CHAPTER 20
Practical 2
Phone tones

Aims

In this practical we explore working from specification. Sometimes you get handed everything you need and the important task is to implement it as faithfully as possible. Imagine you have received a script for the following scene:

spy 1: *Picks up telephone* **(sfx: Dialing tone from handset)**
spy 1: *Dials number* **(sfx: Ringing tone from handset)**
spy 2: "Hello, this is the Badger."
spy 1: "This is Fox. The dog has the bone, the seagull flies tonight."
spy 2: "Good, Fox. Now the Americans will pay for their deception. . . hold on. . . "
(sfx: click - telephone line goes dead)

Create the sound effects for telephone tones heard through the handset when making the call.

Analysis

These are the sounds heard on the receiver, through the handset. The first two correspond to different signalling states within the phone system that occur before both parties are ready to talk and the system switches to a voice link. The dial tone is a constant low frequency purring sound that indicates the system is ready to make a call. Normally it is followed by dialling the number, done either with DTMF tones,[1] or with pulse dialling. If a number is recognised by the exchange the ringing tone occurs. It is a higher pitched broken tone that occurs between dialling a number and the other person picking up.

Model

The signals are electronic in nature. They are specified by a standards document that gives the ideal model so there is no work to do here but implement what we are given. The tone specifications are explained in the CCITT standard for telephony as follows:

[1]DTMF tones are examined in a later practical

Tone name	Frequencies	Modulation	Purpose
Dial tone	480Hz + 300Hz	Continuous	Indicate ready to receive
Ringing tone	480Hz + 300Hz	On 2s, off 4s	Indicate remote ring

fig 20.1: Table of signalling tones

Observation point

This makes a nice example to explore the observer concept. How does what the listener hears differ from the ideal model? There are three possible scenarios not explained by the above script. We could be listening through the ears of Fox, talking to his contact. We would hear the sounds through the handset, loud and close. Alternatively, the audio scene may be from the viewpoint of a third person in the room with Fox. We would hear Fox speaking with room acoustics, but the voice of Badger and the dialling tones as thin, distant and filtered. Finally, we might "zoom out" to reveal Special Agent Smith listening in on a telephone tap. From his viewpoint the signals come directly from the line and both voices and tones are treated accordingly. For this example let's assume we are listening from the perspective of Fox, the first spy.

Method

We construct both the tones by addition of sinewaves. There are only two frequencies in each so the job is easy. `osc~` objects will be used for this. To make the ringing tone broken we modulate it with a low frequency control signal in the message domain. Next we construct a crude model of a telephone line and handset that adds distortion and bandwidth limiting using `clip~` and `bp~ 1`, then listen to the dialling and ringing sounds through it.

DSP Implementation

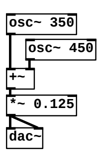


fig 20.2: CCITT dialing tone

First create a sinewave oscillator `osc~` object. Set its first and only creation parameter for frequency to 350Hz. Now copy this object using `CTRL-D` and place the copy close to the first oscillator. Change its frequency to 450Hz. Connect both of them to one `+~`, each to a different side. This explicitly adds the signals. Remember that signals are *implicitly* summed, so this patch could be done without the `+~` object, but it is a nice way to make clear what is happening. To scale this to a reasonable listening level we multiply by 0.125. Finally connect to both sides of the DAC and you should hear the dial tone (Fig. 20.2). In the classical pre-digital telephone system tones are produced at the exchange, not the handset itself, since they are part of the signalling protocol. The observation point is therefore at the end of some channel or connection, classically an electrical connection that is very long and therefore far from ideal. Also the signal will be observed through the handset transducer, a small loudspeaker with a limited frequency range. What will this combination of telephone line and handset do to the signal? Full analysis of the line, which is a complicated affair involving the inductance, capacitance and resistance of the wire is unnecessary since we are making an approximation. It's enough to know that the effect of passing through the line is some distortion, a loss of some frequencies, and accentuation of some other frequencies. The line and handset behave like a cascade of bandpass filters.

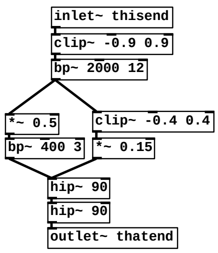


fig 20.3: Approximation of transmisson medium

One inlet and one outlet are connected by a chain of units to crudely approximate a phone line and handset. The subpatch of Fig. 20.3 appears as `pd tline` in subsequent examples. First some distortion is introduced using `clip~`. This widens the spectrum, introducing odd harmonics and causing some loss at the two original frequencies. Next we mimic the band limiting effect of the wire with a resonant filter centered on $2k$Hz. Both our original frequencies are within the range of the filter response, but what we are interested in is the effect this line filter will have on the extra harmonics from the distortion. Next the general effect of a small louspeaker is added. The sounds we are interested in are around 400Hz, so let's place the centre of our filter there and remove all low frequencies. There will also be some distortion from the loudspeaker, which we add in parallel.

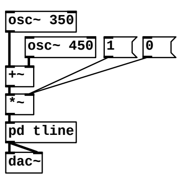
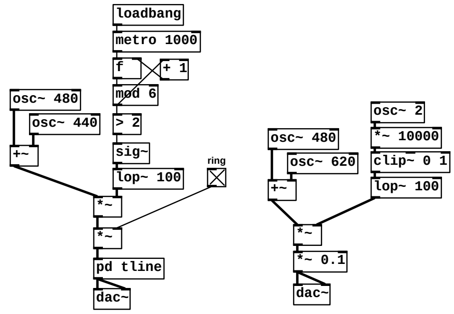


fig 20.4: Dialing tone over a line

Now we can use the telephone line with the dialtone patch. Look at Fig. 20.4 and you will see I've multiplied the dialtone signal by a message rate 1 or 0 to switch it on or off. Try this with the `*~` *following* the line as an experiment. Notice the subtle difference, the change in tone during switching? When switched on the other side of the line from the listener a sudden disconnect drives a high frequency impulse over the channel. The telephone line makes it own sound. The line behaves as a resonator. Patches for the ringing tone and busy tone are shown in Fig. 20.5. They are very similar frequency pairs to the dialling tone but with different modulation timings. Build them to hear the effect and check the timings and frequencies against the CCITT documentation.

Old style pulse dialer

Before DTMF technology telephone systems used pulse dialling. Instead of sending a tone to the exchange the phone sent a series of pulses. The character of this sound is determined by answering the question, where does the energy come from? For a modern cellphone energy comes from the handset. In the case of old pulse dialling phones it comes from the exchange, which sends a current down the phone line. It comes back on the other side of the line carrying



(a) Ringing tone (b) Busy tone

fig 20.5: More signalling tones

voice signals, making a circuit. The sound of a remotely switched current is what we call the *impulse response* of the circuit. When we look at excitation methods of physical bodies later we will see that an impulse equates to hitting something.

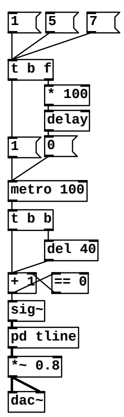


fig 20.6: Pulsedial

An old rotary dialler makes and breaks the line connection to the exchange. The short pulses are DC, so have no frequency except at their start and end points which are step impulses. On each connection, current flows down the line from the exchange and back again. The characteristic sound of an analogue pulse-dial telephone therefore depends almost entirely on the line and handset, upon the character of miles of copper wire and a small plastic box. In Fig. 20.6 a number message causes a 1 to be sent to a metronome with a period of 100ms switching it on. At the same time a delay is scheduled to emit a bang at a time of 100 times the number, which then turns the metronome off. So a message of 7 switches the metronome on for 700ms, and there will be 7 bangs. Each bang from `metro` is duplicated by a trigger and delayed by `delay` to produce a 40ms pulse. This approximates the duty cycle of a typical pulse dialler. The `+ 1` `== 0` is a toggle idiom, with an initial state of zero. It behaves like a counter that can only count 0 or 1, so it's a condensed version of the counter and *mod* 2 operation we used before.

Results

Source .... **http://synthsound.org/sd/phonetones.html**

Conclusions

Sounds can be *defined* as well as existing because of a physical process. They can be given by precise specifications. Telephone dial and ring tones are completely synthetic, man-made things. The observer point, and all intevening processes are relative to the source of energy in this model, and can be modelled by a chain of distortion and filters.

Exercises

Exercise 1

Combine all the effects from this exersise to make a complete "audio scene" with pickup, dialtone, dialing and ringing tone (or busy signal).

Exercise 2

Work on refining the remote disconnect click as heard by a nearby listener. Listen to the sound design from some Hitchcock movies for that classic phone disconnect sound.

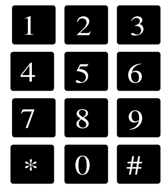
Exercise 3

What causes crackles on a phone line? How would you add these to the line model as an effect?

References

"Technical Features of Push-Button Telephone Sets" in "CCITT Volume VI: General Recommendations on Telephone Switching and Signalling" (International Telecommunication Union) ISBN: 9261010512

CHAPTER 21
Practical 3
DTMF Tones



Aims

Construct a telephone dialer using "Dual Tone Multi Frequency" modulation. The dialler has a keypad containing 16 buttons for the numbers 0 to 9, four letters A, B, C and D, and two special symbols, hash and star. On each keypress the dialer will send a 200ms beep corresponding to the CCITT/DTMF standard tone for that keypress.

Analysis

Begin by researching the CCITT standard and see how audio is used in the dialing or address signalling part of a phone call. The tones are pairings, from a choice of 8 frequencies that are picked for their non-interaction on a noisy audio bandwidth line. The specification sets out some limits like the duration of the DTMF tone, which must be 50ms or more. The minimum interval between digits is 45ms and the maximum is 3 seconds.

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

fig 21.1: Table of DTMF tones

Model

Once again, there is no physical model, all signals are elecronic in nature. They are specified by a standards document that gives the ideal model, so again there is no model to think about, we just copy the specifications as faithfully as possible.

Method

First construct a sub-patch that produces a pair of tones. Create a table using message boxes that maps keypresses onto a set of tone pairs. Then add a keypad to activate the oscillators from entries in the mapping table and operate a control gate to switch them on and off.

DSP Implementation

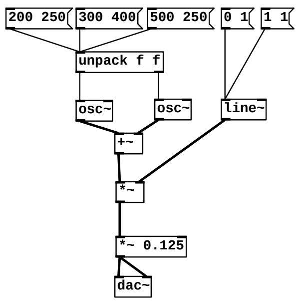


fig 21.2: Dual tone dial signal

The message boxes along the top of Fig. 21.2 represent some test frequencies and two control messages. The first are lists of number pairs, the frequencies of two tones given in Hertz which are unpacked and sent to two separate sinewave oscillators. The sum of the oscillator signals is multiplied by a control signal from a line generator. The two messages on the right are {`destination, time`} pairs that change the state of the line generator very fast, in 1.0ms, to a value of 1.0 or back again to 0.0. Play around with switching the signal on and off and selecting different frequency pairs. If we can control this patch to select the right frequencies and make it switch the tone on then off when a key is pressed the job is almost done. Everything needed to make the dialer work is shown in

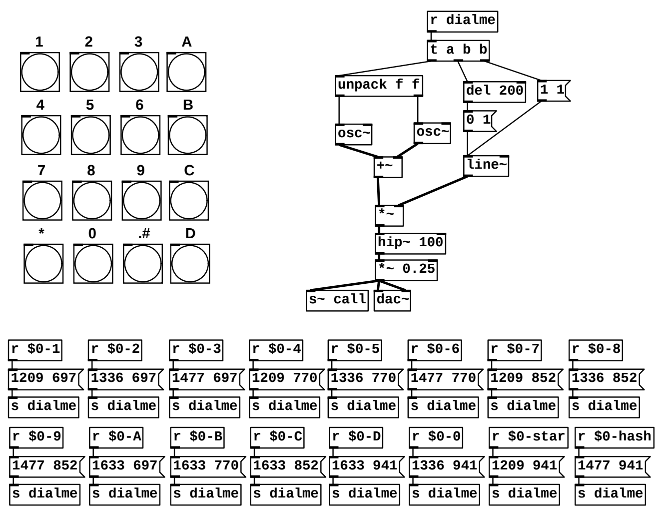


fig 21.3: Keypad and table

Fig. 21.3. Each button in the keypad has its *send-symbol* set to one of the receive destinations labelled `$0-n`. In the table below, these receive objects pick up bang messages and pass lists of tone pairs to the destination `dialme`. Messages received at `dialme` are unpacked and fed to the two oscillators. First

we trigger a message to set the line generator on. After a delay of 200ms a message is sent to return the line generator to 0.0. A final highpass removes any unwanted low frequency components.

Results

Source .... **http://synthsound.org/sd/dtmf.html**

Pressing any of the buttons produces a short beep corresponding to one of the standard DTMF dialing tones.

Conclusions

Lists stored in message boxes can be used to make a table for driving several oscillators. This way we can reuse the same two oscillators for all DTMF tones. A keypad interface can be made by setting the `send symbol` property of each bang button to a message destination.

Exercises

Exercise 1

Try using the key object (if available on your system) to get presses from your computer keyboard to trigger DTMF tones.

Exercise 2

Why exactly are these particular frequencies chosen? Research a little about transmission theory (line propagation) and the distortion of signals and imagine these tones have travelled a bad line with noise added. How might you improve this design to make it more reliable?

Exercise 3 (Advanced)

How would you design a decoder to turn the audio signal back into numbers?

References

"Technical Features of Push-Button Telephone Sets" in "CCITT Volume VI: General Recommendations on Telephone Switching and Signalling" (International Telecommunication Union) ISBN: 9261010512