Lab11: DTMF Decoding

EECS20: Introduction to Real-Time Digital Systems

©1996 Regents of the University of California.
By K. H. Chiang, William T. Huang, Brian L. Evans.
URL: `http://www-inst.eecs.berkeley.edu/~ee20`
News: `ucb.class.ee20`

Assigned: 23 Apr 97, Checkoff: 30 Apr 97, Writeup Due: 2 May 97

1 Introduction

Dual tone, multi-frequency (DTMF) is the standard used in encoding digits for transmission between the customer and the centeral office. If you open up your phone (something for which you could be fined heavily back in the good old days), you'll see a four row, three column matrix switch. Each row has a unique frequency assigned to it and likewise for each column, so each button has a unique two frequency combination.

	1209Hz	1336Hz	1477Hz	1633Hz
697Hz	1	2	3	A
770Hz	4	5	6	B
852Hz	7	8	9	C
941Hz	*	0	#	D

Figure 1: DTMF tones.

When you press a button, the telephone determines the row and column of that button and generates two tones at the two frequencies associated with that button. At the central office, the telephone switch decodes those tones.

Decoding can be accomplished by using a bank of eight bandpass filters, one for each of the eight possible tones. If two of these tones are detected, it is assumed that the key associated with both of these tones was pressed. The design of these bandpass filters is greatly eased by the use of the frequency domain and the Fourier transform.

Hey! How come the telephone has a 4x3 matrix switch, but the table above is 4x4? Well, originally there were functions assigned to the lettered buttons, but the digits were deemed sufficient. The military does use those four buttons though (flash, priority, override, and something else that i can't remember right now), so that the general can order his caviar while the poor sod in the trench eats some more shells.

2 Writeup

1. Using the table above, create a matlab script that takes a set of the digits from 0 through 9, #, and *, and generates the DTMF tones associated with those digits; i.e.

```
>> sound(dtmf(31937),fs)
```

should dial the lab phone number [feel free to annoy your friends and relatives by calling them up with your generated DTMF sequences; sorry, local calls only]. Use $F_s = 8192$ kHz. To ensure that your sequences will actually dial the lab phone, make sure that each tone is at least 40 ms in length, and the pause between tones is at least 40 ms.

2. Using the filter design techniques you learned previously, construct a bank of seven filters to decode the DTMF tones above. Debug your filter bank with your DTMF tone generator from above. Design your filters with $F_s = 8192$ kHz.

3. Decode the given mystery DTMF sequence. This should be available either here or in the `C:\TEMP` directory on the lab machines. Use:

   ```
   >> load mystery -ascii
   ```

   to load the mystery sequence named `mystery` into Matlab.

4. From examining the filter outputs, you can visually determine what digits were in the sequence. How could you automate this procedure?