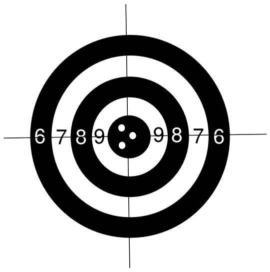
CHAPTER 23

Practicals introduction



“The path is made by walking.” -
African proverb

Practical synthetic sound design

Once you have worked through the next 30 or so exercises you should be ready to tackle most sound design problems from first principles. No set of exercises can hope to exhaustively prepare a designer for every eventuality, so I've picked some common examples that are typical of their general class upon which you can build with further research and experience. Mostly, each exercise increases in difficulty, and later ones will tend to assume you have completed those before. They are split into the following categories;

Artificial

The class of sounds that have no real world counterpart. They can usually be described by a formal specification. Such sounds include telephone dial tones, alarms and indicators.

Natural forces

These sounds result from energetic dynamics in nature like wind, rain, thunder and fire.

Idiophonics

This category includes rigid body collisions or non linear interactions between moving parts that do not change size or mass. Such sounds are produced by friction, scraping, rolling, impacts, crushing and fragmentation.

Machines

Machines extend the idiophonic class into complex man-made devices with several moving parts. This class includes motors, fans and propellers, engines and vehicles.

Life

The most complex sound sources are animate beings. Their material makeup and control systems provide a great challenge to model. Examples are birds, insects and mammal sounds.

Project Mayhem

Here we investigate the class of high energy and high speed sounds where everyday acoustic theory breaks down. This includes supersonic objects such as bullets, shockwaves and explosions.

Sci Fi

A final section that challenges the designers creativity and exercises ideas of metaphor, simile and implication to conjure up fantastic sound effects for unreal objects.

Practical series
Artificial sounds

> “Embrace simplicity” - Lao-Tzu

Artificial sounds

The chapters in this series are about sounds with no real world counterpart. By this we mean things like telephone beeps, alarms or electronic button activates. They are simple sounds. What makes them interesting is they can be generated according to a specification. This illustrates a point, that if given a thorough specification we can proceed quickly to implementation without the more difficult analysis stage. Sometimes the specification is published as a standards document. Other times, a model and method are easy to obtain from simple analysis, as shown in the first exercise. It’s a great place to start because each can be synthesised using simple techniques.

The Practicals

Five practical exercises of increasing difficulty follow.

- Pedestrian crossing, a simple beeping sound. Introduces basic analysis and synthesis.
- Phone tones, making more complex signalling tones from specification. Bringing observer points and energetic analysis into the picture.
- DTMF tones, working from precise specification. Thinking about code reuse and simple interface building.
- Alarms, investigate alert and indicator sounds. Introducing the idea of functional specification and the meaning of sounds (semantics).
- Police, a more detailed electro-acoustic example with some more analysis and synthesis. Explores the idea of data reduction through different synthesis methods.

CHAPTER 24

Practical 1
Pedestrians

Aims

In this practical we will construct a simple beeping tone as used on UK pedestrian crossings and introduce some basic analytical procedures. We will discuss the design and purpose of the beeping and discover there are reasons why it sounds the way it does.

Analysis

This practical was inspired by a discussion on the Yahoo sound design list when a film maker wanted a particular type of British road crossing signal. Being an artificial, publicly recognised sound, it is given by a government standards document. However, getting an audio example was simple enough since I live near to a main road. The recording, captured about 3m away from the source, is shown in Fig. 24.1. Notice the background noise level from car engines and general street sounds. There are three points of interest, the timing of the beeps, their frequency, and the waveform of the signal. Let's begin by measuring the timing. The x-axis scale of Fig. 24.2 is in seconds, so one beep lasts for 100ms. The off time is also 100ms. We call the ratio of on time to off time the *duty cycle* of the signal. In this case it is 1 : 1, sometimes given as a percentage for the on part, thus 50%.

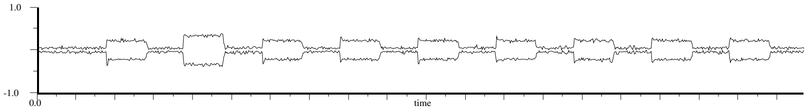


fig 24.1: Recording of a pedestrian crossing signal near a busy road

Next we wish to find out something about the waveform. Experienced ears can guess a frequency below 5kHz with good accuracy. I guessed about 2kHz, but let's see what the spectrum analysis thinks. It is immediately clear from the plot in Fig. 24.3 that there's one strong frequency. The list of numbers on the right side is called the *peaks list* and it shows some weak frequencies at the

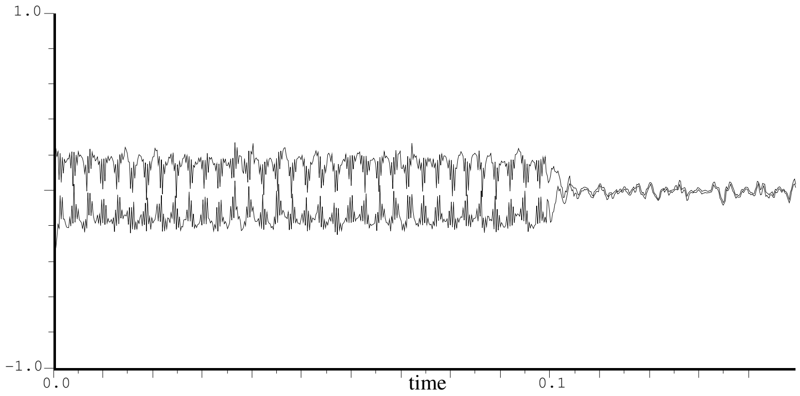


fig 24.2: Measuring the timing pattern of the beeps.

low end of the spectrum, probably originating from traffic sounds. The main peak is given as 2.5kHz. We can also tell from the spectrum that the beep does not have any other significant harmonics [1].

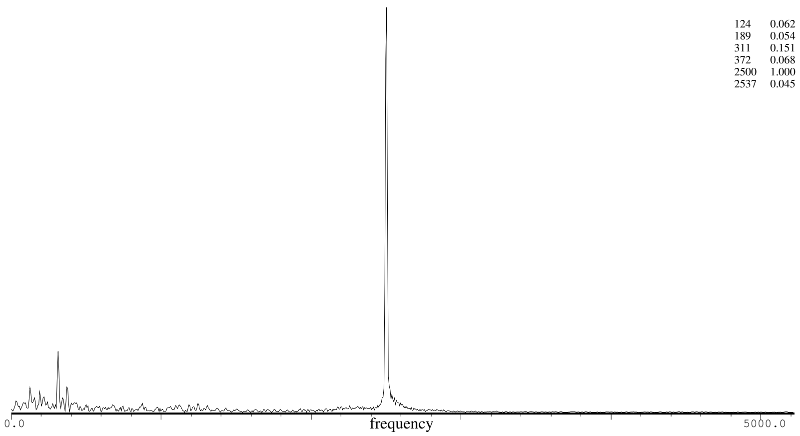


fig 24.3: Spectrum plot of one beep from a pedestrian crossing sound

---

[1] Zooming in on the spectrum plot reveals weak components at 5kHz and 7.5kHz which show it has a little distortion, but we shall ignore this here.

Model

Our model can be succinctly summarised thus: The pedestrian crossing signal is a 2.5kHz sinusoidal wave broken at 100ms with a 50% duty cycle.

Method

We will use a 2.5kHz sine wave oscillator and multiply it by a control signal that alternates between 0 and 1 every 100ms

DSP Implementation

There are several ways to implement the described model even once we decide to use a simple oscillator and control gate. For this exercise I will introduce one simple solution, using a counter.

Counter controlled beeping

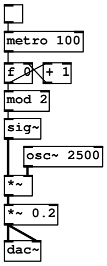


fig 24.4: Crossing beeps

The patch shown in Fig. 24.4 works as follows. A toggle switch activates a metronome with a fixed period of $100ms$. Ten times per second a bang message passes into the hot inlet of a float box which is wired as a counter with an increment of 1. The counter advances upwards without limit. Taking modulo 2 of the counter output gives an alternation between 0 and 1, since $2\, mod\, 2 = 0$, $3\, mod\, 2 = 1$, $4\, mod\, 2 = 0$ etc. From this we derive an audio signal via `sig~` as a modulator. The output of a sine oscillator set to a frequency of 2500Hz is multiplied by the 1 or 0 signal. A fixed scaling of 0.2 is applied to make the output a little quieter. It is sent to both channels of the DAC. Ensure that `compute audio` is switched on. Start the metronome by activating the toggle and you should hear a regular beeping sound.

Results

Source .... **http://aspress.co.uk/sd/pedestrian.html**

Conclusions

By analysing recordings we can extract useful data. A simple indicator sound can be made by modulating a constant tone on and off.

Limitations

One problem is that turning off the metro doesn't always stop the tone. If the state of the counter is 1 at the moment it is switched off it remains that way, with the tone constantly sounding. The result is also somewhat inaccurate. The real crossing sound has some harmonic distortion caused by the transducer, a sudden attack transient caused by the abrupt tone switching, and resonance from its housing.

Practical design considerations

The tone switching causes a click at the start of each tone burst. In this case it is desirable. To see why, consider some other features of the sound. Why choose 2.5kHz?. There are two sides to the road and at least two beepers to aid sight impaired pedestrians (or fully sighted people in bright sunlight). At a staggered crossing where there are several beepers we need to know which one is active for safety reasons. The choice of 2.5kHz is deliberate. It is high enough in frequency to be easily located but not too high to be inaudible to elderly pedestrians. Recall that a sharp attack makes a sound easier to locate. In practice the transducer is housed to make the sound as local to the crossing and easy to locate using IID cues as possible. So the choice of frequency and modulation method is not accidental.

Deviations from specification

The recorded tone did not exactly match the specifications document which defines a range of tolerences rather than precise values. The duty cycle and modulation frequency matched properly, but the tone frequency is given as (as low as) 1kHz but measured closer to 2.5kHz.

Exercises

Exercise 1

Record and analyse another simple indicator sound. You could try a microwave oven timer or a simple electronic doorbell. Specify a model for the sound and synthesise it as well as you can.

Exercise 2

Listen to the sounds next time you cross a big city road. What do you notice about the tone, directionality and timing of the crossing signals? How do you think these help road safety?

References

UK Highways Agency (2005) "TR2509: Performance specification for audible equipment for use at pedestrian crossings"

CHAPTER 25

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	440Hz + 350Hz	Continuous	Indicate ready to receive
Ringing tone	480Hz + 440Hz	On 2s, off 4s	Indicate remote ring

fig 25.1: 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

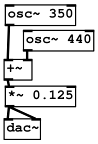


fig 25.2: CCITT dialing tone

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 440Hz. 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. 25.2). In land based telephone systems, tones are produced at the exchange not the handset itself (as for mobile devices), since the tones 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.

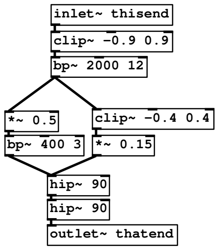


fig 25.3: Approximation of transmisson medium

One inlet and one outlet are connected by a chain of units to crudely approximate a phone line and handset. The subpatch of Fig. 25.3 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 2kHz. 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.

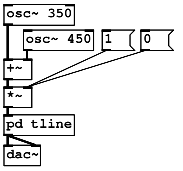
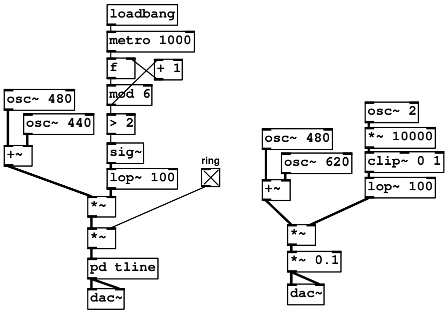


fig 25.4: Dialing tone over a line

Now we can use the telephone line with the dialtone patch. Look at Fig. 25.4 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. Do you notice the subtle difference to the change in tone during switching? When switched at 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 as it behaves like a resonator. Patches for the ringing tone and busy tone are shown in Fig. 25.5. 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 25.5: 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.

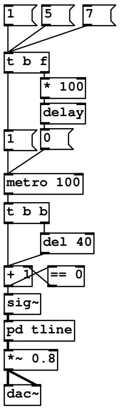


fig 25.6: Pulsedial

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. 25.6 a message causes 1 to be sent to a metronome switching it on. The metronome has a period of 100ms. At the same time a delay is scheduled to emit a bang, 100ms times the incoming number message later, which 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://aspress.co.uk/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. Yet we should take into account all real, physical processes that may affect the sound, like the electrical effect of the telephone line and the acoustic properties of the handset. The observer point, and intevening processes are relative to the source of energy in a model. Here we can approximate the physical effects by a chain of distortions and filters.

Exercises

Exercise 1

Combine all the sound 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?

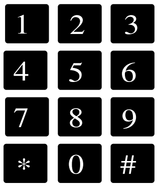
Exercise 4

Create the sounds of a 2600 Baud modem dialling in, establishing a carrier and transferring data.

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 26
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 dialling 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[1]. 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 26.1: 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.

---

[1] Unless a channel that mixes two signals is linear we get intermodulation distortion, new products at integer combinations of the input frequencies. DTMF tones are chosen so that even on a badly distorted line these artifacts won't be confused with recognised frequencies.

Method

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

DSP Implementation

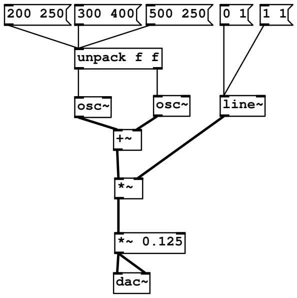


fig 26.2: Dual tone dial signal

The message boxes along the top of Fig. 26.2 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 dialler work is shown in

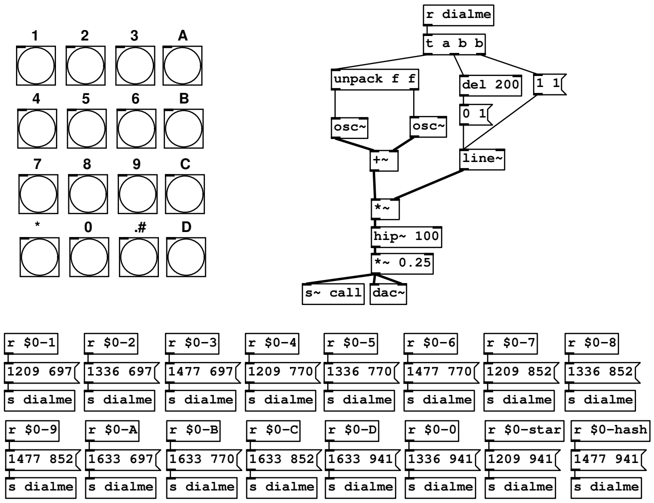


fig 26.3: Keypad and table

Fig. 26.3. Each button in the keypad has its *send-symbol* set to one of the receive destinations labelled `$0-n`. In the lookup section below, a corresponding receive object picks up bang messages and passes a list 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://aspress.co.uk/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 lookup 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