *Vibroplex Original*[2] bug shown on the previous page. The four terminals of the circuit are connected as follows: KEY goes to the isolated contact and KEY GROUND to the grounded contact or base plate of the bug via a pair of two short hookup wires (make sure that *no* additional capacitors are connected across the key contacts !), the center conductor of the keying line is soldered to TX and its braid to TX GROUND. Make sure that the coil cell holder does not make contact with the ground plane of the board, after insertion of a CR2032 battery *DeBugs* is operational and goes into sleep-mode until the key contact is closed.

The foregoing description is meant to be just an incentive and suggestion how *DeBugs* can be built and integrated. If you want to use it with several keys you should consider to build it into a small cabinet with two connectors, and if you choose a cabinet large enough to hold three AA batteries you can expect a service life of several years.

theory of operation

The 12F683 runs with a 4 MHz system clock generated by its internal oscillator. The minimum supply voltage is 2.0 V, powered with 3.0 V the typical current drain is 0.6 mA keyed and 30 µA unkeyed (sleep-mode).

The following oscilloscope screenshots visualize how effective *DeBugs* works. The lower (magenta) trace is the voltage across the KEY contacts, the upper (yellow) trace is the voltage at the base of the NPN keying transistor. Fig. 3 shows a string of dots generated with the bug at a speed of about 36 words or 180 characters per minute. The dots are about 38 ms and the spaces 28 ms long giving a weight of 58%. Fig. 4 shows an extremely bouncy dot at the same speed with a higher resolution. Though the bug's dot contact is shiny clean and well adjusted, it produces heavy and sustained bouncing. But the processed keying signal is perfectly clean without any bouncing artefacts, and at the same time the original dot and space lengths are preserved so that the keying characteristic is not affected.

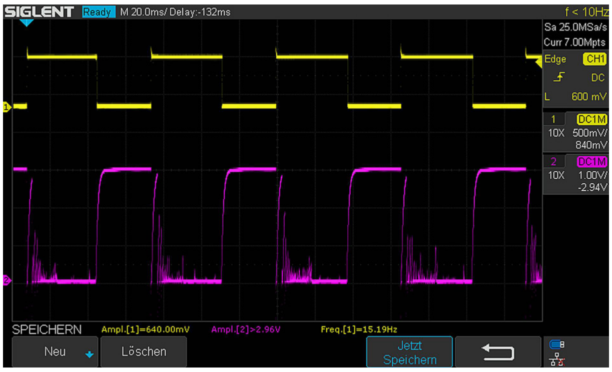

fig. 3 - Dots at a speed of 180 characters per minute.

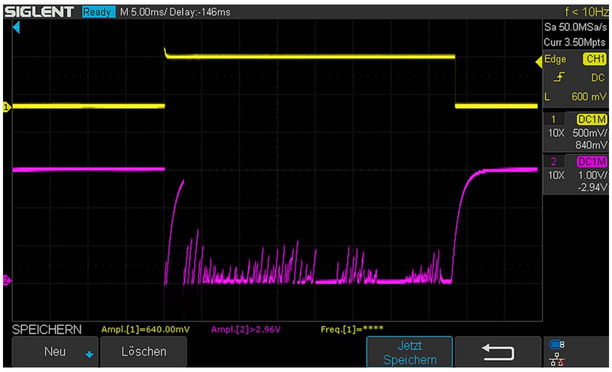

fig. 4 - Extremely bouncy dot in higher resolution.

The firmware performs the following logical steps:

```
MAIN PROGRAM:

1) SLEEP UNTIL THE KEY IS CLOSED

2) SET PIN #5 HIGH TO GROUND TX (TRANS-
MITTER ON) AND GOTO 6)

3) POLL THE KEY UNTIL IT OPENS

4) SET PIN #5 LOW TO UNGROUND TX (TRANS-
MITTER OFF) AND GOTO 6)

5) IF THE KEY IS OPEN GOTO 1) ELSE GOTO 2)

DEBOUNCING LOOP:

6) CLEAR LOOP COUNTER AND MEMORIZE KEY
STATUS

7) IF KEY STATUS CHANGED GOTO 6) ELSE WAIT
16 MICROSECONDS AND INCREMENT LOOP COUNTER

8) IF LOOP COUNTER = 1024 GOTO THE STEP
FOLLOWING THE CALLER OF 6) ELSE GOTO 7)
```

As soon as the key contact maintains its state (stops bouncing), the 16 µs delay loop is executed 1024 times. Therefore the contact state is latched with a minimum hold time of 1024 x 16 µs = 16.4 ms, which equals the dot length at a speed of 73 words or 366 characters per minute. The smaller the number of loop cycles the shorter the hold time and the higher the possible keying speed, but at the same time the possibility for bouncing artefacts on the keying signal increases.

As you can see in figs. 3 and 4 the first bounce is generally the longest. The spring contact touches the fixed contact briefly and bounces back again and again with decreasing energy and duration. The first contact closure has a length of about 3 ms and allows the 10 nF capacitor to be charged through the 100 KOhm resistor almost to the full supply voltage of 3 V. With 16.4 ms the hold time is more than 5 times longer than necessary to trap that first bounce. However, softer springs can produce substantially longer bounces and therefore that safety factor seems appropriate. The subsequent bounces are too short to charge the 10 nF capacitor to the threshold of about half the supply voltage so that no open contact is detected. That's how this capacitor assists debouncing by the firmware logic, and its value should be increased only if bouncing artefacts like prolonged or shortened dots or spaces are experienced.

Please note that though *DeBugs* reliably prevents contact bouncing it does not tolerate dirty or oxidized contacts, so you still have to keep them shiny clean. Whenever you experience chopped or missing dots or dashes clean the affected contact pair as follows: prepare a small strip of very fine grit (> 5000) abrasive paper, insert it into the open contact, close the contact with light pressure and move the strip up and down several times, then turn the strip and repeat that procedure in order to clean the other side of the contact.

references

1. http://cq-cq.eu/debugs.hex
2. The depicted vintage Vibroplex Original bug was operated by the 5th Special Forces in Vietnam in 1966. You can read its interesting story at http://cq-cq.eu/vibro-original.htm

DJ5IL_rt014.pdf
Original version: 27.10.2022
Revisions: 30.11.2022, 14.1.2025