This website has more than enough information about the MT8870 DTMF decoder.

<http://arduinobasics.blogspot.com/2015/08/mt8870-dtmf-dual-tone-multi-frequency.html>

This is where I got the basic sketch from.

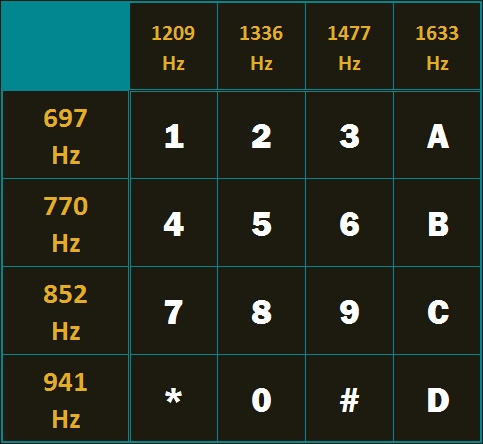
What is DTMF:

<https://en.wikipedia.org/wiki/Dual-tone_multi-frequency_signaling>

**Dual-tone multi-frequency signaling** (**DTMF**) is an in-band telecommunication signaling system using the voice-frequency band over telephone lines between telephone equipment and other communications devices and switching centers. DTMF was first developed in the Bell System in the United States, and became known under the trademark **Touch-Tone** for use in push-button telephones supplied to telephone customers, starting in 1963

What that really means is it is a way to transfer data over a phone line (or airwaves) using a frequency humans can hear, and computers can decode. It has two tones to each signal, making a matrix.

DTMF is used in a telephones, to open garage doors (fire signals, ok it’s slightly different than standard DTMF, but same idea), control radio repeaters, and a number of other things.





The above is a diagram of the output of the MT8870 decoder chip, this can be used to make a sketch for the arduino to read the codes.

IN the sketch, the numbers 1 - 9 return the value of the key pressed. Zero returns the value 10.

Is 11, # is 12, A is 13, B is 14, C is 15, and D returns Zero. (yes, I thought that was strange too, but if you look at the above diagram you soon see why that is)

In the sketch - each time Q1 goes high 1 is added to a DTMFread value,

When Q2 is high 2 is added to that value.

When Q3 is high 4 is added to that value.

And when Q4 is high 8 is added to it.

The two STQ lines are signaling lines to let the microcontroller that the MT8870 has decoded a tone. Only one is needed or used in this sketch, and for the sketch it’s the one that goes high when it has data.

And the Keypad matrix is a HEX matrix (digits 0 to 15). Touchtone phones only use 12 of the same 16 numbers/letters - normally leaving the alpha letters A, B,C,D from the matrix.

For the BittyBot Rover, I took the sketch from the blog, and modified it with my “Drive with class” sketch to make the Rover move around using my tablet (Anything that can make DTMF tones will work thou). The MT8870, has a stereo input, that works very well, it also looks like it has a line level input - put I haven’t gotten it to work right, the mic needs amplified, and the tones need to be pretty loud.

For the robot, I used the number 2 to move forward, the number 8 to move in reverse.

The number 4 to turn left, the number 6 to turn right. The number 5 is a all stop.

The \* and # are used to increase and decrease the speed.

The other numbers and letters are not used in my sketch but could be added for additional features later.

So my final thoughts, thou this does work for the BittyBot Rover, it’s impractical because of needing to be tethered to the phone. IT can be used as a good demo for something to do with DTMF decoders thou.