A Picaxe 08M Tone Signal Generator

16 May 2012

It is a commonly found requirement amongst weak signal/microwave enthusiasts that they need to generate a signal tone so that another station can 'peak' their antenna on their station. That 'tone' may be almost anything – from a steady carrier to a warbling note. It doesn't matter what it sounds like provided that it is easily discernable amongst white noise. In other words, a little bit unique, and certainly not like almost any other modulation likely to be encountered.

During the recent microwave activity day here in Brisbane, I was asked a few times if I could 'put up a signal' so that others could peak their antennas on me. My only option at the time was to transmit a carrier signal – by no means easily distinguished in the noise. I decided that as time permitted, I would build up a project using a PICAXE to do a bit of tone generation.

I already knew what I wanted to do electronically due to my previous experience with PICAXEs in general so I sat down at the computer and laid out a printed circuit board with the required parts, including an audio amplifier to drive a loudspeaker. I also allowed for direct audio connection back into the transceiver transmit side as well as the likely holding of a microphone directly over the unit's speaker itself. I didn't actually draw up the schematic until after I did the PCB layout, something that I find helps me minimize layout errors.

In reality, I made the PCB far smaller than I needed to but these days most of my projects involve operation at microwave frequencies where everything is compact by necessity. The basic board size is 35mm by 35mm, and that is enough space to hold a standard 8-pin DIL PICAXE (an 08M), a 5 volt SMD regulator, an SMD-style LM386 audio amplifier plus extra SMD components for easy switching and filtering. If I had to do the layout again from scratch, it would probably be nearly double the physical dimensions just to make it easy to build.

The PCB was created, the parts attached and then the software side started. I found a morse code (CW) routine that was an excerpt from an old Silicon Chip magazine article back in 2003, modified it slightly for the output port value and then loaded it into the 08M. Morse issued forth as per the pre-programmed message. This was looking healthy. The only thing was that the "volume control" feeding the audio level into the LM386 was almost at minimum and the notes wasn't pure, and was accompanied by a lot of 'clicks'. A quick look with the CRO showed the square wave was being provided with rising peaks on both the positive and negative pulse transitions by the 100nF coupling capacitor and the resistive load. Given the very high level of the audio derived from the toggling of the PICAXE's pin, I added a pair of silicon diodes back-to-back as a hard amplitude limiter after the coupling capacitor and then fed that via an RC low pass filter before it reached the 'top' of the volume control. That made the audio a lot more pleasant to listen to, the clicks virtually disappeared and the waveform on the CRO was now a square wave with rounded leading and trailing corners. Not quite a sine wave but something now starting to resemble it.

The audio side was now virtually resolved so it was again time to see what other noises I could make the PICAXE produce. The morse identification was one option, a more-or-less continuous audio tone was another, but something more distinctive was wanted too. I used the 'sound' command throughout a few subroutines then called different ones depending on which input pins were "grounded" by the selector switch. Looking back, it would probably have been simpler to use the A/D function to check a voltage out of a resistance tree wired to the switch contacts rather than seeing which inputs were logically grounded... ain't hindsight wonderful.

I used the same tone frequency value for the continuous tone as I did for the morse ident, and to make the 'continuous tone' a little more obvious, it has a brief pause (a tick effect) around about every second. The 'distinctive sound' segment was created by varying the tone frequency up and down at a 2-3mS rate for four complete cycles before doing a subroutine return. If that function was still required, it would loop the subroutine again.

For information, I used a two pole 6 position function selector switch. Position 1 was power off, 2 was a CW ident loop, 3 was for a continuous tone (with tick) followed by a CW ident, position 4 for the distinctive rising-falling warble, position 5 for the warble then CW ident then continuous tone. The second pole was used for power switching and all but the first position were wired.

The PCB took about ½ hour to build up once the board was produced, the software for the PICAXE probably another 2 hours while I tried to make it create different sounds. The total cost was probably in the order of $20-$25 plus the box to put it in. I still had a few plastic boxes available, plus some 50mm 8ohm speakers scrounged from discarded computer cases so a few well-placed holes later, the project was about ready to use.

For information, the current consumption for a 9V battery supply was around 25mA while in the constant tone condition, averaged around 17-18 mA during the CW ID phase.

Could I enhance the software to provide more tone functions?. No, not with an 08M chip as there are only about 8 program locations free.
Do I need to - No, it will do me just fine as it is at present !

I omitted to take photos of the underside of the PCB before I mounted it into the plastic case but a few outside and inside views are available below.

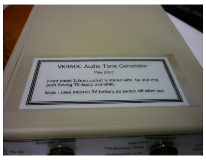
The completed project, overall view.

The jack at bottom LHS is audio direct to the associated transceiver.

The speaker volume and function selectors are rotary controls.

Closer view of the front panel layout.

The layout was created in MS Powerpoint, laser printed, cut to size and covered with a "cold laminate".



The label on the top.. Again, "cold laminated" such that the sticky clear laminate sheet extends beyond the label area thus sticking to the lid of the box.

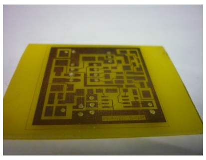
The inside view showing the rear of the rotary function selection switch, speaker volume controls, plus the speaker and the 9V battery at bottom (held in place with double-sided tape).

The small black trimpot on the top of the PCB is pre-set to give a suitable level at the 3.5mm socket - enough to drive the transmitter to close to 100% power on SSB.

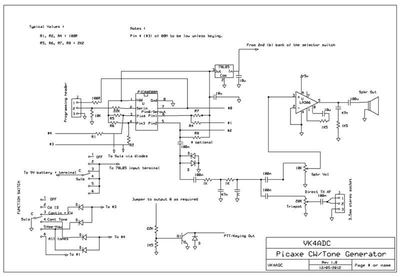
The 3-terminal header pins just the other side of the 08M PICAXE is the "programming header" connection.

A slanted view to show the front panel 3.5mm socket a little clearer.

I decided to mount the 100uF DC isolation capacitor at the speaker itself rather than trying to fit it on the small-ish PCB.

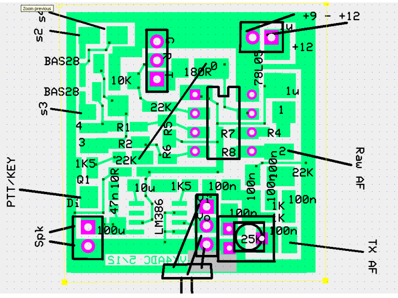


The PCB after etching..



Schematic as a PDF (/~vk4adc/web/images/UserFiles/File/tonegen/PICAXE%20CW%20Tone%20Generator.pdf)

PIcaxe 08M code BAS file (/~vk4adc/web/images/UserFiles/File/tonegen/Picaxe%20tone%20gen.bas)



The PCB layout used. Note that the small dots comprising the layout grid are at 1mm intervals.

Samples of 'sounds produced' ...

These were recorded via Windows Sound Recorder via a stereo lead from the TX AF socket on the front panel, and then converted to a compact MP3 format for use on the web site.

(/~vk4adc/web/images/UserFiles/File/tonegen/cwid.mp3)

CW Identification

(/~vk4adc/web/images/UserFiles/File/tonegen/tone-cw.mp3)

Tone followed by CW ID

(/~vk4adc/web/images/UserFiles/File/tonegen/conttone.mp3)

Continuous tone effect, with 1 sec tick

(/~vk4adc/web/images/UserFiles/File/tonegen/heehaw.mp3)

Heehaw / siren effect