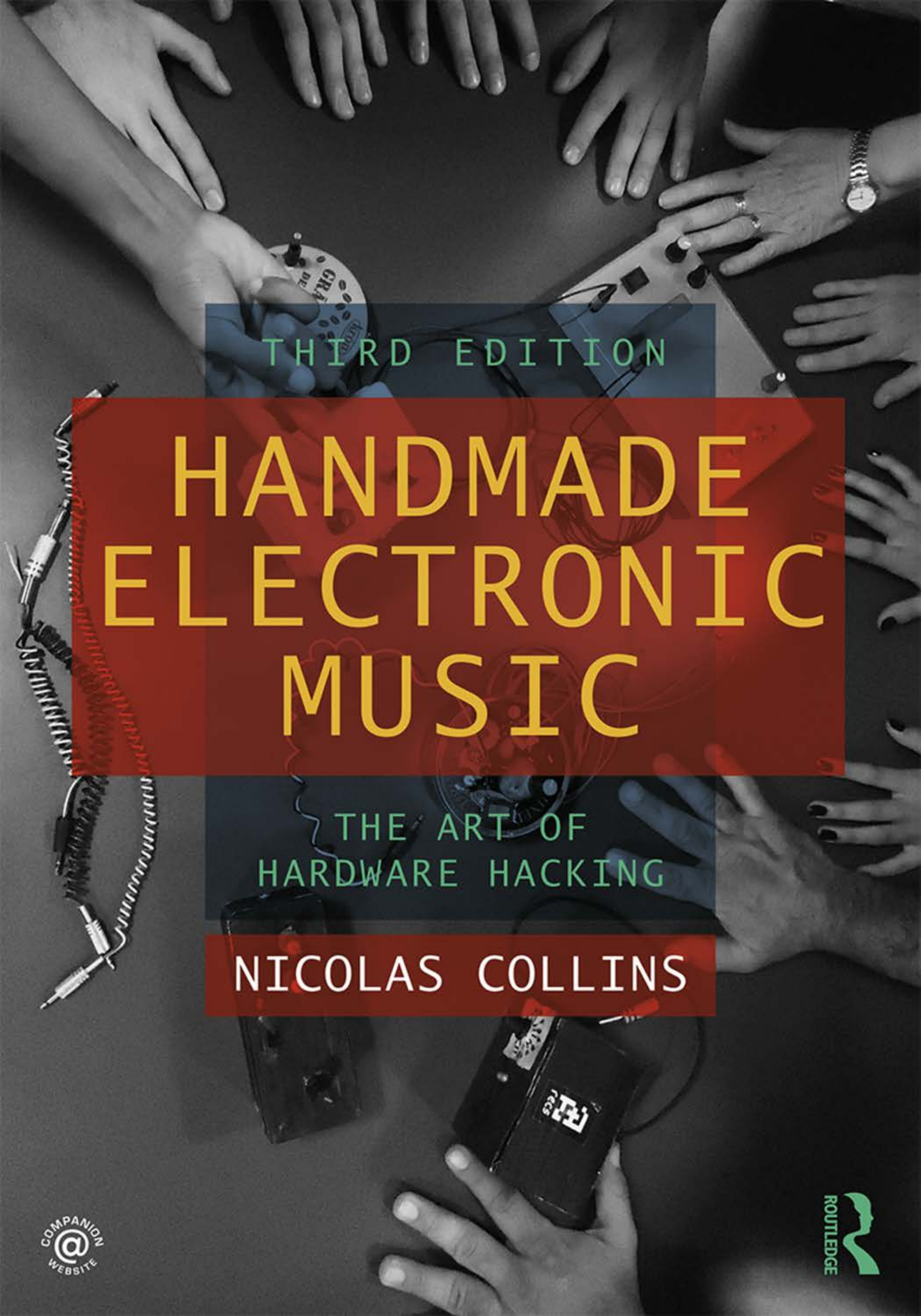


THIRD EDITION

HANDMADE ELECTRONIC MUSIC

THE ART OF HARDWARE HACKING

NICOLAS COLLINS

COMPANION WEBSITE

ROUTLEDGE

Handmade Electronic Music

Handmade Electronic Music: The Art of Hardware Hacking provides a long-needed, practical, and engaging introduction to the craft of making—as well as creatively cannibalizing—electronic circuits for artistic purposes. With a sense of adventure and no prior knowledge, the reader can subvert the intentions designed into devices such as radios and toys to discover a new sonic world. You will also learn how to make contact microphones, pickups for electromagnetic fields, oscillators, distortion boxes, mixers, and unusual signal processors cheaply and quickly. At a time when computers dominate music production, this book offers a rare glimpse into the core technology of early live electronic music, as well as more recent developments at the hands of emerging artists.

This revised and expanded third edition has been updated throughout to reflect recent developments in technology and DIY approaches. New to this edition are chapters contributed by a diverse group of practitioners, addressing the latest developments in technology and creative trends, as well as an extensive companion website that provides media examples, tutorials, and further reading. This edition features:

- Over 50 new hands-on projects.
- New chapters and features on topics including soft circuitry, video hacking, neural networks, radio transmitters, Arduino, Raspberry Pi, data hacking, printing your own circuit boards, and the international DIY community.
- A new companion website at **www.HandmadeElectronicMusic.com**, containing video tutorials, video clips, audio tracks, resource files, and additional chapters with deeper dives into technical concepts and hardware hacking scenes around the world.

With a hands-on, experimental spirit, Nicolas Collins demystifies the process of crafting your own instruments and enables musicians, composers, artists, and anyone interested in music technology to draw on the creative potential of hardware hacking.

Nicolas Collins, an active composer and performer, has worked with John Cage, Alvin Lucier, David Tudor, and many other masters of modern music. Dr. Collins is a professor in the Department of Sound at the School of the Art Institute of Chicago and Research Fellow at the Orpheus Institute in Ghent, Belgium. He has led hacking workshops around the world. He has been Visiting Artistic Director of STEIM (Amsterdam), a DAAD composer-in-residence in Berlin, and a longtime editor-in-chief of *Leonardo Music Journal*.

DISCLAIMER

Despite stringent efforts by all concerned in the publishing process, some errors or omissions in content may occur. Readers are encouraged to bring these items to our attention where they represent errors of substance. The publisher and author disclaim any liability for damages, in whole or in part, arising from information contained in this publication.

This book contains references to electrical safety that must be observed. *Do not use AC power for any projects discussed herein.*

The publisher and the author disclaim any liability for injury that may result from the use, proper or improper, of the information contained in this book. We do not guarantee that the information contained herein is complete, safe, or accurate, nor should it be considered a substitute for your good judgment and common sense.

Handmade Electronic Music

The Art of Hardware Hacking

THIRD EDITION

Written and edited by
Nicolas Collins

Illustrated by Simon Lonergan

Routledge
Taylor & Francis Group
NEW YORK AND LONDON

Third edition published 2020
by Routledge
52 Vanderbilt Avenue, New York, NY 10017

and by Routledge
2 Park Square, Milton Park, Abingdon, Oxon, OX14 4RN

Routledge is an imprint of the Taylor & Francis Group, an informa business

First edition published by Routledge 2006

Second edition published by Routledge 2009

Library of Congress Cataloging-in-Publication Data

Names: Collins, Nicolas, author. | Lonergan, Simon, illustrator.
Title: Handmade electronic music : the art of hardware hacking / written by Nicolas Collins ; illustrated by Simon Lonergan.
Description: Third edition. | New York : Routledge, 2020. | Includes bibliographical references and index.
Identifiers: LCCN 2019058409 (print) | LCCN 2019058410 (ebook) | ISBN 9780367210090 (hardback) | ISBN 9780367210106 (paperback) | ISBN 9780429264818 (ebook)
Subjects: LCSH: Electronic musical instruments—Construction.
Classification: LCC ML1092 .C66 2020 (print) | LCC ML1092 (ebook) | DDC 786.7/1923—dc23
LC record available at https://lccn.loc.gov/2019058409
LC ebook record available at https://lccn.loc.gov/2019058410

ISBN: 978-0-367-21009-0 (hbk)
ISBN: 978-0-367-21010-6 (pbk)
ISBN: 978-0-429-26481-8 (ebk)

Typeset in Bembo
by Apex CoVantage, LLC

Visit the companion website: **www.HandmadeElectronicMusic.com**

Cover image: Azucrina Record's Handmade Instruments workshop at Sesc Belenzinho 2011. Circuits made by workshop attendees. Photo © Manuel Andrade.

Praise for the Second Edition

"Nicolas Collins wants to tear apart your CD player."

WIRED magazine

"Nic Collins' book passes the torch of home-brew electronics to the next generation of musical experimentalists. Providing practical and fun recipes for sonic adventures, it simultaneously introduces the reader to the past and present field of electronic sound art."

Chris Brown, Mills College Center for Contemporary Music

"This is a terrific, unique, and much needed book; I wish I had it fifteen years ago."

Dan Trueman, Princeton Laptop Orchestra, Princeton University

"The most radical music book I've read so far this year. This jargon-free text offers a fresh alternative to the usual instruments prized by the music business."

Christopher Delaurenti, *The Stranger*, Seattle

"With wit, wisdom and enviable clarity, Nicolas Collins guides the would-be hardware hacker through the possibilities and pitfalls of playing with electricity. Those who follow his guidance assiduously will not only be able to make noise that is both personal and instilled with the virtue of self-discovery; they will also gain an education and most important of all, stay alive."

David Toop

"Nic Collins has provided an informative and gently structured doorway through which anyone can enter the limitless world of possibilities to be discovered in a raw, hands-on approach to sculpting original, electronic arts hardware. Even starting with little experience, a motivated reader can emerge with invaluable circuit building, hacking and bending skills, while also gaining an enhanced understanding of what goes on inside the boxes and behind the panels of artist-invented, electronic music devices."

David Rosenboom, Composer-Performer,
Richard Seaver Distinguished Chair in Music and Dean,
the Herb Alpert School of Music,
California Institute of the Arts

"A friendly portal into the seemingly arcane art form of circuit bending and building, rich with insights into the history and spirit of experimental electronic music. Chockfull of projects, ideas, and inspirations . . . enough to keep your neighborhood circuit bender out of trouble for years to come."

Mark Trayle

Praise for the First Edition

"Here we have, at last, an electronics book that caters to people who have ideas first, and electronics second. Collins offers a splendidly integrative look into the history of 'sound art,' basic electronics, and junk revisioning."

Meara O'Reilly, *MAKE:* magazine and makezine.com

"There are times in the history of any art form when its true visionaries set down in words, the blueprint behind an entire generation of genius. Collins has done just that with *Handmade Electronic Music*, an essential manifesto of know-how, trade secrets, and aesthetic accomplishment leaping off from Cage and Tudor and landing in today's classroom."

Thom Holmes, author of *Electronic and Experimental Music*

Contents

The Website
www.HandmadeElectronicMusic.com

1. PROJECT SUPPORT

Sharing Traces—Designing and Fabricating Your Own Printed Circuit Boards (Eduardo Rosario): Data files for Fritzing and Eagle projects
Sharing Traces—Designing and Fabricating Your Own Printed Circuit Boards With Eagle (Eduardo Rosario)
Microcontroller Sound (Joseph Kramer): Data files for Chapter 33
Microcontroller Sound (Joseph Kramer): Additional projects
Paper Circuits (Peter Blasser): Circuit board artwork for Rungling circuit in chapter 26
Sounds From Neural Networks (Wolfgang Spahn): Circuit board artwork for Confetti Neuron circuit in Chapter 31

2. TECHNICAL BOOTCAMP

Ohm's Law for Dummies: How to Understand Resistors
Switches: How to Understand Different Switches and Make Your Own
Jack, Batt, and Pack: Powering and Packaging Your Circuits
Power Supplies: Carbon Footprints From AA to EEE

3. CIRCUIT BENDING

Tickle the Clock: Finding the Clock Circuit in Toys
Hack the Clock: Changing the Clock Speed for Cool New Noises
Beyond the Pot: Photoresistors, Pressure Pads, and Other Ways to Play Your Toy
Video Music/Music Video: Translating Video Signals Into Sound, Hacking Cameras, and Extracting Sounds From Remote Controls
LCD Art: Making Animated Modern Daguerreotypes and Alternate Video Projectors

4. CULTURE AND HISTORY

Do It With Others: Hardware Hacking in South America (Florencia Curci, Alma Laprida, and Sebastián Rey)
Hacer con Otrxs: Hardware Hacking en Sudamérica (Florencia Curci, Alma Laprida, and Sebastián Rey) (Original Spanish version of Do It With Others: Hardware Hacking in South America)

5. PROJECT VIDEO TUTORIALS

6. AUDIO/VIDEO GALLERY

Foreword to the First Edition (2006)

DAVID BEHRMAN

The appearance of Nic Collins's *Handmade Electronic Music* has made me feel nostalgia for the sixties, when I was young and first heated up a soldering iron.

My generation, those of us who worked on a grassroots level with new technology in music in the sixties and seventies, felt a mixture of exhilaration and wonder as we taught ourselves about the fresh marvels made available for the first time: the transistor, a little later the integrated circuit, and then the microcomputer.

Most musicians in those days didn't make their own circuitry. They had other things to do. The few of us who did were aligning ourselves with the tinkerer-inventor tradition handed down from earlier artists who had built things, questioned the establishment, and found new sounds or tuning systems: artists like the Futurists, like Henry Cowell, Conlon Nancarrow, and Harry Partch.

Nic, the talented author of this manual, is roughly a generation away from me—he started building circuits as a way to make music in 1972. When I started around 1965—learning mostly from two artists who were friends and mentors, David Tudor and Gordon Mumma—there were no music synths for sale; when Nic started, synths existed but were out of reach unless you had a fat budget from a university or a record company.

In the sixties, I learned from Tudor and Mumma that you didn't have to have an engineering degree to build transistorized music circuits. David Tudor's amazing music was based partly on circuits he didn't even understand. He liked the sounds they made, and that was enough.

In the old days, there was no distinction between high tech and low tech. The early analog synths were made by creative individuals like Bob Moog and Don Buchla; even the early microcomputers were mostly made by garage start-ups, and there was not much difference between these and the craft shops that had made lutes, guitars, or violins for centuries. There had always been a good relationship between performing musicians and the craftspeople who made instruments—whether those were mbiras, clarinets, or gamelans. That relationship was comfortable—it was on a human scale and almost personal.

Only in recent decades have music instruments and software become corporate, mostly mass-produced and mass marketed, and only recently are the computers used for music generally the same ones found in tens of millions of business establishments.

It isn't surprising that there had to be a reaction among artists to this corporate stain, if one can put it that way, that has spread into the fabric of music.

It's been interesting for me to learn that some independent-minded young artists won't even go near a computer when they think about doing their music. Their instincts tell them to rebel against this "obedient" mode in which artists—like everyone else—are pushed into continually buying, from ever-growing corporations, the latest computer and software packages and then spending a vast number of hours learning how to use them.

There's an inescapable love-hate ambivalence about working as an artist with high-technology tools. On the one hand, computers and digital music-making devices have never been as miraculously powerful and reliable as they are today. They've also gotten much cheaper. Some software packages like Max/MSP are not really corporate products in the bad sense, and they are infinitely personable and endlessly fascinating. I'm amazed when I compare audio recording or post-production work today with the way it was in the sixties when I worked on the night shift at Columbia Records. It used to take three burly professional engineers an hour to accomplish with bulky $50,000 machines what one can do today alone with a laptop in 10 seconds.

But on the other hand, if you think about the "laptop music" style of performance that is currently in vogue, you might notice there could be a problem, even if the music sounds good, with watching a person sitting in front of a computer and operating a mouse and keyboard. It is just too depressingly similar to what hundreds of millions of workers have to do from nine to five at the office. When evening comes, and we go to the concert, we might like to experience something different, something visceral, something that is a direct result of muscular energy. We might like the relief of something zany and crazy. As Antonin Artaud said, there are plenty of people in the real world with two arms and two legs; in the theater we would like to see creatures with three.

Nic Collins's book helps us create creatures with three arms and three legs. It carries the maverick, inventor-tinkerer tradition of Harry Partch, Henry Cowell, and David Tudor into the twenty-first century. And it does so in a light, deft way—its charmingly simple, casual instructions hide the fact that its author is a sophisticated fellow who has done a lot of thinking, conversing, and music-making in the course of his travels and explorations.

Now that we're all stuck in the twenty-first century, whether we want to be or not, we have amazing new high-tech devices to work with, but we have to accept our ambivalent relationship with these products of our corporate world. From the past we have the universe of acoustic instruments as well as the tinkerer's arsenal explored in this manual. The reassuring smell of heated solder remains. The vise grip is still with us. So is the alligator clip. The good old soldering iron, the resistor and capacitor, the voltmeter, the color-coded wires—these remain. The fingers that Nic tells us how to use to coax the hidden treasures out of unknown circuit boards—they're still with us. The finger isn't obsolete. The ear isn't obsolete.

Introduction

This book teaches you how to tickle electronics. It is a guide to the creative transformation of consumer electronic technology for alternative use. We live in a cut-and-paste world: CMD-X and CMD-V give us unprecedented freedom to rearrange words, lines, photos, video, and sound to transform any old thing into our new thing with tremendous ease. But this world is a curiously disembodied—even passionless—one, a realm whose tranquil countenance is only disturbed from time to time by the discrete click of a mouse. Let's party . . . now!

In the 1970s, when fresh questions about what constituted "musical sound" and performance took hold, electronic instruments were far too expensive for anyone but rock stars or universities. Their building blocks (integrated circuits), however, were pretty cheap and *almost* understandable. A small, merry (if masochistic) band of composers and student apprentices like myself presumed to do-it-ourselves. We delved into the arcane argot of engineering magazines, scratched our heads, swapped schematics, and cobbled together homemade circuits—most of them eccentric and sloppy enough to give a trained engineer dyspepsia. These folk electronic instruments became the calling cards of a loose coalition of artists who emerged in the mid-1970s, after John Cage and David Tudor and before Oval and Moby. By the end of the decade, primitive microcomputers had begun to emerge from the primordial ooze of Silicon Valley, and many electronic composers shelved their soldering irons and started coding, but the odd circuit still popped up from time to time, adding analog spice to the increasingly digital musical meal.

I love computers, don't get me wrong, but the usual interface—an ASCII keyboard and a mouse or trackpad—is awkward and reduces performing to a pretty indirect activity, like trying to hug a baby in an incubator. Alternative controllers and sensor-laden Arduinos are steps in a good direction, but sometimes it's nice to reach out and *touch* a sound. This book lifts that baby out of the bassinet and drops her, naked and gurgling, into your waiting arms, begging to be tickled.

The focus is on sound-making performable instruments, aids to recording, and unusual noisemakers—although some projects have a strong visual component as well. No previous electronic experience is assumed, and the aim is to get you making noise as soon as possible.

We start with just clips leads, a speaker, and a battery. After learning basic soldering skills, you will build a variety of listening devices—contact mikes, coils for picking up stray electromagnetic fields, tape heads, binaural mikes—and alternative speakers. You'll transform an old radio into a synthesizer by laying damp fingers on circuit traces, and you'll make your first circuit from scratch: a simple, robust oscillator that can be controlled through a variety of means (light, touch, knobs, switches). With the confidence instilled by this delicious din, you'll proceed with circuits to amplify, distort, chop, and otherwise mangle any sound, whether electronic in origin or not: electric guitars, amplified voice, digital files, etc. You'll move on to designs for linking sound with visual material and discover some convenient "glue" circuits for putting disparate parts together for performance, recording, or connecting to computers. And you wrap up with a handful of chapters that bring in a few software tools. There are notes that direct you to sources of supplies and further resources for information. In addition, this edition includes essays, sidebars, illustrations, and recordings that place the technology in historical and aesthetic context: more than 200 hackers, musicians, artists, and inventors from around the globe are represented in the book and, in multiple media, on the website.

In selecting the specific projects for this book, I was guided by a handful of fundamental goals:

1. To keep you alive. Almost all the projects in this book are battery powered; none plug into the potentially lethal voltage running through your walls. This makes the early stages of unsupervised electronic play activity considerably safer and less daunting for the beginner.
2. To keep things simple. We work with a small number of very basic "axiomatic" circuits and concepts that can be combined with great permutational richness as you proceed and gain experience but are easy to understand and quick to get running at the beginning. The point is to make cool sounds as quickly as possible.
3. To keep things cheap. By limiting ourselves to a handful of core designs, we minimize the quantity and cost of supplies needed to complete this book. You don't need a full electronics lab, just a soldering iron, a few hand tools, and about $50 worth of parts that you can easily obtain online.
4. To keep it stupid. You will find here an absolute minimum of theory. We learn to design by ear, not by gazing at sophisticated test instruments or engineering texts. Ignorance is bliss, so enjoy it.
5. To forgive and forget. There's no right way to hack. I will try to steer you away from meltdowns, and I have included designs that are robust, forgiving of wiring errors, and accept a wide range of component substitutions if you don't have the preferred part. Most of these circuits are starting points from which you can design many variations with no further help from me—if you love a hack, let it run free.

As a result of these guidelines, this is a distinctly non-standard introduction to electronic design. Many of the typical subjects of a basic electronics course are given scant attention. After turning over the last page, you will emerge smarter, if weirder, than when you first opened the book. You will have acquired some rare skills, and ones that are exceedingly useful in the pursuit of unusual sounds. You will have gaps in your knowledge, but these gaps can be filled by a less structured stroll through resources easily available elsewhere. And everything electronic you choose to do after this book will be easy, I promise. Why? Because you will be fearless. You will have the confidence to survey those presumptuous "No user serviceable parts inside!" labels and laugh. You will be a hacker.

NOTES ON THE THIRD EDITION

The first edition of this book was published in 2006 and expanded three years later with several new chapters and a massive DVD. A decade on, I begrudgingly accepted that the contents were showing their age and in need of an overhaul and also that the domain had flourished so energetically—that so many hacks had indeed run free (and begat further free-roaming hacks)—it was beyond the capacity of one person to summarize. So I assembled an elite team of 30 hackers and writers, each with a safecracker's skill in some arcane domain. In this new edition we now have:

- Twelve new chapters with over 40 new hardware projects, from paper speakers and soft circuitry to video hacking and a radio transmitter.
- Four new chapters on software extensions of hacking techniques, including embedded computing, data hacking, and a tutorial on producing your own printed circuit boards.
- Eight new chapters on the culture and history of DIY communities on every continent except Antarctica (although we get close). I always intended *Handmade Electronic Music* to be a "shadow cultural reader," but with this edition we let the sun shine on hacking communities and individual artists across the hemispheres. These form a key component of the next item:
- The website: www.HandmadeElectronicMusic.com. This is, in effect, the paper book's companion volume. Each of the previous editions included media on the verge of extinction: an audio CD in the first, a DVD in the second. While not promising an *infinite* future, this website will be easier to access and easier to update. In addition to all the media from the earlier editions (20 artist's audio tracks, 87 one-minute artist's videos, and 13 video tutorials), this is where you'll find all but one of the new chapters on culture and history. We've also added more artist documentation, data files for several new technical chapters, and other bonus material, including the original Spanish and Japanese texts of chapters translated into English for publication.

  I cannot stress enough the value of the material on the website—the audio and video files put your soldering in a global artistic perspective, the essays will give that global perspective a social context, and the data files will save you a lot of typing. You will be much poorer for not bookmarking the site. And to help you on the way, every reference to the website is indicated in the book by a clever little symbol:

One fact—hammered into me by my long-suffering editors—is that to make room for all this *new*, somethings *old* had to depart, else we would have to distribute each copy of the book with its own plinth. Happily, the website provides the literary equivalent of a capacious attic: if you don't have a well-thumbed copy of the second edition lying around, you can go to the site to find almost everything that's been excluded from the new book—most notably the original chapters on circuit bending, which have been superseded by a new text in the current edition (modern toys are increasingly unhackable, and there are other accessible sources for learning basic techniques), and several chapters on useful but boring technical details (switches, batteries, Ohm's law, etc.), have been relocated to a Technical Bootcamp section of the website.

Seventeen years after succumbing to pressure from students to present a class in what I regarded as an idiosyncratic and somewhat anachronistic pastime of dubious general appeal, I find that the book continues to be popular with a surprisingly diverse group of people. And while I would have been content to let it self-compost over time (hopefully providing nutrients for new growth, tended by the hands of others), my patient editors and energetic colleagues have long argued that refreshing the original with new seeds would be appreciated. I am, naturally, flattered, but I have long felt like a medium in a hacking séance, passing on to others the knowledge my mentors and colleagues have so generously offered. For this third edition, I am especially indebted to the extraordinary new contributors. They all had better things to do than answer my call, but, having done so, they have made this edition of *Handmade Electronic Music* at least 30 times more interesting than I could have done on my own.

PART 1

Starting

CHAPTER 1

Getting Started
Tools and Materials Needed

You will need certain tools and materials to undertake the projects in this book. I have kept your investment to an absolute minimum—none of the fancy test equipment and drawers full of teeny parts found in a typical electronics lab; a few basic hand tools and a modest collection of easily obtainable components will see you through. Each chapter begins with a list of the specific parts needed to build the projects, but you'll need some general supplies as well.

LISTENING

Whereas "proper" electronic engineering is typically taught with *visual* reinforcement—staring at an oscilloscope, meters, or a computer screen—we will work by *ear*, as befits the development of sonic circuitry. A *monitor amplifier* thus becomes your primary tool. Several of our experiments entail touching electronic circuitry with damp fingers, and those fingers should be kept far, far away from the 120 or 240 volts streaming into any device with a power cord that attaches to the wall. We also need a fair amount of gain at the input to our amplifier at the beginning of this book, where we make a variety of microphones that have pretty low output levels. A typical pair of battery-powered speakers intended for amplifying a computer or phone does not provide enough boost for these microphone projects, although they could come in very handy when we move on to building our own circuits. Instead, consider acquiring one of those wee bitty guitar amps made by Fender, Marshall, Dan Electro, etc.—they look like the guitarist's equivalent of a shrunken head (Figure 1.1).

If you are feeling adventurous, an economical and flexible solution is to buy a low-power (< 1 watt) amplifier kit from any of a number of online retailers. These kits include all components, a tidy little printed circuit board, and instructions on where to place which part (Figure 1.2). This is an excellent way to bootstrap your soldering skills while saving some money. Besides the financial and pedagogic advantages of building your own tool, you can connect to these amplifiers using clip leads instead of patchcords, so it's faster and cheaper to test out your projects. The Altoids tin (which will reappear throughout this book with comet-like regularity) makes a very practical

Figure 1.1
Some battery-powered mini-amplifiers.

housing for a small circuit board and a 9-volt battery. Or you can pack the circuit board and a speaker into some kind of mini faux guitar amplifier and begin your hacking career by dazzling your friends with your design aesthetic. A few words of caution: certain amplifier chips are more forgiving of operator errors; the new generation of class D amplifier chips and kits are cheap and powerful but easily damaged by shorting the speaker connections (in Chapter 24 we build an amplifier using one of the most robust chip options).

For some of the projects we'll need a second amplifier, and this one can be AC powered without risk. You have a few options:

- A classic hi-fi stereo amplifier, like your parents used to own. It needn't be very powerful (10–30 watts) or high quality. Look around the local thrift shops or flea markets or on eBay for a used one. It should have connections on the back for external speakers, rather than built-in speakers like the mini guitar amps.

Figure 1.2 A low-power amplifier kit: assembled circuit board (left) and mounted in an Altoids tin (right).

- At the time of writing, there are a large number of small, inexpensive stereo amplifiers on the market. Employing the aforementioned class D technology, these amps are efficient enough to run off a small wall wart power supply. Find one with 10–30 watts per channel and make sure it has connections for external speakers (Figure 1.3).
- Alternatively, you can build your own amplifier using a slightly more powerful version of the amplifier kits described prior (Figure 1.4).

Many of the things we will build produce a very wide range of frequencies. Some of these frequencies are more delicious than others. A cheap graphic equalizer footpedal, such as sold for guitarists, is an excellent tool for separating the yolks from the whites, sonically speaking. Numerous companies make them, and for our purposes they all sound more or less the same—buy the cheapest one you can find.

If you already own a small mixer and a pair of monitor speakers, they will prove useful throughout the book, but there's no need to run out and buy them at this point.

Figure 1.3
Typical low-power class D stereo amplifier.

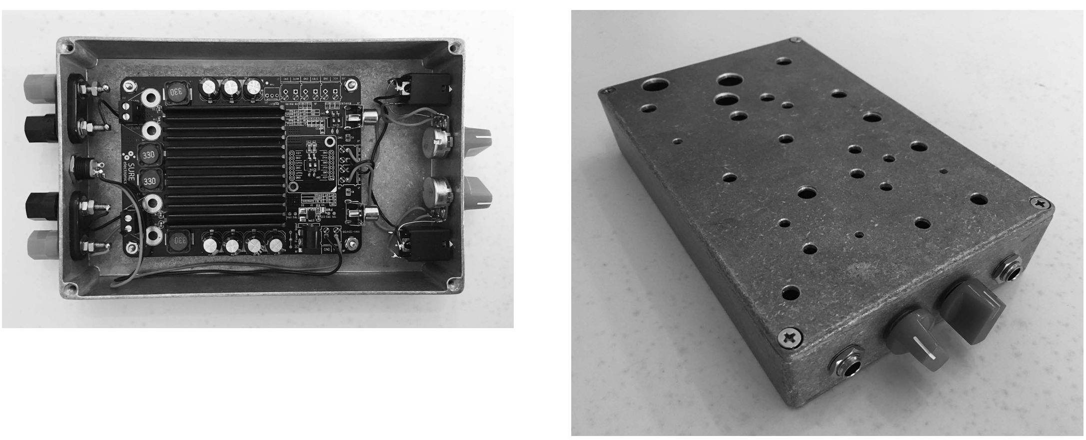


Figure 1.4 25-watt amplifier circuit board by Sure (left), housed in case with ventilation holes corresponding to depth soundings of Buzzards Bay (coastal Massachusetts) (right).

TOOLS

You'll need some basic hand tools (Figure 1.5). Many might already be in your collection if you've ever had to change a washer, wire up a lamp, or serve in the Swiss Army. None are expensive—the only place you might want to splurge a little is on a better-than-terrible soldering iron.

- A soldering iron, with a very fine point, 25–60 watts. Not a soldering gun or anything from the plumber's section of the hardware store. Don't get a cheap iron—it makes it very frustrating to learn soldering. Weller makes good ones that are reasonably priced and have replaceable, interchangeable tips.
- Solder—fine, rosin core—not "acid-core" solder, that's also for plumbers.
- Diagonal cutters, small, for cutting wire and component leads down by the circuit board.
- Wire strippers (unless you have the perfect gap between your front teeth)—simple, adjustable manual kind for light-gauge wire.
- A set of jeweler's screwdrivers (flat and Phillips)—for opening toys with tiny screws.
- A Swiss Army knife.
- A pair of scissors.
- A cheap digital multimeter, capable of reading resistance, voltage, and current.

Figure 1.5 Some handy tools.

- Plastic electrical tape.
- Mini jumper cables with small alligator clips at each end, at least 20 of them—you can never have too many.
- A Sharpie-style fine-tipped permanent marker.
- Some small spring clamps or clothespins.
- A small vise or "third hand" device for holding things while you solder them.
- Basic shop tools—such as a small saw for metal and plastic, files, and an electric drill—are useful when you start to work on packaging.

PARTS

As mentioned earlier, at the head of each chapter you'll find a list of the specific components needed to complete the projects covered (if the list is not self-explanatory, just read on, since each new item is discussed as it comes into use). But here are some of the supplies you'll need throughout the book. These can readily be obtained from a wide range of online retailers (see Appendix A):

- Lightweight insulated hookup wire, 22–24 gauge, one roll stranded, one roll solid.
- Lightweight shielded audio hookup cable, single conductor plus shield.
- Assortment of standard value resistors, 1/8 or 1/4 watt. Sets are easily and inexpensively available from online retailers. If you want to make the minimum investment, the critical values we use are: 100 Ohm, 1 kOhm, 2.2 kOhm, 10 kOhm, 100 kOhm, and 1 mOhm.
- Assortment of capacitors, in the range of 10 pf to 0.1 uf monolithic ceramic or metal film, and 1 uf to 47 uf electrolytic. These can also be bought in sets, but since they are a little more expensive than resistors, you might prefer to purchase a handful of each of a few different values from across the full range and then replace or supplement them as needed. The most commonly used values in our projects are: 0.01 uf, 0.1 uf, 1.0 uf, 2.2 uf, 4.7 uf, and 10 uf.
- 9-volt battery clips—the things that snap onto the nipples at the end of a battery and terminate in lengths of wire. Get five or more.
- Assorted audio jacks and plugs to mate with other devices, such as the headphone jack on your phone or the input to your amplifier.

BATTERIES

Because of our core philosophy of avoiding unnecessary electrocution, we will be working exclusively with battery-powered devices. This means we will need a lot of batteries, for your amplifier, toys, radio, and the circuits you make. Please be *milieu vriendelijk* (a friend of the environment), as the Dutch say, and invest in some rechargeable ones if at all possible. The world's groundwater will thank you.

ARCHITECTURE

You'll need a clean, well-lighted place. It should be well ventilated—soldering throws up some unpleasant fumes. You'll want a fair amount of table space since hacking has an unfortunate tendency to sprawl (Figure 1.6). The table surface can be damaged by soldering, drilling, and filing, so no Boule inlay please. You need electrical power at the table for your soldering iron and a good strong desk light.

OK, are you feeling ready to hack? First, a few rules to live by . . .

Figure 1.6 A typical worktable, before and during hacking.

CHAPTER 2

The Seven Basic Rules of Hacking
General Advice

Like boot camp or Candyland, this book is almost devoid of theory but heavy on rules. Here are a few guidelines for keeping you healthy and happy as you hack:

Rule #1: Fear not!
Ignorance *is* bliss, anything worth doing is worth doing *wrong*, and two wrongs *can* make a right.

Rule #2: Don't take apart anything that plugs directly into the wall.
We will work almost exclusively with battery-powered circuitry. AC-powered things can kill you. AC adapters (wall warts) may be used *only* after you have displayed proper understanding of the difference between insulation and electrocution.

Rule #3: It is easier to take something apart than put it back together.
Objects taken apart are unlikely to function normally after they are put back together, no matter how careful you are. Consider replacement cost before you open.

Rule #4: Make notes of what you are doing as you go along, not after.
Most wires look pretty much alike. As you take things apart, make notes on which color wire goes to where on the circuit board, to what jack, etc.—a cell-phone photo helps but is sometimes not sufficient by itself. Especially important are the wires that go to the battery. Likewise, note what you add as you add it and what you change as you change it.

Rule #5: Avoid connecting the battery backwards.
This can damage a circuit.

Rule #6: Many hacks are like butterflies: beautiful but short-lived.
Many hacks you perform, especially early in your career, may destroy the circuit eventually. Accept this. If it sounds great, record it as soon as possible, and make note of what you've done to the circuit so you can try to recreate it later (see Rule #4).

Rule #7: In general, try to avoid short circuits.

Try to avoid making random connections between locations on a circuit board using plain wire or a screwdriver blade. This can damage a circuit—not *always*, but inevitably at the most inconvenient time.

Additional rules will be introduced throughout the book and are summarized in Appendix B.

PART 2

Listening

CHAPTER 3

The Victorian Synthesizer
Twitching Loudspeakers

You will need:

- A few dispensable raw loudspeakers of any size, but bigger is better (> 4"/10 cm). You can salvage them from old TVs, unwanted boom boxes, thrift shop hi-fis, etc.
- A 9-volt battery.[1]
- Some jumper leads with alligator clips.
- A metal file.
- A sheet of copper, steel, or iron or a chunk of some conductive metal, the more corroded or scratched the better.
- Pop tabs from soda cans, paper clips, loose change, nuts and bolts, assorted scrap metal.
- Plastic or metal bowls/cans, approximately the size of your speaker.
- Optional: a drum of some sort, a cymbal.

British artist John Bowers has developed a beautiful electric instrument, evoking the spirit of nineteenth-century electrical experimentation (think twitching frogs legs and early telephones) out of nothing more than a speaker, some batteries, wire, and scrap metal.[2] Requiring no special electronic skills, this project serves as an excellent portal into the world of hacking.

Hook up the circuit shown in Figure 3.1. Clip one end of a test lead to one terminal of the speaker (it doesn't matter which). Clip the other end to one terminal of the battery (again, it doesn't matter which one). Now tap the loose end of the second clip lead to the open terminal of the battery (momentary contact only, don't clip on). The speaker should pop in or out from its position of repose. Reverse which clip goes to which terminal of the battery to change an inny to an outy or vice versa: the cone that jumped *out* before should now suck *in*.

The cone doesn't twitch as it should? Check that the battery is not dead: measure its voltage with a multimeter if you have one, put it in a functional toy or effect pedal and see if it works there, or use the admittedly icky but nonetheless effective expedient of touching both battery terminals (briefly!) with the tip of your tongue—a good battery will electrically tickle your taste buds, inducing a curious, salty sensation residing in that fuzzy region between pain and pleasure.

Figure 3.1 Twitching a speaker.

Battery OK? Then try another set of clip leads. I've bought mine from a variety of sources, and every batch seems to have about a 10% failure rate. A good worker never blames her tools, unless they end in alligator clips (or crocodile clips, per our British hackers).

What's happening? Passing the battery current through the speaker coil creates an electromagnetic field that interacts with the permanent magnet affixed to the speaker's metal frame; the coil is attached to the paper or plastic cone and moves it in or out, depending on the polarity of the battery and orientation of the fixed magnet (think of all the games you played with magnets as a kid).

Tapping the alligator clip against the battery terminal will produce a nice little percussive accent, both drum-like and "speaker-ish," acoustic and electronic. But that's just the start. Keep one lead connected between the battery terminal and the speaker terminal as before. But this time, instead of connecting the second lead directly from the battery to the other speaker terminal, clip it between the speaker and a metal file or a chunk of some conductive metal: a pie tin, a scrap copper flashing, a guitar string, etc.—the rougher or more corroded the metal surface, the better. Clip one end of a third jumper lead to the other terminal of the battery and the other end to a nail, bent paper clip, knife, or some other pointy piece of conductive metal (Figure 3.2).

Touch the nail to the metal. When it contacts the metal, the nail completes the circuit, sending current through the speaker coil and making the cone jump, as before. Now scrape the nail across the metal: as the contact is broken by the irregularity of the surface, the speaker emits scratchy, percussive sounds whose character is quasi-controllable through hand movement. Drawing the nail across a file elicits sounds reminiscent of turntable scratching.

Figure 3.2 Scratching a speaker.

You may notice small sparks as the contact is made and broken (fun!), and the battery will probably get warm—the speaker coil is almost a short circuit, and it sucks a lot current from the battery. Avoid holding the nail on the metal for an extended period of time—loudspeakers get hot and bothered when presented with a steady voltage, so it's better to send them shorter pulses. Don't try this with your roommate's expensive speakers (and don't plug any speaker directly into the wall!).

Instead of scraping across metal by hand, you can rest the clips inside the speaker cone while you hold the two leads loosely near the clips. The cone jumps when contact is made, breaking the contact for a moment, then the clips fall against each other and the process starts all over—an electromechanical oscillator and the beginning of what Bowers calls "The Victorian Synthesizer" (Figure 3.3 and his audio track on the website). By holding the two contacts close together against the speaker cone as you vary your touch and the location on the cone, you can change the pitch and rhythm of the buzzing sounds. Clip your leads to two paper clips, washers, coins, aluminum pop tabs, etc. and let these metal bits bounce against each other—the weight of the scraps will affect the pitch of the oscillation.

Fill the cone with spare change, nuts and bolts, or other bits of metal and drop your leads into the mess: the mechanical interaction of all the conductive bits creates a kind of pre-computer, analog algorithmic music. You can also line the cone with aluminum foil or apply metal tape (such as the kind sold in hardware stores for preparing windows for home burglar alarms) and connect one lead to the foil or tape and the other to some metal fragment. The tab gets thrown up from the foil or tape, breaking and making contact as before.

Figure 3.3 The Victorian Oscillator.

Sound doesn't always *end* at the loudspeaker; it can start there—listen to the jingle of the scrap metal in our algorithmic exercise. Add gravel or dried lentils inside the cone for additional rhythmic accents. You can use a bowl, a can, a toilet plunger, or even your hands to mute and resonate the sound further. Place a can on the cone, open end down; clip one lead to the can and one to a metal washer placed on top of the can (Figure 3.4). The speaker cone will jump, breaking and remaking the contact as before, but in addition, as the can jiggles, it changes resonance like a trumpet mute. Put some jangly things inside a small glass bottle/vial and place it inside a cone—*maracas de cristal*. Rest the speaker on a snare drum. Watch Zoot Houston's video on the website to see a cymbal being played by a jumping speaker.

You'll notice that different speakers sound different, even in similar configurations. It's mostly a function of size, as with drums, but if you try these experiments with a speaker in an enclosure (such as one from a home stereo), you'll hear that it has considerably more bass presence—the box gives a woofer its woof. Multiple speakers can be wired in series, with contacts resting in each cone, so they interact to produce more complex rhythms.

You can further extend the sound world of the jumping speaker by placing an inductive pickup (Chapter 6) or contact mike (Chapter 7) in the cone and connecting it to an amplifier. The amplified sound will change as mikes bounce around inside the cone.

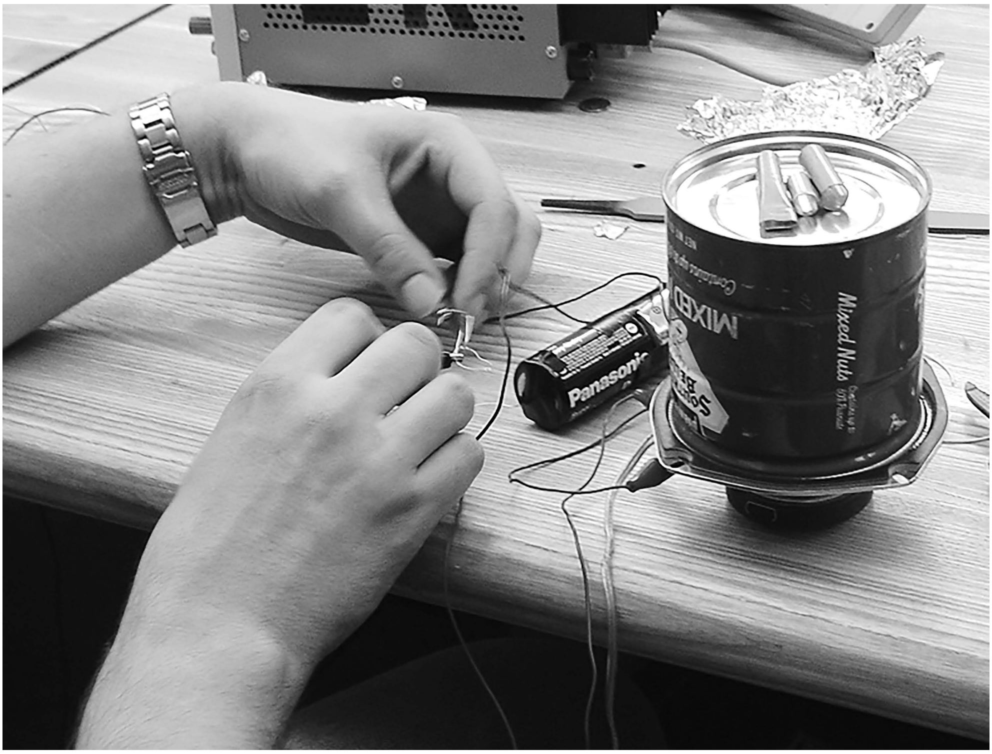


Figure 3.4 A "prepared speaker."

Finally, there's a visual element. You can fill the speaker cone with talcum powder or light sand and watch it make patterns as the cone jumps. For a touch of the old Fillmore light show, waterproof the speaker cone with paint, fill it with water or oil, and turn down the lights; reflect a flashlight or laser pointer off the surface and watch the resulting patterns on the wall or ceiling.

NOTE

1. Despite my advocacy of rechargeable batteries, this is the one project in the book that works better with old fashioned alkaline batteries, for reasons beyond my understanding.
2. John Bowers and Vanessa Yaremchuk, "The Priority of the Component, or in Praise of Capricious Circuitry." *Leonardo Music Journal* Vol. 17 (2007). P. 39.

CHAPTER 4

In/Out

Speaker as Microphone, Microphone as Speaker, the Symmetry of It All

You will need:

- A battery-powered amplifier.
- An additional "raw" loudspeaker (not in a speaker cabinet).
- A pair of headphones or earbuds.
- A small DC motor.
- A pair of jumper leads with alligator clips and a plug to fit the input jack of your amplifier.
- Optional: a dynamic microphone (i.e., Shure SM58).

ELECTROMAGNETISM

There is a beautiful symmetry to the electrical principle commonly used to translate acoustic sound into an electrical signal and back into sound again. In the previous chapter, we made sound by sending electrical current through the wire coil inside a speaker, and this process is reversible. Inside every dynamic microphone (such as a typical PA mike) is a lightweight plastic membrane affixed to a coil of fine wire encircling a cylindrical magnet. Beyoncé sings, and her sound waves jiggle the membrane, which moves the coil in the field of the magnet, generating a very small electrical current. This current is amplified, equalized, flanged, reverberated, compressed, (maybe) Auto-Tuned, and finally amplified even more before being sent back out to a bigger coil wrapped around an even bigger magnet. Now this shimmering electromagnetic field pushes and pulls against the big magnet, moving a paper cone back and forth, producing sound waves of a louder, possibly improved, Beyoncé.

A record player cartridge is basically a microphone with a needle where the diaphragm should be, and record cutting heads are beefy backwards phonograph cartridges. Headphones are tiny speakers. You get the picture?

Not only is the same electromagnetic force used for both input and output devices (microphones and speakers), but sometimes the gizmos themselves are interchangeable. Plug a pair of headphones or earbuds into the input jack on your amplifier; speak into them and listen (many a band's demo tape was recorded through headphones). Use clip leads and a plug to connect a raw loudspeaker to the input of your amp. Motown engineers recorded kick drum and bass guitar with large speakers placed in front of the drumhead and bass cabinet—essentially "subwoofer microphones." (The Beatles borrowed this technique when recording *Paperback Writer* in 1966—compare the bass sound on this track to that on *I Want to Hold Your Hand* from three years earlier.)

Backwards speakers don't sound as generically "good" as a $5,000 Neumann tube mike, but for special applications they can be very effective (as Motown's sales attest). Non-standard microphones like these "pre-produce" the sound of whatever you are recording or amplifying by introducing idiosyncratic equalization or distortion that you might otherwise add later in the mixdown process—if you like the effect, why not get it over with at the start of your session and avoid the trouble of approximating it further down the line? A *flat* sound is not always the *best* sound, and delaying the fixing until the mixing is sometimes a sign of procrastination rather than perfectionism.

Likewise, any dynamic microphone (i.e., based on a coil and magnet design, such as the ubiquitous Shure SM58, spat upon by singers in clubs around the world) can be used as a low-level speaker or headphone. Use whatever chain of adaptors is needed to plug a dynamic mike into the headphone jack of your amplifier. Patch some audio source into the amp input (phone, sound file on your laptop) and hold the mike up to your ear. Slowly turn up the volume until you can hear the music. In a pinch, many a PA engineer has substituted a mike for missing headphones when tracing a suspected fault through a mixing board (although watching an engineer seemingly amplifying his earlobe can be disconcerting for nearby audience members).

Microphones have very delicate coil windings, however, and can be easily blown out, so BE CAREFUL. Also, "condenser mikes" (both phantom-powered studio microphones and "plug-in-power" mikes for video and audio recorders) use a different principle of translation, not so easily reversible, so:

> **IF THE MIKE USES A BATTERY OR PHANTOM POWER OR IS REALLY, REALLY EXPENSIVE, DON'T USE IT BACKWARDS.**

Which brings us (occasionally pyrotechnically) to our next Rule of Hacking:

Rule #8: In electronics some things are reversible with interesting results, but some things are reversible only with irreversible results.

Some of you may recognize that the Eighth Rule of Hacking is a sobering exception to the First Law of the Avant-Garde:

Do it backwards.

TELHARMONIUM LITE

A related experiment will introduce you to the fundamental operating principle of what is generally accepted to be the first music synthesizer: the Telharmonium. Patented by Thaddeus Cahill in 1897, the Telharmonium weighed in at over 200 tons and resembled a power station on a railroad flatcar more than a musical instrument. It generated sine tones by spinning the shafts of dynamos to produce AC currents, like that running through your household wiring, but with variable frequency instead of a fixed 60 or 50 Hz. The resulting music was available by subscription (the roots of Spotify!) over the wires of the recently installed telephone grid until it was banished from the network for overpowering conversations. We can mimic the Telharmonium effect, on a more modest scale, by clipping the terminals of a small DC motor to a plug connected to the input of an amplifier (Figure 4.1). Turn on, turn up, spin the shaft, and you should hear a whirring sound whose pitch is proportional to the speed of the shaft.

If you have a dead hard drive in your desk drawer, open it up. When you get over your awe of the Rolex-caliber engineering inside, remove the motor from the surrounding circuitry and metalwork. Connect clip leads between the motor terminals (usually not too hard to find) and your amplifier. Spin the platter. You should hear a smooth whir that descends in pitch as the disk slows down. Mellow.

Artist Lorin Edwin Parker has built an off-the-grid steam-powered synthesizer by connecting a DC motor to a homemade steam engine (Figure 4.2 and his video on the website).

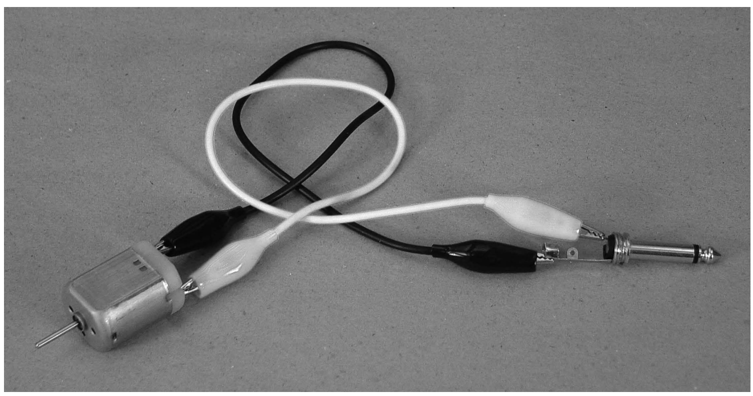


Figure 4.1 Motor-as-oscillator: listen as you spin the shaft.

Figure 4.2 Lorin Edwin Parker's steam-powered synthesizer.

CHAPTER 5

How to Solder
An Essential Skill

You will need:

- A soldering iron with a fine tip.
- A small damp sponge (or, in a pinch, a folded wet paper towel).
- Rosin-core solder.
- Diagonal wire cutters.
- Wire strippers.
- Some light-gauge insulated wire, solid or stranded.
- An audio jack or plug of some kind.

Soldering is one of the fundamental skills of hardware hacking. It is almost impossible to hack hardware *without* knowing how to solder. We've had fun with our clip leads, and some hackers make it a point of pride to use them exclusively, eschewing solder for the transient glory of an alligator's jaws.[1] But the day will come when these miniscule members of the order *crocodilia* will fail you: plugging a Stratocaster into a Marshall stack, lowering a hydrophone into the Mariana Trench, or building a sequencer. As a skill, soldering commands a lower hourly wage than Java or C++, but your friends and parents might be impressed by your acquisition of such arcane knowledge (as if you had learned fire-eating or Linear B).

Successful soldering, like fundamentalist Christian comedy performed in mid-winter by an L-Dopa patient, depends on cleanliness, heat, steady hands, and . . . timing!

Despite appearances, soldering is not just a question of dropping melted solder onto the two things to be joined. The strongest bond results from first melting a thin layer of solder onto each surface ("tinning"), then letting them cuddle up to one another while you heat both surfaces to remelt the solder until it commingles. The process is similar to gluing wood: it's better to saturate the surface of each piece of wood with a layer of glue before assembly, rather than just squeezing out a blob and slapping them together.

We'll start by soldering wires together—high-temperature knitting. Not very exciting, but a cheap way to learn (and much easier than trying to solder old license

Figure 5.1
Not the way to start soldering.

plates, which is how my father and I tried—and failed—to learn when I was 10—see Figure 5.1).

1. Plug in the iron and place it somewhere where the tip will not touch flammable, meltable, or scorchable surfaces or its own power cord (a cute little metal rest often comes with the iron). Wait (a long time) for it to warm up. The iron is hot enough to use when solder touched to the tip melts.
2. Wipe the tip of the hot iron across a damp sponge. The tip must be smooth and clean enough that the solder flows evenly, leaving a shiny silver coating. If blobs of solder fall off, or the tip remains gray and crusty even after sponging, unplug the iron; after it has cooled down, polish the tip with steel wool, fine sandpaper, or a file and try again (Figure 5.2). If the tip of the iron is seriously pitted, you will need to replace the tip (or, if it is a cheap iron with non-replaceable tip, the whole iron).
3. Use your wire strippers to remove about 1 inch (25 mm) of insulation from the ends of two pieces of wire. There are several styles of wire stripper, differing in cost and complexity of mechanism—I prefer the cheapest kind, which resembles pliers with an orthodontic problem (Figure 5.3). Use the adjustment on the strippers, or a fine sense of touch, to avoid cutting

Figure 5.2
A happy soldering iron (top) and a sad soldering iron (bottom).

Figure 5.3 Simple wire strippers.

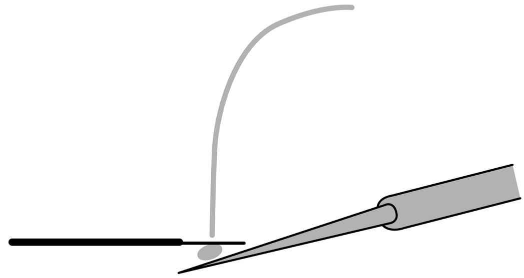


Figure 5.4 Tinning a wire.

through the wire as you nick and slide back the insulation—connect with your inner female dog carrying her puppies. If the wire is stranded, twist the strands to eliminate frizzing. Hold the wires in something so that the tips are up in the air but don't wiggle around too much. You can use a fancy "third hand" gizmo (two articulated arms with alligator clips, affixed to a metal base), or a vise, or just weigh down the coil of wire under this book or invite over a friend (with nerves of steel and fingers of asbestos).

4. "Tin" the wires. Melt a small blob of solder on the tip of the iron. Hold this blob under one of the wires—this blob conducts the heat from the iron to the wire. Hold the tip of the solder roll against the *wire*, not the iron. After a few seconds, the wire should be hot enough that the solder will melt, flowing around the wire to coat it evenly in a smooth layer; if not, apply a *tiny* bit more solder to the tip of the iron and try again (Figure 5.4). Remove the iron from the wire. The solder should cool to a smooth, shiny silver; if it is rough and gray, you did not get the wire hot enough—try again. Then tin the second wire.
5. Twist the wires around one another like strands in rope. Again, apply a small blob to the iron and use the blob to conduct heat to the bundled wires. After a few seconds, the tinned solder should remelt and flow together; you can apply a little bit more solder to strengthen the joint, but only as much as can flow and distribute itself smoothly—like a wax-impregnated candle wick. Wait several seconds *without moving* for the joint to cool and harden. Blobs of solder on the wire or dull gray solder are signs of a "cold solder joint" (Figure 5.5)—such a joint is neither electrically nor mechanically reliable. Try again.

When tinning and soldering, apply heat for the minimum amount of time needed to get the solder to flow, otherwise you may damage the parts you are soldering (for example, melting the insulation off the wire).

6. Repeat this process until you get strong bonds and you feel comfortable with the "touch" of soldering—how much heat and solder to apply for how long.


Figure 5.5 A happy solder joint (left) and a sad solder joint (right).

7. Now try soldering wires to a plug or jack. Tin the wire and jack terminals as before, then solder them together by holding the wire against the terminal as you heat both with the soldering iron. If the terminal lugs on the jack have wire-sized holes, you can make your life easier by bypassing the tinning and simply looping the end of the wire through the hole to secure it before soldering.

When you start soldering circuit boards, it's important to use as fine a tip as possible—the landscape gets crowded. Keep the tip cleaned and tinned by frequent swipes across the sponge. Use solder sparingly to avoid blobs of excess solder unintentionally connecting separate pads on the circuit board (what are known as "solder bridges").

Be advised that cold solder joints sometimes sort of work, for a while, but will come back to haunt you at the most inauspicious moment, so it's worth getting soldering right before going on stage.

NOTE

1. Phil Archer, "Clip Art." *Leonardo Music Journal* Vol. 17 (2007). Pp. 29–30.

CHAPTER 6

Circuit Sniffing
Eavesdropping on Hidden Magnetic Music

You will need:

- A battery-powered AM radio or two.
- A battery-powered amplifier.
- A telephone pickup coil, an electric guitar pickup, an inductor (c. 100 mH), a relay or solenoid, or a wall wart.
- Optional: 100 feet of light-gauge insulated wire, an audio plug, and two pieces of wood, approximately 1 inch × 2 inches × 5 feet.

RADIOS

Radios make the inaudible audible. Unlike ordinary microphones and amplifiers, which make very quiet *acoustic* sounds much louder, radios pick up *electromagnetic* waves that have no acoustic presence whatsoever and translate them into electronic signals that can be amplified and heard through a loudspeaker. Radios are manufactured for listening to intentionally transmitted electromagnetic waves (i.e., those sent from radio stations), from which they extract music and speech through a process of *demodulation*—multiple stages of amplification, filtering, and frequency shifting. But radios can also be used to sniff out other types of electromagnetic signals, such as those emitted by lightning, sunspots, Aurora Borealis, meteorites, subway trains, security gates in stores, and a gaggle of household appliances. Generally speaking, AM radios do a better job of picking up these "spurious" noises than FM radios. (In fact, the invention of FM technology was celebrated as the triumph of signal over noise thanks to its superior rejection of exactly the kind of weird stuff we want to hear here.)

Put batteries in your radio and turn it on. If it has a *band select switch*, set it to AM. Tune it to a dead spot at either end of the dial and turn it up loud. Place the radio near various electrical appliances: fluorescent lights, electric motors, computers, cell phones, CD players, and infrared remote controls are especially noisy. Fire off a camera flash. Experiment with tuning the radio to gaps between stations, as well as the dead bands at either end.

As the FCC often warns you, certain electrical appliances can cause "radio interference." What this means is that, as a byproduct of whatever useful thing the appliance is doing, it emits lots of spurious electromagnetic radiation in the same frequency region as radio and TV broadcasts. The radio filters specific frequency ranges of electromagnetic waves, shifts the signal down into the range of our hearing, demodulates the music or speech, and amplifies the result. Compared to radio stations, your appliances put out very weak signals. The noise from a computer drops off rapidly as you move the radio a few feet away since electromagnetic waves follow what physicists have so elegantly dubbed "the inverse square law": their strength fades by the square of the distance from the source.

If your radio has an FM band, try it as well. Although FM radio is designed to minimize interference, strong periodic signals (like the clock frequency of a computer) can sometimes be tuned in, just like a proper radio station.

COILS

An alternate approach to picking up electromagnetic signals is to use a simple coil of wire and an amplifier. A *telephone pickup* consists of meters of thin copper wire wrapped around an iron slug (Figure 6.1). Plugged into an amp, this coil acts like a radio antenna for frequencies low enough to be within the range of our hearing without the need for demodulation (i.e., between c. 20 Hz—20 kHz). Stuck on a telephone earpiece (or held against any other loudspeaker), it picks up the electromagnetic field generated by the voice coil of the speaker, allowing you to record your landlord or ex-lover making unsavory threats.

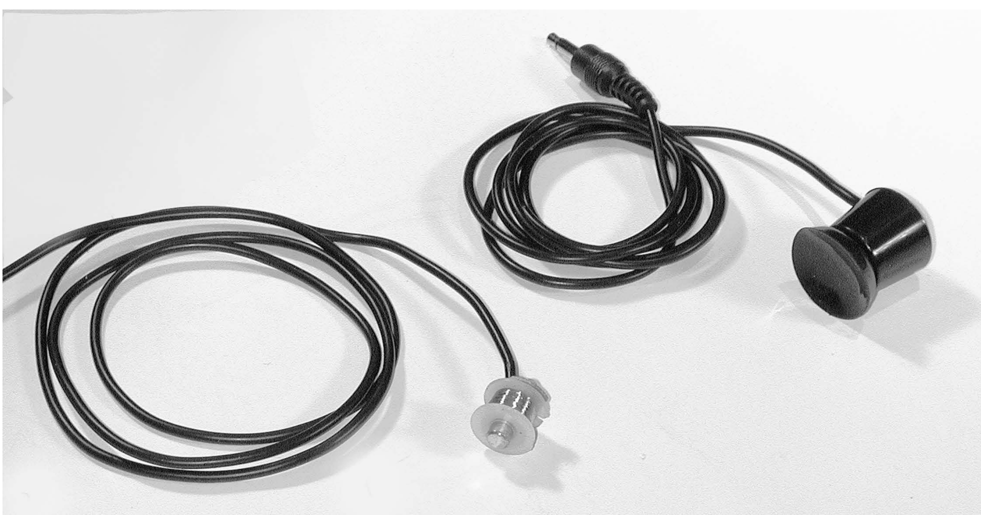


Figure 6.1 A telephone coil pickup, showing internal coil construction (left) and packaging (right). From the collection of Michael V. Hayden.

ART & MUSIC 1

MORTAL COILS

In addition to constituting the basic mechanism of radios, microphones, and speakers, electromagnetic fields have spookier aspects that have been central to instrument design and artists' works for nigh on 100 years. The siren song of the Theremin (the earliest commercial electronic instrument, invented by Leon Theremin in 1924) resulted from two high-frequency radio signals beating like out-of-tune strings on a piano. Seventy years later, Gert-Jan Prins (Netherlands) (Figure 6.2) and David First (US) (see his audio and video on the website) created music out of Theremin-like interference and feedback between radio receivers and transmitters. Some of the earliest realizations of computer music were heard through radios placed on top of the central processing units of mainframes: engineers would run programs with looped instruction cycles whose lengths were calculated to emit a composed sequence of radio frequencies, which were duly demodulated by the radios.

Alvin Lucier's (US) *Sferics* (1980) is a recording of electromagnetic "tweaks," "bonks," and "swishes" originating in the ionosphere, the result of self-immolating meteorites, the dawn chorus, and the Aurora Borealis.[1] More recently, Lauren Carter (US) and Joe Grimm (US), in a Ben Franklin moment, sent a wire-wrapped kite up in the sky to record the same atmospheric sounds (see their video on the website). The squeal and chatter of mistuned shortwave receivers has been an inspiration to composers from Karlheinz Stockhausen (Germany), whose 1968 composition *Kurzwellen* used four receivers in live performance,[2] to Disinformation (UK), who has made a career of recording and performing with radio signals from across the spectrum, emanating from both human activities (power stations, navigation satellites, submarine communication, camera flashes) and natural sources (sun spots, thunderstorms). Australian artist Joyce Hinterding has worked with enormous coils both in gallery settings and in the wild (see Caleb Kelly and Pia van Gelder's chapter on Australian music on the website for more information). Telephone taps have been used

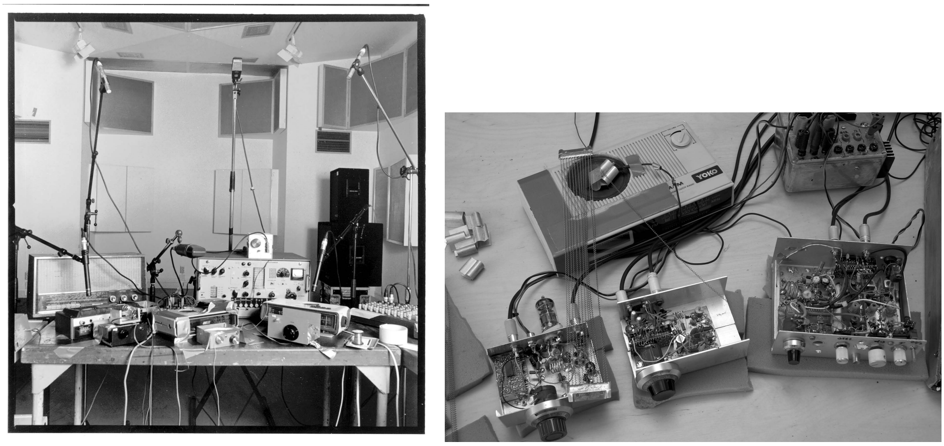


Figure 6.2 Gert-Jan Prins, installation for *Sub V* at STEIM, Amsterdam, The Netherlands, March 1996 (left), and detail of his radio transmitter and receiver feedback system (right).

Figure 6.3 Christina Kubisch, *Electrical Walks*, Birmingham, UK, 2006.

to pick up stray radio emissions from laptops, CD players, subway trains, and a host of other unexpected objects by artists such as Nathan Davis (US), Haco (Japan), Andy Keep (UK) (see audio on website), Jérôme Noetinger (France), Rob Mullender (UK), and Sonia Yoon (US).

Toward the end of the 1970s, German sound artist Christina Kubisch began using electromagnetic induction to transmit local sound fields that followed wires she arranged around rooms to form "sound labyrinths," heard over specially designed receivers, often embedded in headphones (Figure 6.3). In 2003 she began a series of site-specific urban *Electrical Walks* in which listeners don special headphones and follow maps that guide them through a series of specific sonic landmarks resulting from the electromagnetic signals emanating from ATMs, the electrical grid, heavy machinery, security gates in stores, elevators, subways, etc. (see her video on the website).

Slovak artist Jonáš Gruska not only works extensively with electromagnetic phenomena in his own music but, under the name of Elektrosluch, designs and sells a range of beautiful inductive pickups optimized for sound (Figure 6.4).

Crystal radios receive signals through a seemingly impossible combination of bits of rock, diodes, coils of wire, and headphones—no batteries needed. Sawako Kato (Japan and US) builds radios from these raw materials in her séance-like performance work *Ishi–Listening to Stones* (Figure 6.5 and her video on the website). And by playing the 12 keys of his FMkbrd, Portuguese musician Vasco Alves gates on and off 12 radios, tuned as he wants, to create an update of Cage's infamous *Imaginary Landscape No. 4* (see Figure 6.6 and his video on the website).

Figure 6.4
Jonáš Gruska, Elektroucho Pro pickup.

Figure 6.5 Sawako Kato, crystal radio made with pyrite.

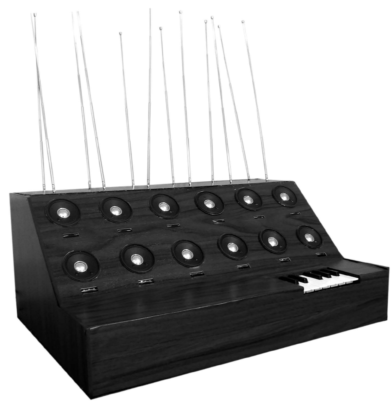


Figure 6.6 Vasco Alves, FMkbrd.

Telephone pickups are available pretty cheaply from a wide range of online sources, but there are alternative devices that work just as well for our purposes:

- Guitar pickups. A guitar pickup consists of a coil of wire wrapped around a magnet, inside an elegant package (Figure 6.7). At guitar repair shops you can often buy (rather cheaply) low-end pickups removed when better ones are installed. Use your new soldering skills to connect the pickup to a plug that fits your amp. As most guitarists know, "single coil" pickups are sensitive to hum and electromagnetic interference, while "humbuckers" are so called because they tend to suppress exactly the kind of weird signals we want to hear. But we can work with whatever you can find—even a pickup still sitting in a guitar.
- Inductors. The inductor is a passive electronic component used in a variety of electronic circuits. They are inexpensive and available through numerous online

Figure 6.7
A guitar pickup with case broken away to reveal internal coil.

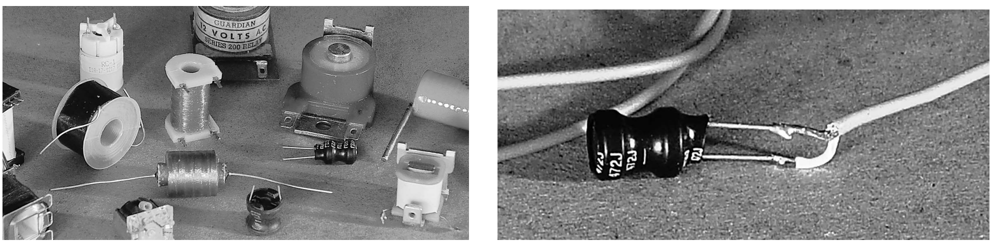


Figure 6.8 Assorted inductors and relay coils (left) and a coil pickup made by soldering an inductor to a shielded cable (right).

retailers. Inductance is measured in (the charmingly named) “Henrys” and comes in a wide range of values. The exact value is not important for our application, but the *larger* the value the *louder* the output. A 100 mH (milli-Henry) puts out a signal of about the same level and impedance as a guitar pickup; as such it interfaces well with most amplifiers, mixers, and effect pedals (Figure 6.8, left).

- Almost anything else. Any chunk of ferric metal (a magnet or piece of metal that is attracted to a magnet) with enough wire wrapped around it will pick up magnetic fields. You can solder a plug to the wires coming out of any transformer (such as a wall wart external power supply), a relay coil, a solenoid, or an electric motor; connect it to your amp and listen (Figure 6.8, right).

SNIFFING

Plug whatever coil you have—telephone pickup, guitar pickup, inductor, wall wart, etc.—into your amplifier, turn it up (the signal can be quite quiet), and repeat the experiments we did with the radio. Boot your laptop and note the change in sound as you move the coil across its surface, from the CPU area, to the RAM, to the battery, to the screen (see Andy Keep’s audio on the website). Stick it onto a portable CD player and notice the racket as you press “>>|” and “|<<”—you’re hearing the frantic electromagnetic twitching of all those little motors and servos that spin the disk and move and focus the laser. Eavesdrop on a cell phone as you initiate a call. Take money out of an ATM. Listen to motors in fans, vibrators, and toys; notice the change in pitch as you vary the motor speed. Cozy up to a neon sign.

Coil pickups are highly directional, like stethoscopes or shotgun microphones. Wire up two of them to the left and right inputs of a portable audio recorder before you ride on a subway—the sounds of the motors and doors as you come in and out of stations acquire a vivid stereo presence (see Nicolas Collins's *El Loop* audio file on the website). (Refer to Figure 7.17 in Chapter 7 for advice on soldering two coils directly to a stereo plug.)

If you move the coil close enough to the speaker of your amplifier, it will begin to feed back with the coil that moves the speaker cone—if it doesn't squeal, turn it around so the other end of the coil faces the speaker. As with feedback between a microphone and speaker, the pitch is a function of the distance between the two parts, but here the pitch changes smoothly and linearly, without the odd jumps caused by the vagaries of acoustics, giving you a Theremin-like instrument. Try this with a full-size guitar amplifier for greater pitch range and more impressive volume. If you connect your amplifier to a raw speaker, you can place the speaker on its back like a candy dish, rest the coil inside, and turn up the gain: the coil should bounce around and change pitch as it feeds back.

A coil glued to the magnet on the back of a speaker can also serve as a convenient, wide-range pickup for recording or further amplification. Stick it on the speaker inside a toy or radio, plug the cord into your amplifier, and you'll hear bass that never emerges from the speaker's small cone. When attached to the speaker in a guitar amp, the coil gives you a sound halfway between a mic'd cabinet and a direct out: the "woof" of the speaker minus the boom of the roomtone. A cheap and handy DI box (Direct Box).

VLF

Fans of what is known as VLF (very low frequency) radio make huge coils by wrapping meters of wire around big wooden crosses and then camp out on remote hilltops. Get far enough from civilization's ubiquitous 60/50 Hz hum and you may be lucky enough to pick up the Aurora Borealis, "whistlers" induced by meteorites self-immolating as they enter the earth's atmosphere, the pipping of GPS satellites, or top-secret submarine radio communication.

Take 100 feet or so of ordinary insulated wire and wrap it around a lightweight wooden armature (nail two 5-foot pieces of 1-inch by 2-inch lumber together and notch the ends to keep the wire from slipping off). Solder one end of the wire to the *tip* of a plug that fits into your amp or tape recorder and solder the other end to the *sleeve*. Plug in, turn on, drop out.

DUELING RADIOS

In the process of receiving and demodulating transmissions, radios in turn generate and send back out intermediary electromagnetic signals. These transmissions aren't very powerful, but they are strong enough that airline passengers are warned not to turn on radios in flight for fear of disrupting the navigation system (knowing just how weak these signals are further diminishes my faith in air travel safety). What one radio transmits another will receive: turn on two AM radios, tune them to the dead band at the end of the tuning range, and set them close together; by moving the radios and varying the tuning, you should be able to produce Theremin-like whistling and interference patterns (see David First's audio and video on the website).

ON AIR

Of course, instead of just eavesdropping on the Aurora Borealis, you can join the ranks of the BBC and start broadcasting yourself. See Chapter 27 for instructions on building your own radio transmitter.

NOTES

1. Alvin Lucier, *Sferics*, Lovely Music, Ltd.VR 1017, 1988.
2. Karlheinz Stockhausen, *Kurzwellen*. Deutsche Grammophon LP 1969.

CHAPTER 7

How to Make a Contact Mike
Using Piezo Disks to Pick Up Tiny Sounds

You will need:

- A battery-powered amplifier.
- A piezoelectric disk.
- About 8 feet of lightweight shielded cable.
- A plug to match the input jack on your amp, recorder, or mixer.
- Plastic insulating electrical tape.
- Optional: a can of Plasti-Dip or similar rubberized plastic paint (sold in hardware stores for dipping tool handles).
- Small spring clamps.
- Molex-style terminal block.
- Hand tools, soldering iron, and electrical tape.
- Sparklers, small blowtorch, guitar strings, metal scrap, Slinky, springs, and condoms.
- Optional: a graphic equalizer.

THE PIEZOELECTRIC EFFECT

In addition to electromagnetism, another common principle of reversible sound translation is the "piezoelectric effect," which depends on the electrical properties of *crystals*, rather than electromagnetism: bang certain crystals with a hammer and they will generate a pretty sizeable electrical signal (enough to light a flashlight bulb); conversely, if you send an electrical current into the crystal, it will twitch a bit.

Piezoelectric disks, made by bonding a layer of crystal to a thin sheet of brass, are everywhere today inside almost everything that beeps: appliances, phones, toys, alarm clocks, computers, etc. (Figure 7.1). Because they are manufactured in huge quantities, out of very few separate parts (many fewer than go into a traditional speaker), they are incredibly cheap. They also happen to make even better contact microphones than they make speakers. Drum triggers and commercial contact mikes are often made from piezo disks and sold at absurdly high prices. As gratuitous markups are anathema to the hacker, making your own cheap contact mikes is a satisfying way to while away an evening.


Figure 7.1
Assorted piezo disks.

HOW TO MAKE A PIEZO DISK CONTACT MIKE

1. If you're not comfortable soldering, try to find a piezo disk that already has wires attached—soldering directly to the disk's surface can be touchy. Get a few of them in case you break any. You can salvage disks from all sorts of trashed electronic devices or buy them from any number of internet retailers. If you can only find wireless disks, skip ahead to step 8.
2. The disk may be encased in a kind of plastic lollipop. If so, carefully pry open the case and remove the disk. Try not to impale yourself, and avoid bending or scratching the disk, since this can result in distortion. Prior work experience at the Oyster Bar pays off here.
3. The disk may have a tiny circuit board attached. Snip off the connecting wires close to the circuit board so that the wires attached to the disk are as long as possible.
4. Once removed, the disk should appear as a circle of gold- or silver-colored metal, with a smaller circle of whitish crystal within. Depending on the design, there will be two or three wires connected to the disk. One will be connected to the metal portion, somewhere near the edge; this is the "ground" connection. One will connect to the main portion of the inner crystal circle; this is the "hot" connection. On some disks there will be a narrow, tongue-like shape differentiated within the crystal, to which the third wire connects; this we will call the "curious but unnecessary" (CBU) connection.
5. Cut the connecting wires so that they protrude about 2 inches from the disk. Strip about 1/2 inch of insulation from the ends of the ground and hot wires; don't bother to strip the CBU wire—you may cut it off near the surface of the disk or leave it dangling. Tin the stripped ends.
6. Shielded cable consists of stranded wire inside insulation, surrounded by a layer of braided or twisted wire, which is in turn covered by another layer of

insulation—like a matryoshka doll. A cross section looks like tree rings or a target. Shielding a hot signal by wrapping it with ground protects it from hum and other electromagnetic interference. Shielded cable comes with any number of internal conductor wires, but for audio purposes most varieties have one or two internal conductors plus the shield. Unless otherwise specified, we only need cable with a single internal conductor plus shield; if your cable has more internal conductors (in addition to the shield), that's OK, but we'll only use one, so pick the color you like and cut off the others.

Rule #9: Use shielded cable to make all audio connections longer than your hand is wide, unless they go between an amplifier and a speaker.

Cut a 5-foot section of shielded cable, the thinner and more flexible the better. Strip back 1 inch of the outside insulation. Unbraid the shielding and twist it into a single thick strand. Now strip back 1/2 inch of the inner insulation and twist the center conductor into a neat strand. Keep the two strands separate. Tin both strands, quickly and carefully, so as not to melt back the insulation (Figure 7.2).

7. Twist together the hot wire from the center of the piezo disk and the inner wire from the shielded cable. Solder them together. Twist together the ground wire from the edge of the piezo disk and the shield from the shielded cable. Solder them together (Figure 7.3). Wrap both joints separately with a little piece of electrical tape so that they cannot short out if they touch each other.
8. If you are using a piezo disk that arrived without wires already connected, you will have to solder the tinned ends of the shielded cable directly to the disk's surface. First tin a contact spot somewhere on the metal perimeter—this is not so

Figure 7.2 Shielded cable prepared for soldering.

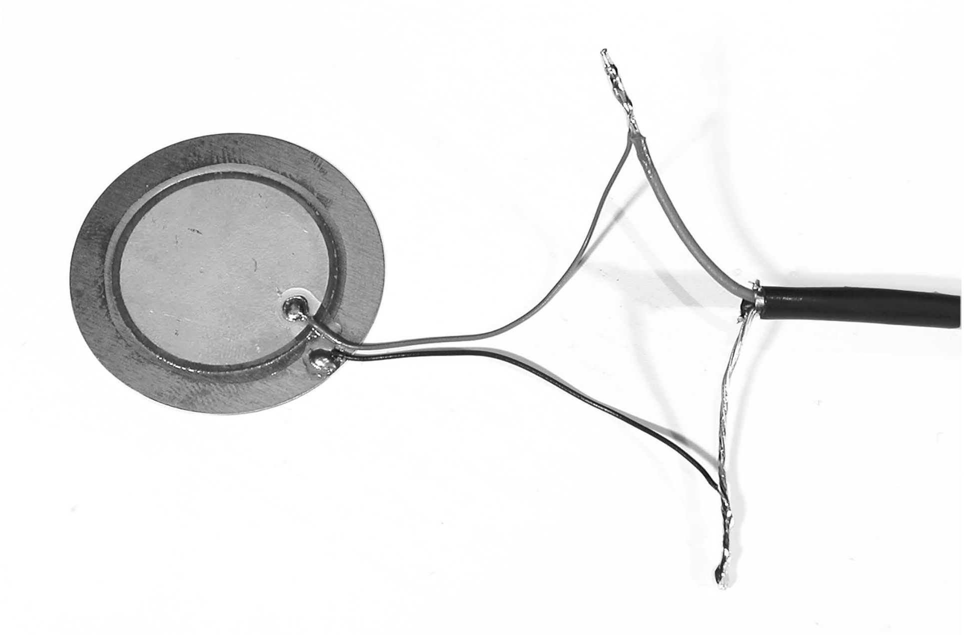


Figure 7.3 Piezo disk with leads soldered to cable.

tough, just make sure the metal is clean and shiny (you can rough it up a bit with a file, steel wool, or fine sandpaper). Solder the end of the tinned shield to the tinned spot on the edge of the disk. Snip the bare, tinned end of the center conductor down to 1/4 inch or less and solder it as quickly as possible to the tinned pad on the crystal itself—as you will discover, this material shies from heat (Figure 7.4). If everything holds, congratulations—you've just earned your Advanced Soldering merit badge! If, on the other hand, you find this impossibly difficult, go back out, find a disk with wires attached, and start all over again.

9. Now move on to the other end of the shielded cable and strip it as in step 5: 1 inch outer insulation, twist shield, 1/2 inch inner insulation, twist conductor, tin the wires.
10. Select the appropriate plug for whatever device you plan to use with your contact mike: a 1/4-inch (6.35 mm) "guitar plug" for connecting to a guitar amplifier or most mixers or a 1/8-inch (3.5 mm) plug to connect to phones and some flash recorders. (If you want to connect the contact mike to the XLR inputs of a microphone preamplifier, please see Chapter 11, Figure 11.3, for wiring instructions.) Unscrew whatever plug you are using. Slip the barrel back over the shielded wire toward the disk so that the threaded portion faces the freshly tinned end (don't forget this; it's usually impossible to get the barrel on *after* you've soldered). Unscrewing the barrel should reveal two solder tabs on the plug: the shorter one connects to the "tip" of the plug and the longer one connects to the "sleeve."

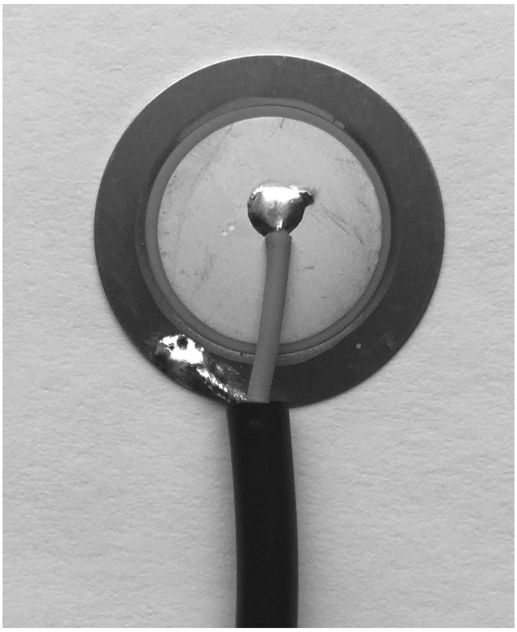


Figure 7.4
Piezo disk with shielded cable soldered directly to surface.

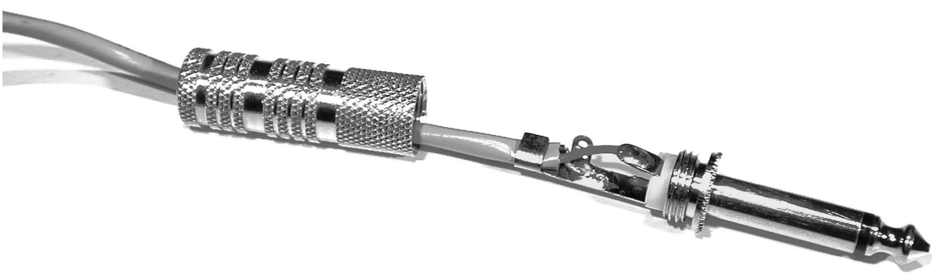


Figure 7.5 Cable soldered to plug.

Solder the inner conductor of the shielded wire to the tip of the plug and the shield to the sleeve (Figure 7.5). Sometimes there are small holes in the connector tabs that you can hook your wire onto so that it is held in place before soldering. Otherwise you will have to tin the tabs and hold each wire against its respective tab while soldering—a job for three hands, a vise, or a fearless buddy with a steady hand.

Now is as good a time as any to introduce the 10th Rule of Hacking, if it is not obvious already:

Rule #10: Every audio connection consists of two parts: the *signal* and a *ground* reference.

In the case of a contact mike, the *signal* comes from the white part of the piezo disk while the *ground* is the brassy bit; on the plug the tip carries the signal and the sleeve is the ground. In future chapters I may get a bit sloppy and only refer to the signal when describing connections—always assume that a ground connection must accompany every signal.

11. Plug into your amp and check that your new contact mike works—tapping the mike should make a solid thunking sound. If there is no sound, check the joints at both ends of the cable to make sure they are good and there are no shorts. If there is hum, you may have reversed the hot and ground connections—de-solder and reverse them. If it works, screw the barrel down onto the plug and test again—sometimes squeezing the barrel down over a marginal solder joint will break or short it. A small piece of electrical tape can be used to isolate the connections if excess wire causes problems when the barrel is screwed down.

    Whoops! Did you forget to slide the barrel onto the wire before you soldered? If so, de-solder the plug and go back to step #10, but don't feel too stupid—everybody makes this mistake at least once. But remember:

Rule #11: Don't drink and solder.

12. When you are sure you have an electrically functional contact mike, cover the ceramic side with a piece of electrical tape—you can trim it around the circumference with scissors or a knife, or you can wrap the edges over to the other side of the disk. Don't worry, it doesn't have to be pretty.
13. Buy a can of Plasti-Dip or similar rubber paint if possible—it's sold in most US hardware stores as a coating to improve one's grip on tool handles. In Europe it's harder to find, but sometimes you can obtain "latex paint" that works as well. Find a well-ventilated space. Open up the can and stir. Slowly dip the contact mike end of your cable into the goop until you have covered the wire past the electrical tape (Figure 7.6). Slowly withdraw it and hang it up (preferably outside) to dry. Go away and take a break—this stuff is stinky. You can dip a second layer after the first one dries thoroughly, which can take several hours. More than three layers tend to muffle the sound, so don't overdo it without listening carefully after each new layer.

Figure 7.6 Contact mike encased in Plasti-Dip.

The tape and Plasti-Dip treatment serves several functions:

1. Strengthens the connections between the wires and the piezo disk.
2. Insulates the disk from electrical shorts and prevents hum when you touch it.
3. Waterproofs the contact mike so you can use it to record underwater sounds, freeze it in ice cubes, dangle it in a drink, etc.
4. Deadens slightly the pronounced high-frequency resonance of the disk (similar to the effect of gaffing tape on the head of an unruly snare drum).
5. Looks really cool.
6. Dipping fumes can offer an illicit treat after all that soldering.

The discovery of Plasti-Dip as the perfect contact mike sweater must be credited to the ingenious Robb Drinkwater of the School of the Art Institute of Chicago.

WHAT TO TRY WITH YOUR CONTACT MIKE

Contact mikes are great for greatly amplifying sounds hiding in everyday objects. The trick is making firm physical contact with the vibrating object.

Use double-stick tape to attach the mike to the surface. Try: guitars, violins, drums, pots and pans, wrists and knees, foreheads, pinball machines.

Use small spring clamps to hold things to the contact mike. Try: strips of metal, gaffing tape, rulers, popsicle sticks.

"Terminal strips" are used to make electrical connections in lamps and other appliances. They can be salvaged from discarded appliances (Ikea loves them in lighting fixtures) or bought from any number of sources, including neighborhood hardware stores. Cut the terminal strip into small sections and clamp them onto the mike with a spring clamp. Insert thin objects into the terminal openings and hold them in place by tightening the screws (Figure 7.7). Try: Slinkies, springs, loose guitar strings, toothpicks, satay skewers, broom straws, porcupine quills, cactus needles.

This is an excellent way to replicate the old-fashioned phonograph cartridges used by John Cage in his visionary work of live electronic music, *Cartridge Music* (see Art & Music 2 "John Cage—Father of Invention").

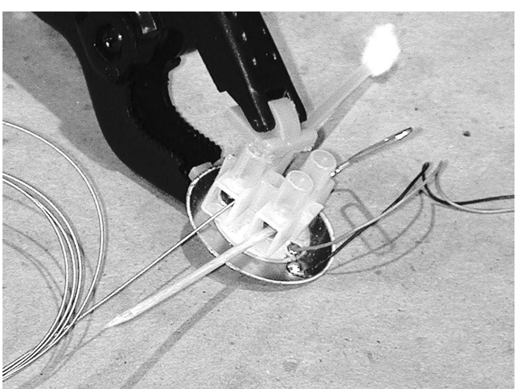


Figure 7.7
Terminal strip holding junk clamped to a contact mike.

ART & MUSIC 2

JOHN CAGE—FATHER OF INVENTION

The influence of John Cage (1912–1992) on American avant-garde music cannot be overstated. Given the breadth of his impact as a composer and theoretician, his significance in the rise of hacker electronic culture is sometimes overlooked. Throughout his career, Cage had a passionate curiosity for new sounds and compositional strategies. Lacking institutional support in the way of orchestral commissions and the like, Cage, the son of an inventor, chose to develop new instruments from everyday technology and commonplace objects. At the beginning of his career, he literally made do with rubbish: his early percussion music, such as *First Construction in Metal* (1939), used brake drums and other scrap iron from junkyards. In the 1940s, Cage (together with Lou Harrison) inserted screws, erasers, and other pocket detritus into the strings of pianos to create the gamelan-like sounds of the prepared piano. *Imaginary Landscape No. 1*, composed in 1939 for piano, bowed cymbal, and record player, was the first documented piece of music to feature the DJ as a musical performer, while *Imaginary Landscape No. 4* (1951) was composed for a dozen ordinary radios.

In *Cartridge Music* (1960), performers substitute springs, twigs, pipe cleaners, and other thin objects for the needles in cheap record player pickups; the surprising richness of these enormously amplified "microsounds" rivaled the more labor- (and capital-)intensive synthetic sonorities coming out of the European electronic music studios and opened the ears of a generation of sound artists to the splendor of the contact mike. *Hpschd* (1967–1969), composed in collaboration with Lejaren Hiller, was one of the first works of computer music.

Cage's later music, such as *Études Australes* (for piano, 1975) and *Ryoanji* (for mixed ensemble, 1983–1985), reverted to more traditional instrumental resources. (Cage once told me, "If I don't write for these virtuosos they'll have to play music by even worse composers.")[1] But the ethos of hacking lived on in his continually surprising methodology and in his persistent invention of new performance techniques.

Many metals make unusual sounds as they heat and cool. Clamp a fireworks sparkler in the terminal block, light, and listen (see Nicolas Collins's video *Tall Poppies* on the website). Or clamp steel wire and heat with a torch (see Richard Lerman's audio track on the website).

COMPANION @ WEBSITE

Although a naked piezo disk picks up very little airborne sound (try speaking into one), you can make unusual air mikes by clamping the disk to lightweight, flexible material. Try drum heads, thin sheets of metal, glass windows, plastic "clamshell" packaging from salad bars or toy purchases, Styrofoam or paper cups, etc.

Connect the contact mike to an amplifier and wire a raw speaker to the amplifier output jack. Place the speaker on its back, like a candy dish. Rest the contact mike inside the cone and turn up the gain. The contact mike should jump up and down as it feeds back with the speaker—a slightly higher-tech variation on the jumping speaker in Chapter 3. (This is the underlying principle of Lesley Flanigan's Speaker Synth—see her video on the website.)

COMPANION @ WEBSITE

Once waterproofed with the electrical tape and Plasti-Dip, the contact mikes will also serve as hydrophones.

- Fill a plastic yogurt container with water, drop in the contact mike, and pop it in the freezer. Listen as it freezes. Once frozen, remove the ice block from the container, float it in a bowl of hot water, and listen to it melt (see Collin Olan's audio track on the website).
- Drop it in the water next time you go fishing and check if the fish are really laughing at you.
- Hold the mike in your mouth while you drink or chew, but please observe safe sex practices: put an extra layer of protection between you and electricity by encasing the contact mike in a condom or balloon, and

NEVER CONNECT A WET DISK TO AN AMPLIFIER OR RECORDER CONNECTED TO AN ELECTRICAL OUTLET—ONLY TO BATTERY-POWERED EQUIPMENT.

A few hours spent with a contact mike and a fistful of junk should convince you of the significance of the Second Law of the Avant-Garde (and the First Law of Pop):

Make it louder, a lot.

This is also the credo of the "Piezo Music" movement that sprang up in the backwash of Cage's experimentation and the fortuitous invention of the economical piezo disk (see Chapter 7, Art & Music 3 "Piezo Music" and "The Contact Microphone: A Cultural Object" on the website).

PIEZO MUSIC

ART & MUSIC 3

In the aftermath of Cage's *Cartridge Music*, many sound artists sought affordable techniques for amplifying mechanical vibration and microscopic sounds. Since the mid-1970s, the proliferation of piezo disks in beeping appliances has put contact mikes within reach of anyone with a soldering iron. Whether as pickups on mandolins or as hydrophones for eavesdropping on whales, the disks have insinuated themselves into diverse corners of our recorded soundscape and have given rise to a genre called "piezo music." Hugh Davies (1943–2004, UK) and Richard Lerman (US) were two of the earliest innovators. Davies began building piezo-amplified instruments in the 1970s, the most poetic of which consists of a disk with short steel wires soldered around its rim. By plucking or blowing gently at these wires, he could elicit a wide range of surprisingly deep, marimba-like sounds, which he incorporated into composed and improvised work.[2] Lerman (who has for many years maintained an informative website with tips for working with piezo technology) uses similarly bewhiskered disks and plays them with a small blowtorch: the whoosh of the gas creates an effect similar to that of bowing a cymbal, and when the wire heats and cools, it snaps with gong-like solemnity (see Figure 7.8 and his audio track in the "Contact Microphones & Electret Microphones"

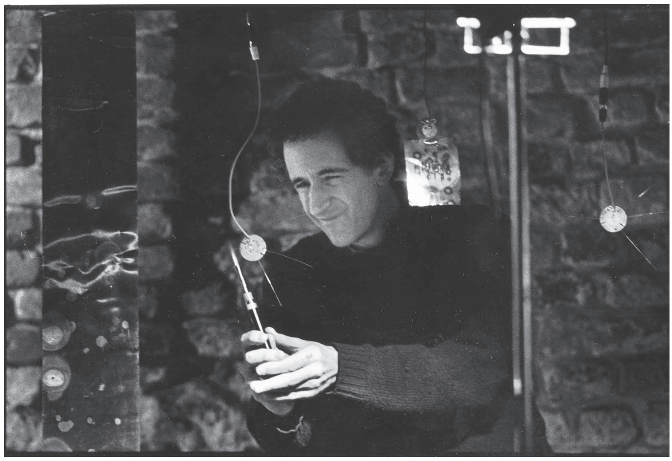


Figure 7.8 Richard Lerman performing *Changing States 6*.

Photo credit: Andrzej Kramarz, used by permission of Richard Lerman and Andrzej Kramarz. © Andrzej Kramarz.

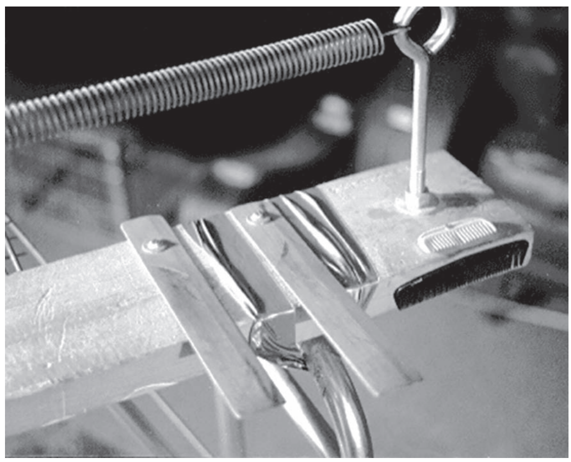


Figure 7.9 Eric Leonardson, "Springboard."

Photo credit: © Eric Leonardson, used by permission.

section of the Gallery on the website). Eric Leonardson (US) performs regularly on his Springboard—a plank of wood festooned with springs, wires, and other bits of scrap metal that, when heard through the piezo pickup, evoke the sound world of a Balinese gamelan (see Figure 7.9 and his video in "Contact Microphones & Electret Microphones").[3] ADACHI Tomomi's Tomoring takes a different design angle on amplifying springs, rubber bands, and combs (see Figure 7.10 and his video in "Contact Microphones & Electret Microphones"), and Ivan Palacky (Czech Republic) creates similar textures by adding contact mikes to a 1970s-era knitting machine (see Figure 7.11 and his video in "Contact Microphones & Electret Microphones").

Recently, several radical turntablists have taken the cue from Cage and extended their technique beyond the vinyl groove. Otomo Yoshihide (Japan) inserts bent wire and springs into the cartridge opening to "play" a cymbal placed on a turntable spindle—plugged into a guitar amp and turned up well past the point of feedback, the resulting din is one of the loudest things I have ever heard. By adding two extra tonearms to his turntable, Janek Schaefer (UK) turns any record into a three-voice round or canon (see his video in the "DJs" section of the Gallery on this website). Using his stylus pen, Vasco Alves (Portugal) can play static records by drawing on them. Australian Michael Graeve assembles an

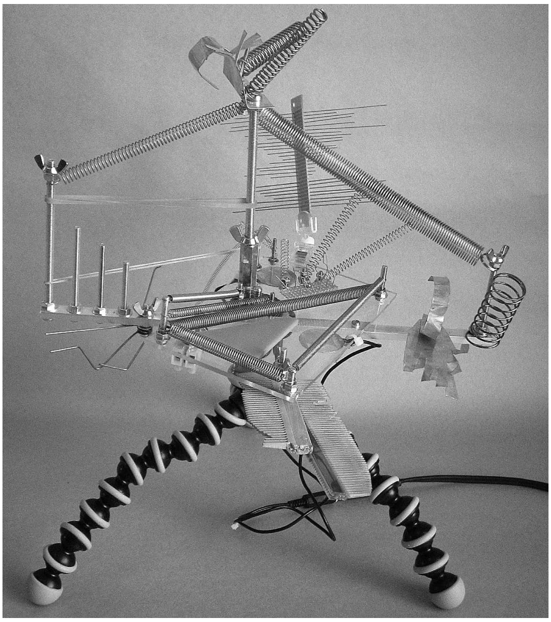


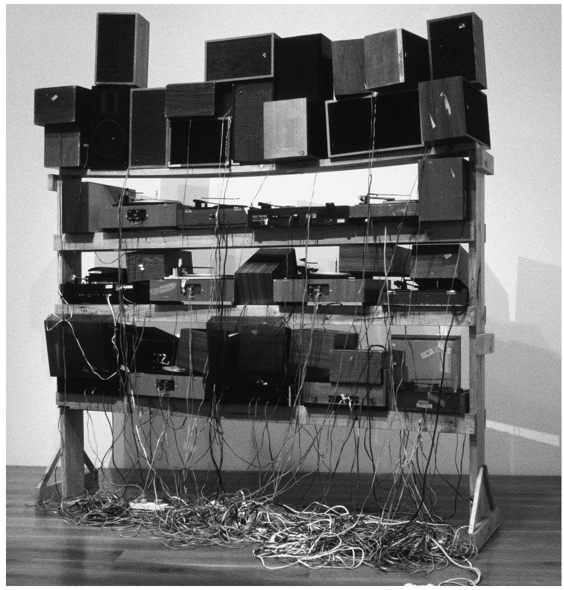
Figure 7.10
ADACHI Tomomi, "Tomoring."
Photo credit: © ADACHI Tomomi, used by permission.

Figure 7.11 Ivan Palacky, Amplified Dopleta 160 knitting machine.
Photo credit: © Martin Vlcek, used by permission of Ivan Palacky.

impressive collection of cheap old hi-fis and builds thick, dronelike textures from feedback and the sound of the needles playing the rubber mats of the platters (Figure 7.12).

Many artists have used piezo contact mikes to record inaudible or elusive soundscapes. In the mid-1970s, Australian neuroscientist Alan Lamb began composing music with recordings of the wind-driven sounds of abandoned telegraph wires (subsequently, several artists, including Warren Burt and Jon Rose, began creating music by bowing and otherwise playing abandoned fences around the

Figure 7.12
Michael Graeve, *0, 16, 33, 45, 78* (2004).
Photo credit: Jeremy Dillon, © Michael Graeve, used by permission.



Australian countryside). Collin Olan (US) froze waterproofed contact mikes in a block of ice and recorded the cracks and whistles of escaping air bubbles as it thawed (see his audio track in "Contact Microphones & Electret Microphones"),[4] while Peter Cusack (UK) used similar homemade hydrophones to record the breakup of ice on Siberia's Lake Baikal (see his audio track in "Contact Microphones & Electret Microphones").[5]

Read Daniela Fantechi's "The Contact Microphone: A Cultural Object" in the "Culture and History" section of this website for a deeper look at the history of this versatile musical tool.

MAKE IT NICER

The piezo disk contact mike has a very wide frequency range, from sub-audio (piezo sensors are used in seismographs) to ultrasonic (picking up music that will set your dog's tail a thumpin'). But the response is far from flat and is colored in particular by a peak at the frequency at which it would normally beep if used as a speaker. You will probably want to pass the signal through an equalizer of some sort. Most mixers have some kind of EQ (as equalizers are also known), and computer audio recording and editing software comes with various plug-in equalizers. But if you're using a contact mike for live performance, I strongly suggest you invest in a graphic equalizer in the form of a cheap guitar effect pedal (as mentioned at the start of the book)—an easy way to reshape the sound to your preferences.

HI-FI

Ultrasonic transducers, such as those found in many motion-detecting alarm systems, contain very small piezo disks (see Figure 7.13). These disks often have a slightly lower output level than the more common large brass ones, and are more expensive, but have

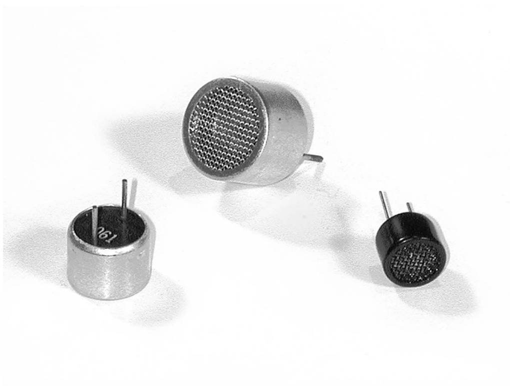


Figure 7.13 Some ultrasonic transducers.

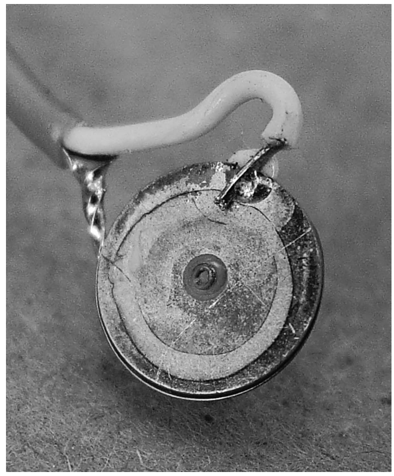


Figure 7.14 Element from ultrasonic transducer wired to shielded cable for use as contact mike.

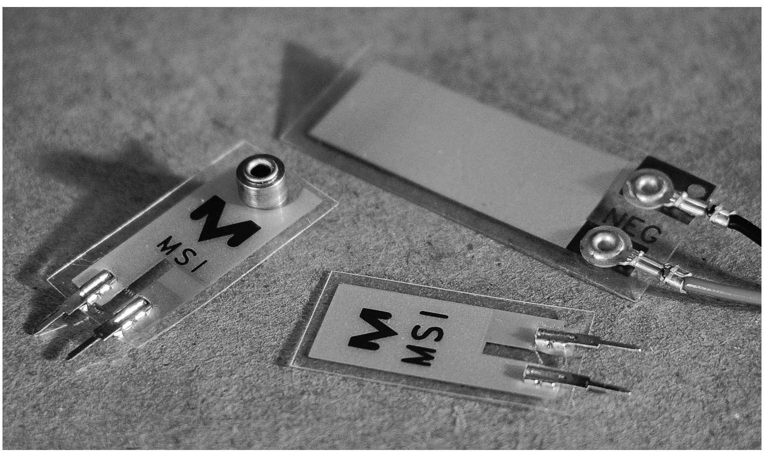


Figure 7.15 Piezo film vibration sensors from Measurement Specialties.

a flatter frequency response. You will need to solder one connection to each side of the disk since they have a different mechanical construction from the typical lollipop-encased beeper (Figure 7.14).

Although also somewhat more expensive than piezo disks bought on the surplus market, piezo film elements such as those by Measurement Specialties have a superior sound quality that justifies the cost in some cases (Figure 7.15).

STEREO

Those of you running the contact mike into the 1/8-inch (3.5 mm) stereo microphone input of an audio or video recorder have a few wiring options to consider. If you solder the mike to a mono plug (as discussed in step 10 prior), the audio signal will

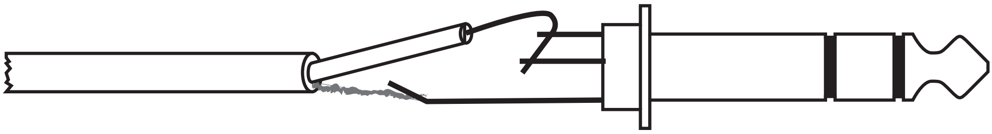


Figure 7.16 Hot lead of a single contact mike soldered to both the left and right connector tabs of a stereo plug.

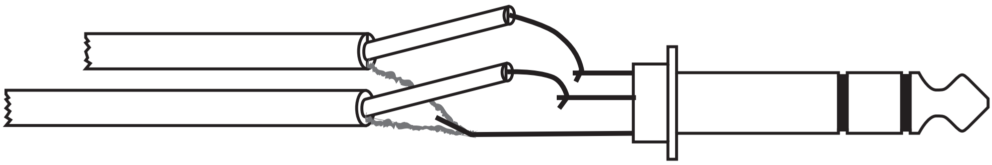


Figure 7.17 Wiring of a stereo contact mike pair to a stereo plug.

usually record on the left channel only (the right will be silent). If you open a stereo plug, you will notice that there are two short lugs in addition to the long sleeve—one connects to the *tip* of the plug (typically the left channel) and one to a *ring* between the tip and sleeve (the right channel). You can solder the hot/center conductor of your cable to both the tip and ring connectors (Figure 7.16), in which case the mike's signal will record on both the left and right channels. Or you can solder up two contact mikes; twist the shields of both together and solder to the long sleeve lug, then solder the inner "hot" conductor of each contact mike to one of the two short lugs so that one mike connects to the tip and one to the ring (Figure 7.17)—don't forget to slide the barrel over *both* mike cables before you start soldering. Voilà—stereo contact mikes!

HISTORICAL NOTE

The term "piezoelectric" originates in the Greek word *piezen*, meaning "to press." The discovery of the effect is credited to Jacques Curie and his brother Pierre. The latter, along with his wife, Marie Curie, is better known for his work with another rather energetic, if less user-friendly, phenomenon: radioactivity.

NOTES

1. Response to question in public lecture by Cage at Wesleyan University (Middletown, CT), February 1974.
2. Hugh Davies, *Sounds Heard*. Soundword, Chelmsford, UK, 2002.
3. Eric Leonardson, "The Springboard: The Joy of Piezo Disk Pickups for Amplified Coil Springs". *Leonardo Music Journal*, Vol. 17 (2007). Pp. 17–20.
4. Collin Olan, *Rec01, Listen 001*. Apestraartjje CD, 2002.
5. Peter Cusack, *Baikal Ice*. ReR CD, 2004.

CHAPTER 8

Turn Your Wall Into a Speaker
Resonating Objects With Transducers, Motors, and More

You will need:

- An amplifier with connections for external speakers.
- An audio transducer (see text)
- A second battery-powered amplifier or mixer.
- Your contact mike from the previous chapter.
- Another piezoelectric disk.
- Various plugs and jacks, as needed.
- One small audio output transformer (i.e., Xicon model 42TL013-RC or 42TL003-RC from Mouser.com).
- A few feet of lightweight speaker cable or stranded hookup wire.
- Electrical tape.
- Small spring clamps or clothespins.
- A sound source, such as computer or phone, and cable to connect it to the amplifier input.
- Small DC motors (from pagers, cell phones, vibrators, etc.).
- A low-voltage relay or solenoid.
- A small speaker.
- A wine cork.
- Glue or double-stick tape.

TRANSDUCERS

Sometime in the 1960s, the "audio transducer" appeared on the audio market. Bearing a passing resemblance to a hockey puck, the transducer consists of the guts of a loudspeaker: a magnet and a coil of wire (Figure 8.1). But instead of being coupled to a paper cone, the moving coil is attached to a threaded socket, by which the transducer can be bolted onto any solid object. In the "bigger-is-better" era of the hi-fi arms race, transducers were sold with the misleading advertising line of "turn your wall into a speaker"—which they did,

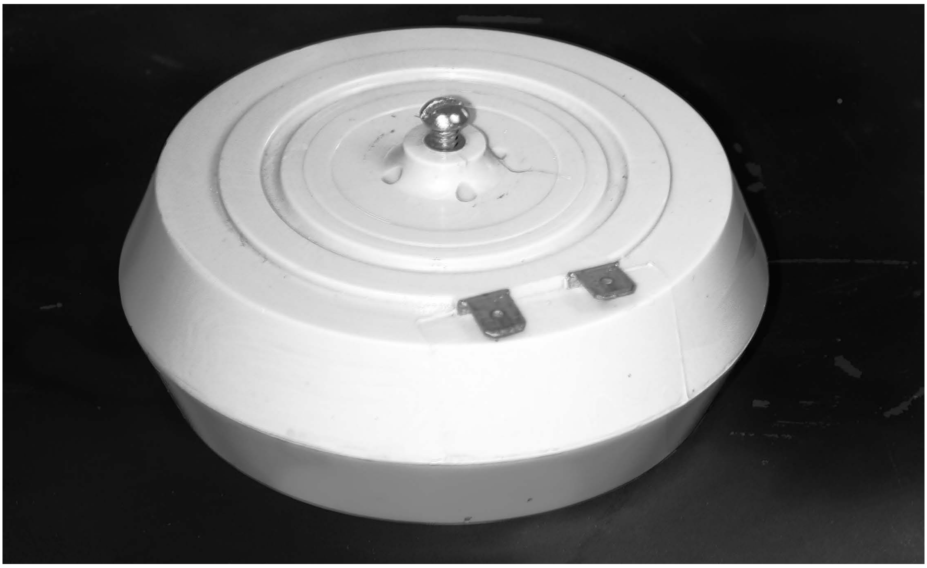


Figure 8.1 A Rolen-Star transducer.

but the frequency response of 120 square feet of sheetrock or knotty-pine paneling was far from flat: quirky resonances rendered Sinatra indistinguishable from Stockhausen—or from David Tudor, who, as John Cage's longtime musical enabler, saw the potential of the transducer as a signal processing tool, rather than an inadequate substitute for a loudspeaker (see Art & Music 4, "David Tudor and *Rainforest*"). The transducer transforms scrap metal into a singing sculpture, and a contact mike clamped to the metal turns the sculpture into a filter and reverberator—mechanical signal processing.

ART & MUSIC 4

DAVID TUDOR AND *RAINFOREST*

David Tudor (1926–1996) began his career as a leading pianist of the avant-garde. By the early 1950s, he was serving as pianist for the Merce Cunningham Dance Company and assisting in the realizations of both Cage's piano music and his electronic works. Tudor gradually abandoned the piano and emerged as the first virtuoso of experimental electronic performance. Expanding on Cage's work with the "found" technology of radios and record players, Tudor embarked on the (at the time quite arduous) process of acquiring enough knowledge of circuit design and soldering to construct his own new instruments. He believed that new, object-specific, intrinsically *electronic* musical material and forms would emerge as each instrument took shape: "I try to find out what's there—not to make it do what I want, but to release what's there. The object should teach you what it wants to hear."[1]

Although Tudor was not the first composer to make his own electronic instruments (he was inspired and assisted by pioneering composer/engineer Gordon Mumma), in no other composer's work is the ethos of music *implicit* in technology so fundamental and clear.

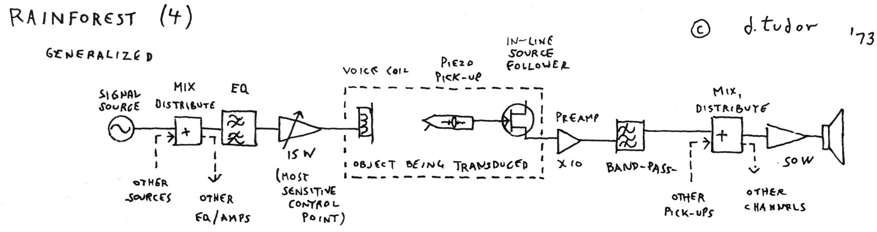


Figure 8.2 David Tudor, generalized diagram for *Rainforest IV*.

Beginning in 1968, Tudor composed a series of pieces under the title of *Rainforest*, culminating in *Rainforest IV*, developed in conjunction with a workshop in electronic performance that he gave in Chocurua, New Hampshire, in 1973 (Figure 8.2). The principle underlying all the *Rainforest* variations was similar: sounds are played through transducers affixed to solid objects; the objects filter, resonate, and otherwise transform the sounds; the processed sounds are directly radiated by the transduced objects, which serve as "sculptural speakers"; contact mikes on the objects pick up the vibrating surfaces of the objects, and these microsounds are amplified with contact mikes and heard through ordinary loudspeakers around the space. With an open-form score that encouraged experimentation in the design of sound generators and resonated objects, this work served as a creative catalyst for the workshop participants and, later, other young composers who were drawn to Tudor by word-of-mouth. They subsequently formed a loose collective ensemble called Composers Inside Electronics. Over the next decades, this group served as a laboratory for artist-designed circuitry and experimental electronic performance, presenting dozens of installations of *Rainforest IV* worldwide, as well as performances of works by individual members of the ensemble, which over the years included John D. S. Adams, Nicolas Collins, Paul DeMarinis, John Driscoll, Phil Edelstein, Linda Fisher, D'Arcy Philip Gray, Ralph Jones, Martin Kalve, Ron Kuivila, and Matt Rogalsky.[2] (See video of Oscillatorial Binnage's realization of *Rainforest* and Nakai/Johnsen's essay on Tudor, both on the website.)

Commercial failures, these early transducers (also known as "drivers") soon went out of production—Tudor and his handful of disciples may have been the only customers (see Art & Music 8, "Composing Inside Electronics," Chapter 15). But for some unfathomable reason, they have returned. Richtech Enterprises manufactures the Rolen-Star driver originally used by Tudor for *Rainforest*. Several companies make low-frequency drivers: Bass Shakers bolt to the floor of your trunk and turn the whole back end of your car into a big subwoofer (Figure 8.3); Butt Kickers drop your gaming throne into an M1 tank.

Big transducers require big amplifiers (25–100 watts). But there are also many smaller models available today, costing less and requiring only a modest amount of amplification. The two main manufacturers are Dayton Audio and Tectonic (Figure 8.4). The best source I know for these is Parts Express, but if you're outside the US, you can enter those brand names or the generic term "tactile transducers" on the

Figure 8.3 A Bass Shaker.

Figure 8.4
Some small transducers.

website of many electronic retailers, including the ubiquitous Amazon and eBay. Prices start at a few dollars/euros.

To send audio into the transducer, you'll need an amplifier with speaker terminals (as explained in Chapter 1). Strip both ends of some two-conductor speaker cable or two lengths of any flexible wire. Solder one end to each of the two terminals on the transducer. Connect the other end of the cable to one of the pairs of speaker connectors on your amplifier. Plug something into the audio input of the amp: laptop, phone, synth. Place the transducer on top of a cardboard box or an upside-down paper cup. Turn on the amp, play some music, and raise the amp gain until you hear your invented speaker.

Hold the vibrating transducer up against different objects. Thin materials work better than thick ones: pie tins, cymbals, paper cups, tin cans, and balloons, rather than bricks, anchors, and baseball bats. Firm contact is important—try double-stick tape.

A quaint reverb unit can be made by sending signals into a spring or plate of metal through a transducer and picking them up with a contact mike. This is similar to the technique used in early plate reverb units common in recording studios before digital reverb. You can patch the amp/driver/contact mike assembly into your mixer just as you would a reverb or effect processor: connect a send bus output to your driver-amplifier input, then connect the contact mike to any mixer input to amplify and balance in the "reverb" with the dry signal.

Often flexing or dampening the object can affect the character of its filtering of the original sound—this is especially noticeable with that irritating semi-rigid clear plastic packaging used around toys and other goods or the clamshell cases from salad bars, but you should also try loose guitar strings, Slinkies, balloons, plastic bags, bubble wrap, vinyl records, drumheads, old license plates, oil drums, buckets of water, and bowls of Jell-O. Whether as a reverb substitute in a mixdown situation or as part of a live performance setup, this is a cheap, easy, and fun route to unusual signal processing.

You can create feedback by plugging a contact mike into the amp input, and a transducer into the output, and attaching the two to the same object. Flexing or dampening the object can affect the feedback pitch and turn a piece of garbage (such as molded plastic packaging) into a playable instrument—an electronic musical saw. You can configure several channels of amps, drivers, and contact mikes to send audio signals through a series of objects for multi-stage processing; using Y-cords, you can branch off and mix after each resonator-object. Get together with your buddies, find a small garage, and form a transducer band.

You can insert some equalization, in the form of an inexpensive "stomp box" graphic EQ, to process the signal driving the transducer and/or the contact mike picking up the vibrating object—EQ will be especially noticeable in feedback networks.

HOMEMADE TRANSDUCERS

If you can't find a local source for transducers, don't despair—in the true spirit of hacking, you can also roll your own. Inside every cell phone is a small motor that spins an eccentric weight when you put your phone in "vibrate" mode. As luck would have

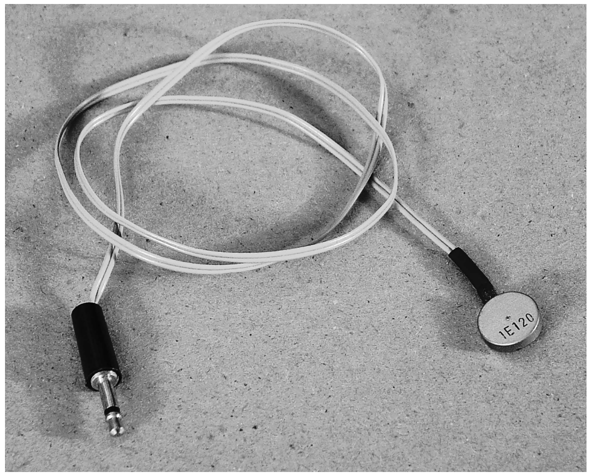


Figure 8.5 Cell-phone vibrator motor wired as driver.

Figure 8.6 A corked speaker.

it, this motor has similar electrical characteristics (impedance) as an ordinary speaker, and it couples well to any low-power amplifier. Connect two wires between the motor terminals and the speaker terminals of an amplifier (Figure 8.5). The motor should twitch in response to whatever sound source you play through the amp. Clamp the body of the motor directly to an object to transmit the vibration. Transducers motors work best with lower frequency sounds.

You can also make a driver by gluing a cork to the center of a small loudspeaker (Figure 8.6). Connect any sound source through an amplifier to the corked speaker and hold the speaker against a sheet of metal, drumhead, cymbal, etc. The cork vibrates the material and processes the original signal. You can pick up the vibrating surface with a contact mike, transforming metal scrap into a plate reverb (Figure 8.7). The end

Figure 8.7
Copper plate reverb.

of the cork can be treated to further affect the sound: a thumbtack brightens it (like a honky-tonk piano) while a piece of felt softens it, and wood is somewhere in between. A spring reverb can be constructed by stretching a fat spring or Slinky between the center of a speaker cone and a contact mike. You can attach the Slinky to the cone with tape or glue (Figure 8.8).[3]

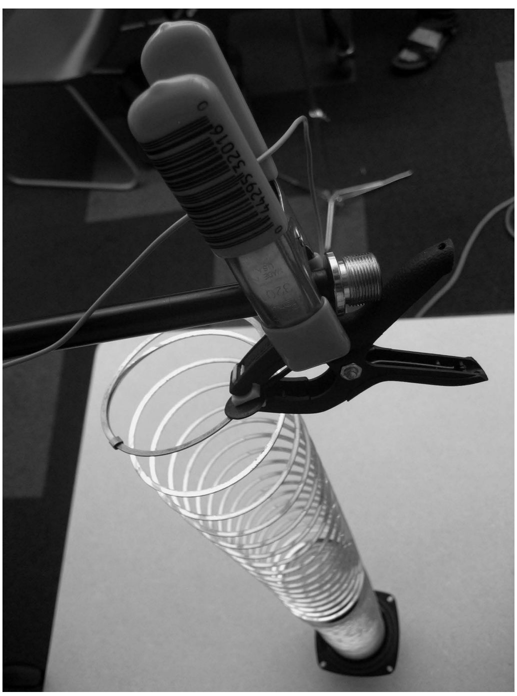


Figure 8.8
Slinky reverb.

PIEZO DRIVERS

Piezo disks beep very efficiently. They require very little current and therefore are well suited to battery-operated devices. As general-purpose loudspeakers they display a rather uneven, non-"hi-fi" response, but they can nonetheless be quite useful when coupled to other materials as lightweight transducers.

Due to the physical properties of the crystal, to get the most vibration out of a piezo disk speaker, it is necessary to feed it a very high voltage signal, albeit at a minuscule (and therefore harmless) current. A *transformer* is a chunk of iron wrapped in wire—basically two coil pickups (see Chapter 6) placed back to back. It functions as a kind of audio-lever that jacks up the voltage of an electrical signal without any additional "active" circuitry (transistors, integrated circuits, etc.). For this project we will wire up an *output transformer* backwards (see Rule #8) to step up the output voltage of a small amplifier from around 1 volt to over 100 volts. I learned this technique from Ralph Jones, a founding member of David Tudor's legendary ensemble, Composers Inside Electronics (see Art & Music 8, "Composing Inside Electronics," Chapter 15).

Output transformers were once used in most amplifiers using tubes ("valves," in the UK), as well as in certain older transistor-based designs, between the electronic circuitry and the loudspeaker. Although not so common these days, output transformers can be purchased from online retailers. The Xicon model 42TL013-RC is readily available (internationally) from Mouser.com.

The transformer has a *primary* and *secondary* side, with separate connections. The primary will have an impedance of around 1,000 Ohms (1 kOhm) and may have two or three wires. We will use the outer two connections—a center connection, if present, can be ignored. The secondary will often have just two connections and an impedance of 8 Ohms. A data sheet from the retailer will designate which side is which.

Solder one of the primary wires to the tip connection of a female jack that mates with the plug you used on the contact mike you made in the previous chapter. Solder the other primary wire to the sleeve connection of the jack (Figure 8.9). Polarity is irrelevant here—it does not matter which of the two primary leads goes to which connection on the jack.

Connect the *secondary* wires to the speaker connections of your amplifier (not to the headphone jack). Polarity is irrelevant here as well—it does not matter which of the two secondary leads goes to the + and which to −. You must use an amp connection designated for

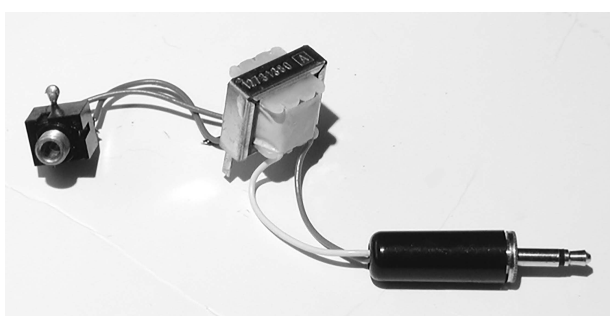


Figure 8.9
A transformer interface for driving a piezo disk.

a loudspeaker, not a headphone jack, which cannot drive the low 8-Ohm load of the transformer. A small hi-fi amp, like the one we used earlier for the other transducers, is suitable.

Plug a contact mike into the jack end of the transformer assembly (the primary side of the transformer). Plug a sound source into the amplifier input: laptop, phone, microphone, synthesizer, etc. Slowly raise the amp gain. The disk should start to radiate sound. If not, check your connections. Hold the disk up against different objects. Even more so than with the beefier transducers, thin materials work better than thick ones: pie tins, cymbals, paper cups, tin cans, and balloons. Firm contact is important—double-stick tape works well.

Whereas size does not greatly affect the loudness of a contact mike made from a piezo disk, it makes a big difference when you are making a driver. If you have a choice, use the largest possible disk to get a bigger sound out of whatever you are driving. You will notice that this is not a high-fidelity device: the sound is often limited to high frequencies, displays the peaked resonance inherent to a piezo disk, and can be quite distorted—larger disks have a wider frequency response, as well as being louder and less distorted than small ones.

By the way, if you place your finger across an un-Dipped, bare piezo disk while it is being run as a driver, you may experience a mild, not entirely unpleasant electric shock. This demonstrates how high the voltage gets when jacked up by the transformer. Electrical tape or Plasti-Dip will reduce distracting stimulation and protect the driver from damage. (The Taser stun gun employs similar step-up technology to more nefarious ends.)

SPOOKY ACTION THROUGH A DISTANCE

It is possible to resonate guitar strings electromagnetically, using coils instead of the vibrating transducers described previously. The EBow (invented in the late 1960s, patented in 1978, still in production in 2020) is a handheld device containing a pickup coil that is amplified and fed into a driver coil; when placed over a string on an electric guitar, the EBow produces local feedback to resonate the string, with a bowing effect (hence the name).

As an economical alternative to the EBow, find a low-voltage relay or solenoid. If you have a multimeter, measure across the coil connections: you want one that reads somewhere in the range of 5–30 Ohms. Plug your guitar into a small amplifier, attaching the relay/solenoid coil to the speaker output and holding this coil near the strings. A simpler method involves just hovering a battery-powered guitar mini-amp (such as those pictured in Chapter 1) above the strings and raising the gain to the point of feedback.

The guitar manufacturer Fernandes makes a sustaining device based on similar feedback principles—they market a few different guitars with the device built in, but you can also buy "Sustainer Kits" for retrofitting any existing instrument. Any of these systems can be hacked to resonate the strings with other signals besides feedback: open the device, find where the pickup connects to the amplifier circuitry, disconnect the pickup and solder a jack in its place, plug the guitar output into an amp, and try playing sounds into the strings.

If you move the driving coil or speaker too close to the guitar pickup, you'll hear the "dry" sound source you're using to resonate the string since the coil also functions as a sort of local radio transmitter. See Nicolas Collins's audio track *El Loop* on the website and the description of the Backwards Guitar in "Drivers," Art & Music 5, for more information on resonated strings.

COMPANION @ WEBSITE

ART & MUSIC 5

DRIVERS

David Tudor's *Rainforest* is one of the most conspicuous compositions to use transducers to resonate materials, but other sound artists have employed similar technology to different musical ends. In *Music for Solo Performer* (1965), Alvin Lucier placed loudspeakers on the heads of drums and against cymbals, gongs, and other percussion instruments; the performer's amplified brainwaves were routed to the various speakers, and the low-frequency (10–14 Hz) bursts played drum rolls on the percussion through the speaker cones.[4] In Lucier's *The Queen of the South* (1972), sounds are played through transducers affixed to a sheet of plywood (or other large plate) onto which fine powder has been sprinkled; as the pitch, timbre, and loudness of the sounds are changed, the powder forms different patterns according to the vibrational modes of the plate.

COMPANION @ WEBSITE

Berlin-based singer Ute Wassermann built her Windy Gong in 1995 (see Figure 8.10 and her audio track on the website). She sings into a mike, amplified through a small speaker with a cork attached to the center of the cone. The speaker is placed against the surface of a gong, which is in turn amplified with another microphone and contact mike. The vocal sounds are filtered and resonated by the gong, and the transformations can be manipulated by moving the speaker and microphone.[5] In *Kupferscheibe* (1993–1997) she extends long springs from speakers built into her clothing out to resonators encircling the performance area; her voice is processed by these spindly springs and heard, tin-can-telephone style, through the megaphones.

Figure 8.10 Ute Wassermann, Windy Gong.

For *Tisch* (1994–1995) German artist Jens Brand installed speakers inside a circular plastic café table. The surface of the table is oiled, and eight empty wine glasses are placed on top. Very low-frequency sound is played through the speakers. Although barely audible, the vibration from the speakers causes the glasses to move very slowly across the surface. When they make contact, they ring against one another. The piece ends when the last of the glasses tips over the edge and crashes onto the floor. For *Mini-Fan Music* (1992), Brand and his collaborator, Waldo Riedl, placed handheld fans next to a dozen string instruments strewn around the performance space; the fan blades strum the strings until the batteries run down (typically 3–4 hours with cheap batteries), the droning sound field slowly changing as the fans slip along the floor and lose speed.

Nicolas Collins has been working with "Backwards Electric Guitars" since 1981 (see Figure 8.11 and the audio excerpt from *It Was a Dark and Stormy Night* on the website). In these instruments sounds are sent *into* guitar pickups or coils (scavenged from relays) whose fluctuating electromagnetic fields vibrate the strings of guitars and basses. The strings filter, resonate, and reverberate the original sounds (similar to the effect of shouting into a piano with the sustain pedal depressed) and are picked up, amplified, and further processed through distortion and other typical guitar effects. The filtering is "played" by fretting and dampening the strings, like a one-handed guitarist.[6] Dan Wilson uses similar electromagnetic drivers to resonate everything from pitchforks to grills (see his videos on the website).

Dutch sound artist Felix Hess has created beautiful large-scale installations with tiny circuits that drive piezo disks directly (Figure 8.12). Pressed against small sheets of balsa wood by the weight of a stone, the piezos produce a cicada-like chirping of astonishing intensity.[7]

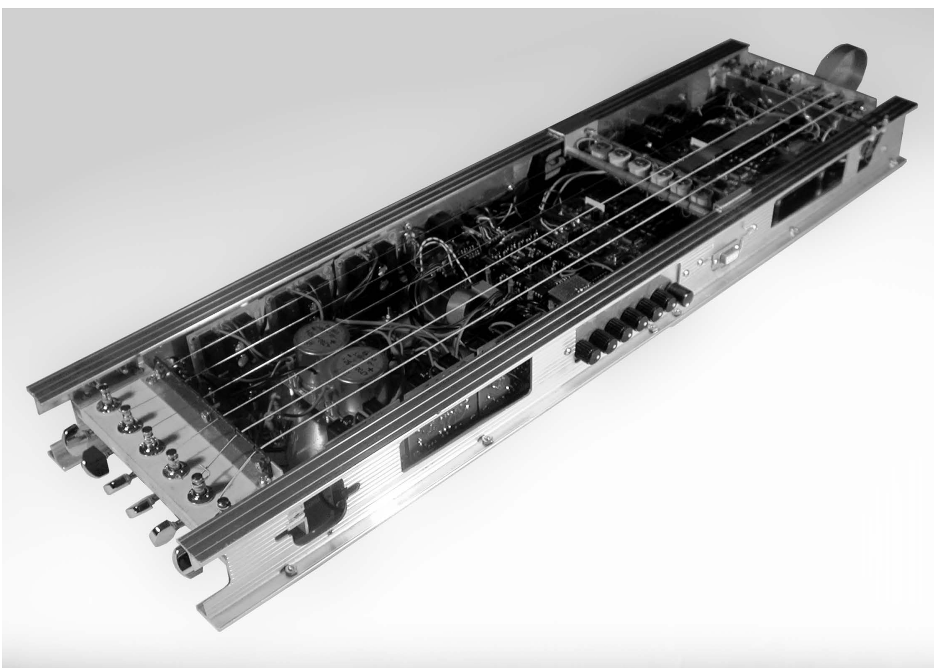


Figure 8.11 Nicolas Collins, Backwards Electric Guitar.

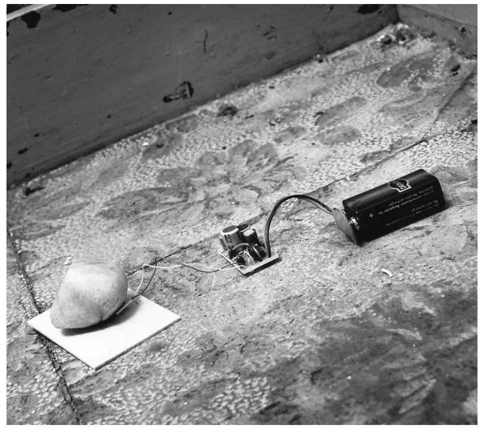


Figure 8.12
Felix Hess, Luftsdruckschwankungen circuit from installation of *It's in the Air*, for the Stroomgeest exhibition at Groot Bentveld, Netherlands, 1996.

Chicago-based artist Jesse Seay updated some of the fundamentals of *Rainforest* in her installation *Untitled (Resonant Objects)* (2006): a curtain of small, seemingly silent transducers hangs from the ceiling above a line of hollow objects (bottles, plastic yogurt containers, etc.); visitors are free to press the containers against the transducers, at which point her soundtrack miraculously radiates from the objects (see Figure 8.13 and her video on the website). Similarly, in his installation *Listening to the Reflection of Points*, Japanese artist Toshiya Tsunoda arranges a series of objects on top of small speakers through which white noise is played; each object resonates and radiates the sound differently. The New Zealand duo of Chris Black and Christine White create

Figure 8.13 Jesse Seay, *Untitled (Resonant Objects)*.

Figure 8.14
Chris Black and Christine White, detail of performance of *Crude Awakening*.

performances in which they filter sounds through materials ranging from guitar strings to baking pans, using only small loudspeakers and contact mikes (see Figure 8.14 and their video on the website). Canadian Frederic Brummer's "Speaker Drum" is self-explanatory (see his video on the website).

Electromagnetic fields can resonate anything ferric—you can step up from the one-dimensional Euclidean world of the string to the two-dimensional playgrounds of cookie sheets, grills, and pitchforks. Replace the guitar in the previous experiment with a loose guitar pickup or the coil you built in Chapter 6. Plug the pickup/coil into the amp and connect the amp output to the driving coil. Move both above any thin iron surface, and it should start to rattle. (See Dan Wilson's videos on the website.)

As Art & Music 5, "Drivers," shows, many artists have made creative use of the kind of "physical filtering" that can be accomplished by sending sounds through objects via all sorts of drivers.

NOTES

1. Victor Schonfeld, "From Piano to Electronics" (interview with David Tudor). *Music and Musicians* Vol. 20 (August 1972).
2. Nicolas Collins, "Composers Inside Electronics—Music After David Tudor." *Leonardo Music Journal* Vol. 14 (2004).
3. Nicolas Collins, "Build Your Own Reverb." *Tape Op*, 76, March–April 2010.
4. Alvin Lucier, *Music for Solo Performer*. Lovely Music, Ltd. VR 1014, 1982.
5. Ute Wassermann, *Improvised Music from Solitude in Eleven Parts*. CD, animato acd 6008-3, Germany, 1994.
6. Nicolas Collins, *It Was a Dark and Stormy Night*. Trace Elements Records, 1992; Nicolas Collins, *Sound Without Picture*. Periplum Records, 1999.
7. Felix Hess, *Light as Air*. Heidelberg, Germany: Stadgalerie Saarbrücken, Kehrer Verlag, 2001.

CHAPTER 9

Paper Speakers

JESS ROWLAND

You will need:

- Papers of various weights and character.
- Conductive metal foils (aluminum or copper).
- Some neodymium magnets.
- Amplifier with speaker terminals.
- Lightweight hookup wire.
- Optional: a craft cutter.

One logical extension of experimenting with transducers is building your own speakers. As the familiar black cone has shown, humble paper turns out to be an excellent mechanism for transferring mechanical motion into sound waves. You can build paper speakers at home with supplies from your local hardware store. Grab yourself a roll of aluminum foil, some construction paper, tape, scissors, a magnet, and a few other odds and ends, and you can be designing your own flexible speakers in no time. The process can be refined as desired, with techniques ranging from home craft to high tech depending on what you want and the availability of materials.

Paper speakers employ the same basic physics behind any electromagnetic audio speaker. As seen in Chapter 3, in an ordinary speaker, coils of wire are formed into a hollow coil that can move freely in and out of a hole in a magnet. The coil is affixed to a cone, usually of paper or plastic. The audio signal passes through an amplifier to the coil; the current of the alternating voltage creates a fluctuating magnetic field in the coil that interacts with the fixed field of the central magnet to push and pull the cone out and in, sending sound vibrations through the air to your ears. In this chapter we'll apply these 3D principles to 2D surfaces. This kind of speaker is not as efficient as a "normal" speaker, so if you're looking for *loud*, try something else. But if you want to disperse sound in different way, paper speakers might work for you. And they present inherently visual artistic possibilities that conventional speakers do not.

MATERIALS: FOIL, MAGNET, AND PAPER

Instead of ordinary wire, we need conductive material that is flat, flexible and thin and can be applied to the surface of paper. Metal foils made of aluminum or copper are great for this purpose because they can carry an audio signal well with little resistance and can be arranged in a wide variety of patterns and designs.

Commercial-grade aluminum foil is about as thin as common foils get, roughly 0.6 mils (1 mil is 0.001 inches or 0.0254 millimeters). "Heavy-duty" aluminum foil is a bit thicker at 1.3 mil. I would not recommend using any foil greater than 2 mil—for the sake of both your cutting tools and your hands (those edges are sharp, careful!). With copper, I suggest 1 mil foil, the thinnest available.

For maximum sound output, use the strongest magnets you can find. Neodymium rare-earth magnets are much stronger than ceramic or clay magnets and can be found at any well-stocked hardware store (thanks to widespread use in computer drives, neodymium magnets have become remarkably inexpensive).

The choice of material used for the vibrating surface is important, too—some materials work better than others. In a paper speaker, the surface corresponds to the cone of a traditional speaker. Firm, taut membranes are good for propagating vibrations to the air and pushing sound across a room; thick and floppy ones are not so good. You can test a material's suitability by flicking it with your finger and listening for the sound it makes. You'll notice that floppy material won't make much sound while sized papers, vellum, and plastic film will "ping" slightly, like a quiet drum. Poor acoustic materials include tissue paper, fabrics, and wood (unless it's thin veneer).

PREPARING THE MATERIALS

You can build a simple flat flexible inductive circuit with copper or aluminum foil tape available at hardware stores (this tape is commonly applied to windows for intrusion alarms and apparently used as a ground border to keep snails out of gardens). Apply the tape to the paper so that it makes flush contact with a good strong magnet or set of magnets (Figure 9.1). Circular patterns around a circular magnet work well or straight lines around a rectangular one. Direct contact between the path of the tape and the magnet creates the physical vibration that produces sound. Make sure you have an "in" and "out" end of the tape to hook up to your amplifier.

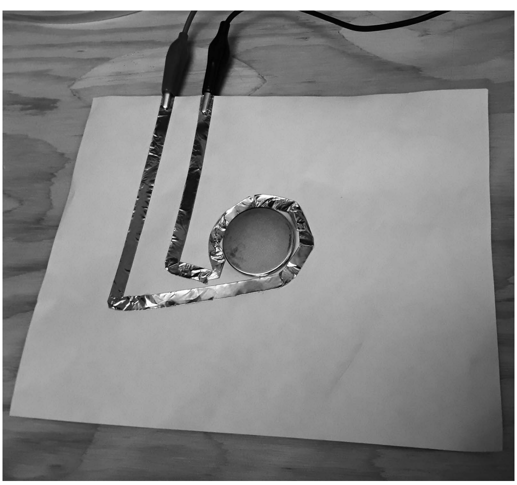


Figure 9.1
Paper speaker using copper foil tape and neodymium magnet on paper.

An alternative method, which takes a little more work but offers a wider range of possibilities, is to apply foil onto a low-tack adhesive backing such as frisket film (which can be found in any art store). Roll the foil over the backing, applying an even, moderate pressure. This ensures that the foil adheres to the surface evenly, without too many wrinkles (it can take a bit of work to perfect your technique). Once you've got your foil on an adhesive backing, you have a conductive foil sheet out of which to create any flat flexible circuitry you could desire. You can cut out your design from this surface with scissors and apply the design to paper directly or cut the design with a craft cutter.

USING A CRAFT CUTTER

Cutting a labyrinth by hand can seem to take forever, and foil tape has limited design possibilities. A more effective way of cutting out patterns uses a craft cutter. The craft cutter works just like a printer, except instead of a printhead, it has a blade that will cut into flat material. Like a printer, you feed it a digital file of a graphic image. You can design and create your own circuit pattern using either the software that comes with the cutter or a graphic design program such as Adobe Illustrator. The cutter cuts out the appropriate pattern from whatever you feed into the machine. They are designed for cutting paper and vinyl, but with a few tweaks you can use them for cutting your speaker foil.

Craft cutters come with instructions for use, and you'll need to make some slight adjustments in order to cut foil. Usually, the cut-depth and strength settings need to be near the very low end of the range so the blade will cut through the foil without cutting through the backing. Make sure you have a fresh, sharp blade in the cutter.

Take your foil sheet adhered to backing and tape it down to the cutting mat (cutting mats come with most cutters, but you can also just use thick paper). Feed your foil face-side up into the cutter so that the cutter can cut your pattern onto the foil (Figure 9.2).

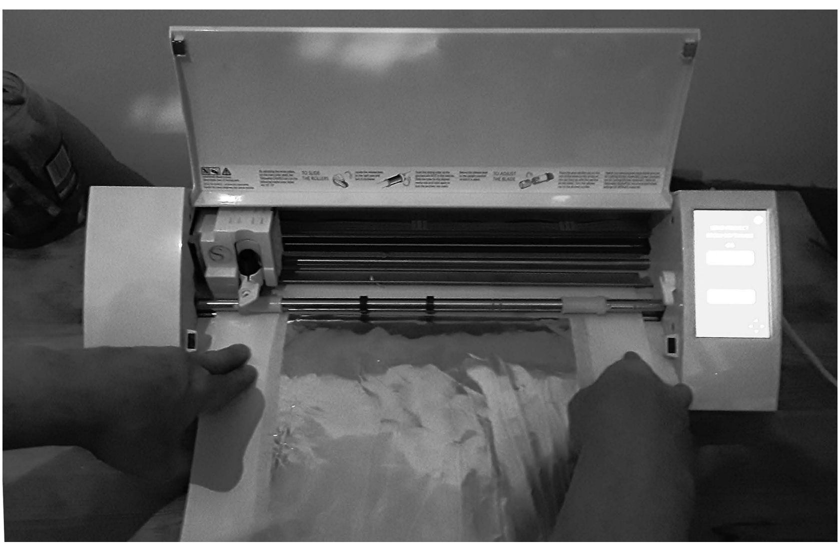


Figure 9.2 Feeding a foil sheet with adhesive backing, placed on a white cutter mat, into the craft cutter.

COMBINING IT ALL TOGETHER

The craft cutter will cut out your design, but you still need to remove the unwanted bits of foil from the adhesive backing—a process known as "weeding." A dull X-Acto blade works well for this. Just be careful not to cut accidentally into bits of the design you want to keep.

Once the weeding is done, you can affix your design to the paper surface. Spray a bit of standard spray adhesive on the paper. Carefully place your pattern, foil-side down, onto the paper. Then peel off the low-tack temporary adhesive backing, and you're left with a permanently adhered circuit. Now you can see why a low-tack backing is essential: it needs to be *less* adhesive than the permanent surface, otherwise peeling it off will be difficult.

Next you need to adhere magnets to your paper sheet. In an ordinary speaker, the magnet and coil are separate, but for paper speakers, everything lies on the same paper surface. Instead of vibrating a cone coupled to the coil, here the electromagnetic field shimmies the paper that holds both the magnet and the circuit foil. The strength of this vibration depends on the strength of the coil's field and the fixed magnets—multiple magnets often make the speaker louder.

The taping technique can yield interesting speaker geometries. Figure 9.3 is an example of a speaker made with a craft cutter: just a single circuit of copper foil weaving around in a circular pattern, with the magnets on the back held in place with a sheet of adhesive film. They fit into the "circles" of the circuit design to maximize edge-to-edge connection with the copper foil.

THE ELECTRICALS

Now it's time to send some audio into your speaker. It's almost impossible to solder to aluminum, so if you are using aluminum tape for the patterns, use clips leads to make the connection from speaker cable to the tape; copper tape can be soldered, but do so

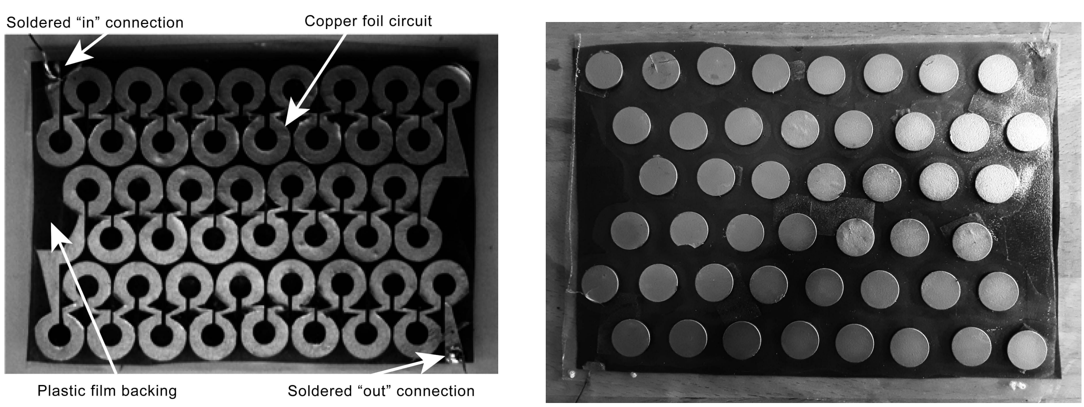


Figure 9.3 Both sides of a copper foil circuit on plastic film, showing magnets in place.

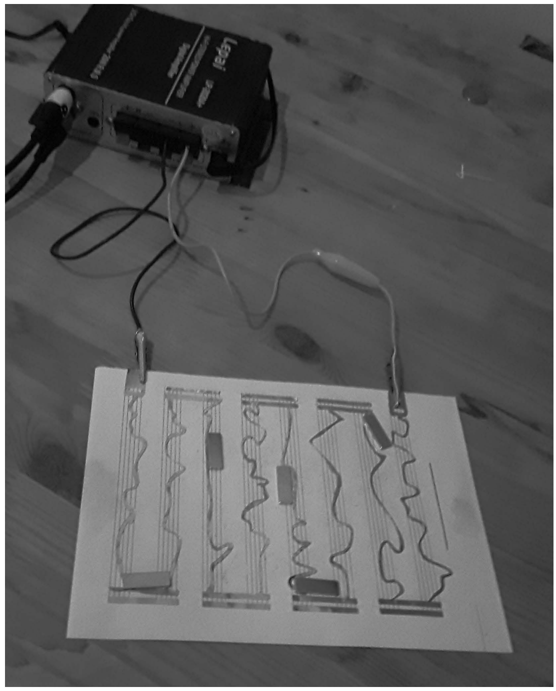


Figure 9.4
Hookup for a paper speaker with rectangular magnet accents.

quickly: you don't want to set the paper on fire. We can use the same amplifier (with speaker terminals) from our transducer experiments in the previous chapter. You hook up your paper speaker just like you would a "normal" speaker in a component stereo system. Connect the in (+) and out (−) of your amplifier to the two ends of your foil pathway (Figure 9.4). Keep in mind that the resistance along the circuit might not match the rating for the amplifier you use. A small circuit of foil can have extremely low resistance (around 1 Ohm or less) while most amplifiers are rated for 4–8 Ohms. Effectively, such a low resistance acts like a "short circuit." With this in mind, try placing a very small resistor (4–8 Ohms) in series with your foil circuit, and you will be able to turn up the amplifier and get a stronger signal without fear of damaging the amp. (The resistor should be rated at 5 watts or greater to prevent it from overheating.)

MAKE IT INTERACTIVE

Instead of gluing a magnet to the paper, you can hold the magnet and move it over the speaker surface to vary the sound. A band around your hand with a magnet adhered makes performance easier. Place it very close—or on—the surface of your paper speaker circuit, turn up the volume, and roam the surface to change the sound.

You'll notice that the vibration is louder in some locations than others. The strongest part of a magnetic field—for both the permanent magnet and the foil—is at its edges: placing the edge of your magnet against the edge of the circuit will yield the strongest response. The strength of the interaction weakens with the distance between the magnet and the tape traces on the paper. Exploring the surface with a magnet can yield unexpected results from the wily electromagnetic pathways you have created in your circuit.

FOUND MATERIALS FOR YOUR PAPER SPEAKER

Found materials (the soul of hacking) can be great. Other conductive materials can add flair and style to your paper speaker. Chocolate bars are usually wrapped in aluminum foil. Hershey's don't work well for this purpose, since the foil wrapper is extremely thin, but high-end chocolates often come in tinted aluminum already applied to a paper surface (and, as a further bonus, contain tasty ingredients).

There are also colorful conductive aluminum papers and foil origami sheets available at craft or art stores. Shopping can be tricky, though, since only some foil papers are conducive, while most use metallic plastic, which looks promising but will not conduct at all. Bring a multimeter when you go shopping and test the foil with the resistance setting—conductive material should read less than 1 Ohm between any two points on the surface. Caveat emptor, but you might get lucky.

WEARABLE AND WIRELESS

Besides paper, other flat, flexible materials can make for good speakers as well. As mentioned at the start of this chapter, you need a surface that propagates sound well. Most fabrics are, sadly, poor at this. But sew-on patches (not too thick), ties, and scarves can sometimes work. Figure 9.5 shows a speaker tie. If you tried to make a Windsor knot with this, it would mangle the copper foil circuit—walk that line between classy and tacky by wearing a clip-on tie. Of course, for most wearables, you still need to worry about the amplifier and sound input, but it's easy enough to hack a Bluetooth speaker for parts. Figure 9.6 shows a scavenged circuit from a Bluetooth speaker used for the

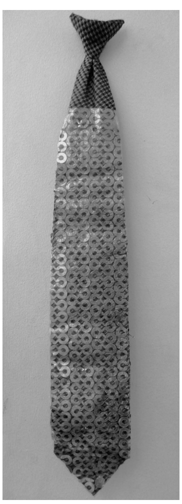


Figure 9.5 Clip-on tie speaker

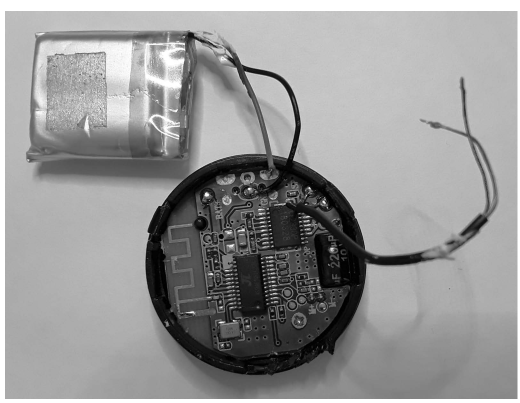


Figure 9.6 A scavenged Bluetooth speaker circuit with battery (top left) and connections to speaker (wires on the right).

back of the tie: break open a Bluetooth speaker (carefully) to extract the goodies from inside. The guts usually include a Bluetooth receiver, an amplifier, a rechargeable battery, and wires to connect to your speaker.

ONWARD!

The best way to discover is to explore and test. It can be difficult to know which circuits, which magnets, and which materials will work the way you want them to, and the only way to find out for sure is to try them out. But messing with paper speakers opens up all kinds of new avenues for discovery, not just for speaker building, but for ways of spatializing sound and integrating audio electronics into visual design. Remember David Tudor's sage advice: a loudspeaker isn't just a hole for sound to come through, it's a musical instrument. Or an art object.

CHAPTER 10

Tape Heads
Play Your Credit

You will need:

- An expendable tape recorder/player or a loose tape head.
- Some magnetic media: cassettes, reel-to-reel tape, transit cards, credit cards, etc.
- A battery-powered mini-amplifier with considerable gain.
- An additional sound source, such as a laptop or phone.
- Optional: a surplus credit card reader.

Even in the age of smartphones and Spotify, there's a lot of data sitting around in magnetic particles: music and phone messages on cassette tapes, personal data on your credit card, virtual money on transit cards. Whereas some of us still remember what a cassette sounded like as we drove down the interstate, it's not often that we get to *hear* the information on other magnetic storage media. All it takes is a tape head and an amplifier.

A tape recorder translates audio signals into a fluctuating electromagnetic field through a tape head, the small metal Brancusi-esque object inside the well of a cassette player (Figure 10.1). The tape head's undulating magnetism in turn aligns tiny magnetic domains in the iron-like powder covering one surface of the recording tape, as if they were midget compass needles. When the tape is played back, this process reverses: the varying magnetic orientation retained by the mini-magnets on the tape now induces tiny currents inside the tape head that, when greatly amplified, resemble pretty closely what went into the tape recorder earlier—another instance of the *reversibility* of electromagnetism discussed in Chapter 4. It's not so different from translating sound vibrations into grooves cut into a record's surface, to be traced later by a needle whose wiggling is transformed back into sound waves—only with tape it's magnetic fluctuations instead of shimmying grooves. Certain digital media—such as hard drives, credit card stripes, and old-fashioned floppy discs—are like cassette tape, only cruder: the magnetic domains just flop back and forth between two states, 0 and 1, instead of tracing the nuanced contour of an analog waveform.

Figure 10.1
Tape heads.

ART & MUSIC 6

TAPE

Although invented for straightforward recording and playback of speeches in the service of the Third Reich, and largely remembered today for its more benign role in the rise of the record industry, magnetic tape has proven to be a wonderfully flexible *performance* medium as well. Composers such Pauline Oliveros (*1 of 4*, 1966), Steve Reich (*Come Out*, 1966), Terry Riley (*Rainbow in Curved Air*, 1969), and Alvin Lucier (*I am sitting in a room*, 1970) have all made pieces derived from the properties of tape loops and tape delays. When the tape is taken off the reels, it becomes surprisingly instrumental. In 1963 Nam June Paik, on the threshold of his transformation from composer to video artist, attached dozens of strips of prerecorded tape to the wall of a gallery in Wuppertal, Germany, and invited the visitors to play it back via handheld tape heads. According to legend, John Cage once did a similar thing in reverse: he fully covered a tabletop with blank tape and invited the public to scribble across it with tape heads attached to pencils through which electronic sound was playing; at the end of the evening, the tape was wound onto a reel and played back for all to hear.

Laurie Anderson's Tape Bow Violin (built in 1977 in collaboration with Bob Bielecki) substitutes a tape head for the violin's bridge and a strip of tape for the hair of the bow; the tape contains a recording that Anderson plays backwards and forwards as she draws the bow across the head (Figure 10.2). "I began to work with audio palindromes, words that produced different words when reversed. Audio palindromes are not predictable like written palindromes ('god' is always 'dog' spelled backwards). With a lot of experimentation I produced songs for 'The Tape Bow Violin' that could be played forwards and backwards."[1]

Years later, César Eugenio Dávila-Irizarry glued recordings of percussion instruments onto the body of the gourd typically used to make a güiro (a percussion instrument consisting of a gourd scribed with notches that are

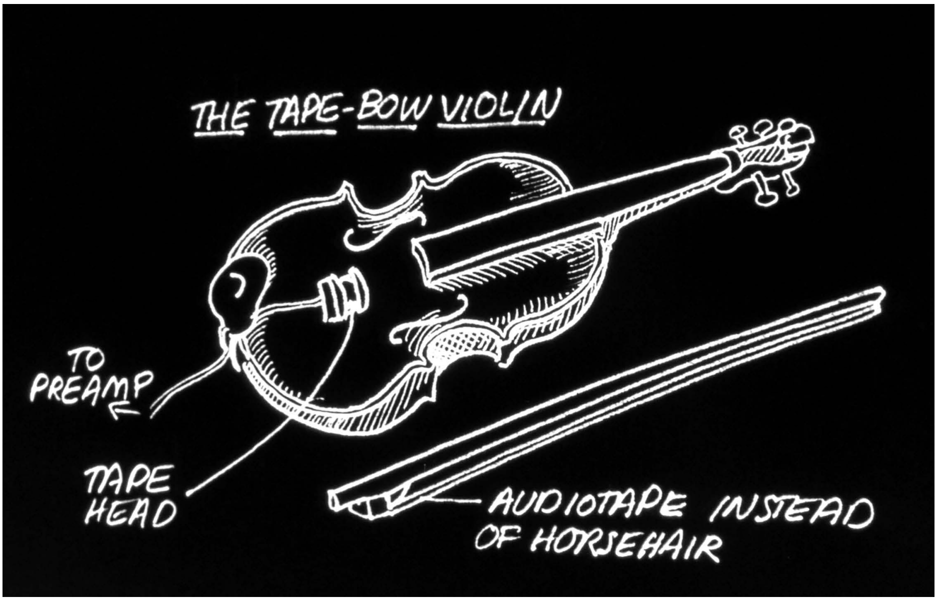


Figure 10.2 Laurie Anderson, Tape Bow Violin.

Figure 10.3 César Eugenio Dávila-Irizarry, Tape Güiro.

scraped rhythmically with a comb-like raspa); his new instrument is played with a handheld tape head (Figure 10.3). In the installation version of Mark Trayle's *¢apital magnetics* (1999), visitors insert their credit cards into what appears to be an ordinary ATM; Trayle has programed an internal computer to generate short musical compositions based on the data on each card, heard through speakers embedded in the ATM.[2]

PREPARATION

The easiest place to find a tape head is inside a broken or otherwise unwanted cassette player. If you have a functional boom box or other device that records as well as plays back tape, skip ahead to the "Recording" section that follows; if you have a working Walkman or other playback-only device, skip to "Playback." Many web-based electronic surplus stores sell individual tape heads or credit card data readers at reasonable prices.

The advantage of pulling a tape head from a dead player is that audio wiring is attached, often in the form of a short shielded cable; in this case cut the cable near the circuit board so as to leave as long a section as possible attached to the tape head. Other times the head will be connected to a circuit board through a translucent flexible band, which we will discard.

The back side of a stereo tape head will have four connections (as shown in the examples in Figure 10.4); a mono head will have two. If the head has a cable attached, each pin will probably be attached to a separate fine wire in a multi-conductor shielded cable, and the shield might be affixed to the metal shell of the head. Trace the free ends of the wires back to the pins on the head (often the individual wires will probably be color-coded, simplifying your job). When wiring this cable from a *stereo* head to a *stereo* plug, solder "A" to the tip of your plug, "C" to the ring, and "B" and "D" and the shield to the sleeve; when wiring a *stereo* head to a *mono* plug, solder "A" and "C" to the tip of your plug and "B" and "D" and the shield to the sleeve. When wiring a *mono* head to a *mono* plug, connect "A" to the tip and "B" and shield to the sleeve (Figure 10.4).

If the tape head arrives unwired, solder directly to connecting pins on the head, to the shield, and to the tip of the jack, following the aforementioned routing instructions. To minimize hum use shielded cable and solder an additional connection between the shield and the metal shell of the head if possible. But bear in mind that tape heads are very hummy things by nature, and some noise is inevitable (in fact, you can substitute a tape head for a telephone tap coil to pick up electromagnetic fields, as described in Chapter 6).

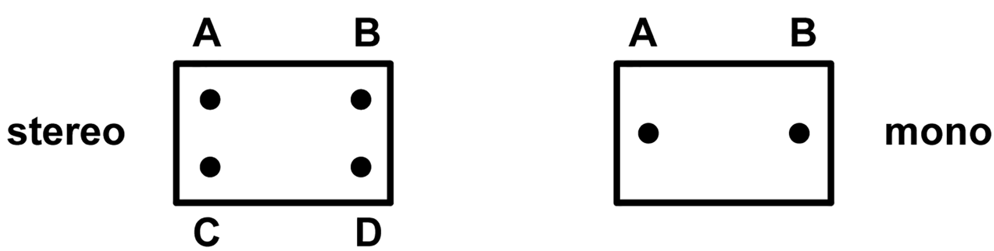


Figure 10.4 Wiring orientation for tape heads.

PLAYBACK

If your tape head is inside a functional tape player, remove it from the player without disconnecting the cable if possible. You will probably need to extend its wiring with enough scrap shielded cable that you have room to move the head freely. Either cut the existing wires in half and splice in some additional cable or de-solder the head's wiring at the circuit board (make careful note of which wire goes where!) and solder the extension cable between the board and the pigtails attached to the head. In either case make sure you reconnect the matching ends.

Press the "play" button on the tape recorder (plug in headphones if it doesn't have built-in speakers). If you are working with just a loose tape head, plug it into a high-gain amplifier or the microphone input of a mixer.

Now rub the head over some recorded media: transit cards and credit cards, eviscerated cassette tapes, computer disks. If you're using cassette tape, stretch the audiotape across a sheet of cardboard, a tabletop, or some other flat surface and fasten it down with the sticky kind of tape at either end. You will notice that one side of the tape (the emulsion side) will be *much* louder than the other (backing). Digital data (credit cards, transit cards) tends to make a much louder sound than audiotape, and one that often sounds curiously like turntable scratching (see "Card Readers," following).

Sometimes the hum will increase when you touch the head—you can minimize this by wrapping it in plastic electrical tape. If you find it awkward to handle the tiny tape head, you can lash it to the end of a popsicle stick or pencil with some electrical or gaffing tape or solder it to a metal fingerpick (Figure 10.5).

RECORDING

You can try *recording* with a handheld tape head as well. Stretch cassette or reel-to-reel tape over a tabletop as before, making sure the emulsion side is up. If you are working with a loose tape head, plug a sound source into the input of an amplifier and connect the tape head to the speaker or headphone output. While playing the sound source, move the tape head over the tape surface—keep the head in close contact with the tape. After a while stop recording and try playing back the tape—either by amplifying

Figure 10.5
Tape heads mounted on fingerpicks, Nicolas Collins.

the head while moving it by hand across the surface (as before) or by reloading the tape into a cassette (or onto a reel) and playing it back on a tape recorder. Sometimes this works and sometimes it doesn't, so don't be too disappointed if you are unsuccessful.

(The student ID cards at the School of the Art Institute of Chicago, where I teach, also serve as debit cards for use in on-campus vending machines. One day I came back to my hacking class after a break to discover two of my students trying to "copy" money from one card to another: they carefully moved two tape heads along the magnetic stripes on the two cards, one connected to the input of a small amplifier, one to its output. Sadly, the experiment was not successful.)

You can get much better results if you start with a functional boom box or cassette recorder that you're willing to sacrifice on the altar of the weird. Disassemble the device to the point that you can carefully remove the record head from its mount in the cassette well. You will probably need to extend its wiring, as described earlier in the "Playback" section. Mount some scrap cassette or reel-to-reel tape on a flat surface. Connect a signal to the boom box input or tune in its radio. Insert a blank cassette in the well, or press that little prong with your pinkie, in order to enable the record function; press the "record" and "play" keys to start recording. Move the head smoothly across the scrap tape. Press "stop," then "play," turn up the boom box volume, and retrace your movements over the tape—you should hear the original signal, altered by the inconsistencies in speed and smoothness between your two passes. You can speed up, slow down, and reverse your original sounds by changing the speed and direction of your playback motion. Or you can reload the tape into a cassette shell for playback on any cassette machine.

Although a bit awkward, this method of recording and playing back tape by "scratching" offers a cheap, highly performable alternative to cutting dub plates.

CARD READERS

Online retailers of electronic surplus sometimes sell "card readers" from ATM machines, credit card terminals, etc. The reader consists of a tape head inside a housing that guides the card smoothly past it, along with circuitry needed to decode the digital data (Figure 10.6). Stealing credit card data is advanced hacking (see Art & Music 6, "Tape"), but for our immediate purposes you can discard the digital circuitry, wire the head up as described earlier (Figure 10.4), plug into an amp, and end up with a very nice instrument for "scratching" cards.

Figure 10.6
Nicolas Collins, Scratchmaster card reader. From the collection of Ted Collins.

NOTES

1. Laurie Anderson, *The Record of the Time*. Lyon: Musée Art Contemporain Lyon, 2002. The Tape Bow Violin can be heard on "Three Walking Songs (for Tape Bow Violin)", "Sax Solo (for Tape Bow Violin)", and "I Dreamed I Had To Take a Test . . . " from *United States*, Warner Brothers LP, 1984.
2. Mark Trayle, "Free Enterprise: Virtual Capital and Counterfeit Music at the End of the Century." *Leonardo Music Journal* Vol. 9 (1999). Pp. 19–22.

CHAPTER 11

Electret Microphones
Binaural on a Budget

You will need:

- An electret microphone element.
- 8 feet of lightweight shielded cable.
- An amplifier, audio recorder, or mixer.
- A plug to match the jack on your amp, recorder, or mixer.
- 9-volt battery and battery hookup clip.
- A 2.2 kOhm resistor
- A capacitor c. 0.1 uf.
- Packaging supplies as needed: heat-shrink tubing, soda straw, and Altoids tin, along with a few additional connectors.
- Hand tools, soldering iron, and electrical tape.

We've saved the most "normal" form of microphone for last: after backwards speakers, coils, contact mikes, and tape heads, we finally get around to your basic hear-my-song mike. From any number of sources one can buy, quite cheaply, high-quality electret condenser microphone elements (Figure 11.1). You can also scrounge them from cell phones, boom boxes, and even some toys. These are the basic building blocks of recording microphones that can sell for several hundred dollars. All that stands between your $2 purchase and a pretty good mike is a handful of cheap components, a soldering iron, and some ingenuity.

If you have any choice when shopping for elements, download some data sheets first and look for the models with highest signal-to-noise ratio and a flat, extended frequency response. If you have a choice between *cardioid* (directional) and *omnidirectional* pickup pattern, bear in mind that the omnidirectional microphones often have a smoother response curve—the quality of sound often more than compensates for the lack of directionality (many purists in the field of classical music extol omnidirectional microphones).

You may recall that when we transformed a speaker into a microphone in Chapter 4, all we had to do was connect the terminals to an amplifier—electromagnetism did the rest, miraculously converting acoustic sound waves into electronic waves. This is how a *dynamic* microphone, such as a typical PA mike, works. Electret microphones, on the other

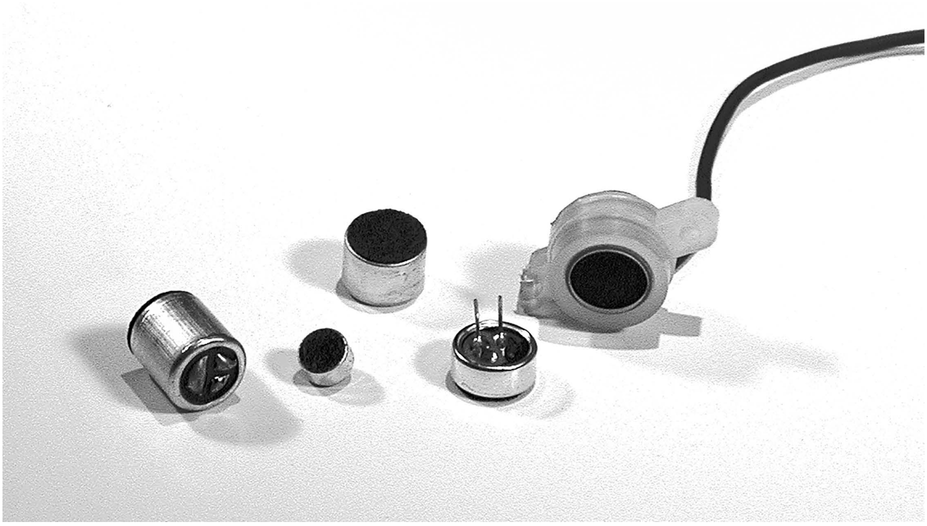


Figure 11.1 Some electret condenser microphone elements.

hand, do not contain magnets and coils of wire; they operate on even spookier *electrostatic* principles whose theory I'll skip over for now (you can find explanations online). The critical thing for you, the hacker, to understand at this point is that electret microphones require a small amount of externally applied voltage, typically from a battery, in order to function. Most electret elements use some form of what is known as "phantom powering" to send this voltage to the mike along the same wires that the mike uses to send its signal on to your mixer, recorder, amplifier, etc.

Figure 11.2 shows the connections for a typical two-wire electret microphone. You will notice that the capsule has two pads to which wires can be soldered: one,

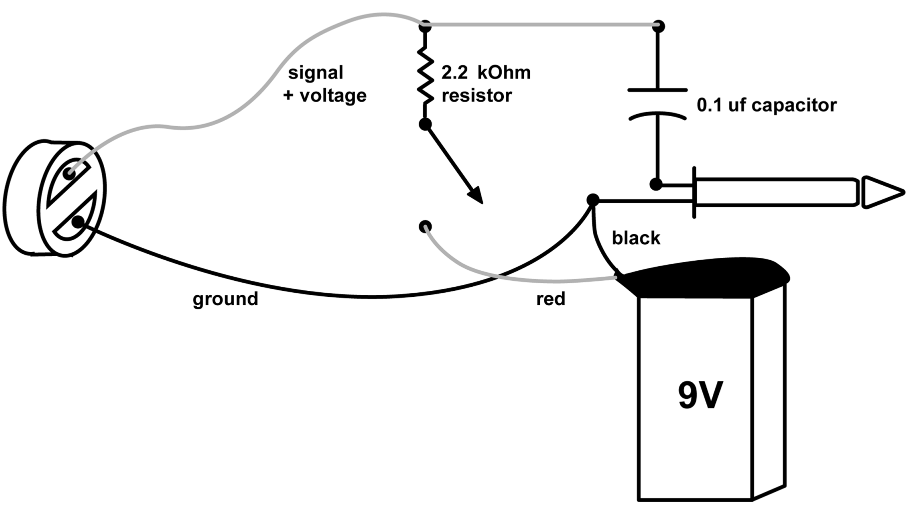


Figure 11.2 Basic electret microphone wiring.

marked "signal, + voltage" in the diagram, connects to both the positive terminal of the battery and your audio input (via the tip of a plug)—with some parts in between; the other, marked "ground," connects to both the negative terminal of the battery and the shield of the audio plug. The pads on the actual electret element are not labeled, but you can usually identify the ground terminal by the presence of thin traces connecting it to the metal shell of the capsule (data sheets come in useful here as well).

The positive terminal of a 9-volt battery (the red wire) connects to the mike through a resistor with a value of around 2.2 kOhm. A capacitor (around 0.1 uf) blocks this voltage from entering your amplifier or mixer. You can buy these two parts very cheaply at any electronics store or online retailer (we'll discuss more about how these parts work in later chapters—for now you can take them on faith). An optional switch turns the battery on and off—the microphone drains very little current, but the switch (or removing the battery when not in use) will extend the battery life from months to years. It's best to turn the mike on *before* connecting to your other audio equipment to avoid big thunks.

Solder on whatever plug matches your recorder, amplifier, or mixer—I've included in Figure 11.3 the connections needed if you want to use an XLR connector to plug into the microphone input on a mixer or a professional field recorder. (Please note: if you connect this XLR-equipped electret design to a mixer's microphone input, make sure the mixer's internal phantom power is switched *off*; powering an electret capsule off a mixer's phantom power—typically 48 volts—requires a different, more complicated circuit.)

Electret microphone elements can usually be powered by a wide range of voltages. I've indicated a 9-volt battery in the figures, but two 1.5-volt batteries are usually sufficient—you can use AA, AAA, or even those tiny, absurdly overpriced button cells. Sometimes a single 1.5-volt battery will suffice. If a data sheet is available from the manufacturer, it will tell you the safe voltage range; otherwise, you'll have to experiment.

Occasionally one finds electret elements with three wires instead of two. Instead of combining signal and battery power on a single wire, one wire will be designated as the signal and will connect directly to the tip of the plug (or pin 2 of the XLR); another will be labeled "power" and will connect to the "+" of the battery; the third connects to both the connector ground and the battery's "−" terminal (Figure 11.4). This microphone element requires shielded cable with *two* internal conductors (plus the shield) rather than one.

These diagrams show you how to make the electrical connections needed to get sound out of the mike but leave you holding an unwieldy rat's nest of wire, batteries, and loose components. Mechanical packaging can be trickier than the basic electronic connections. Where do you put the various parts?

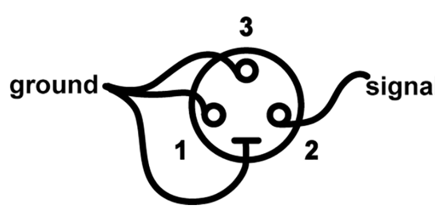


Figure 11.3
Wiring an XLR microphone connector.

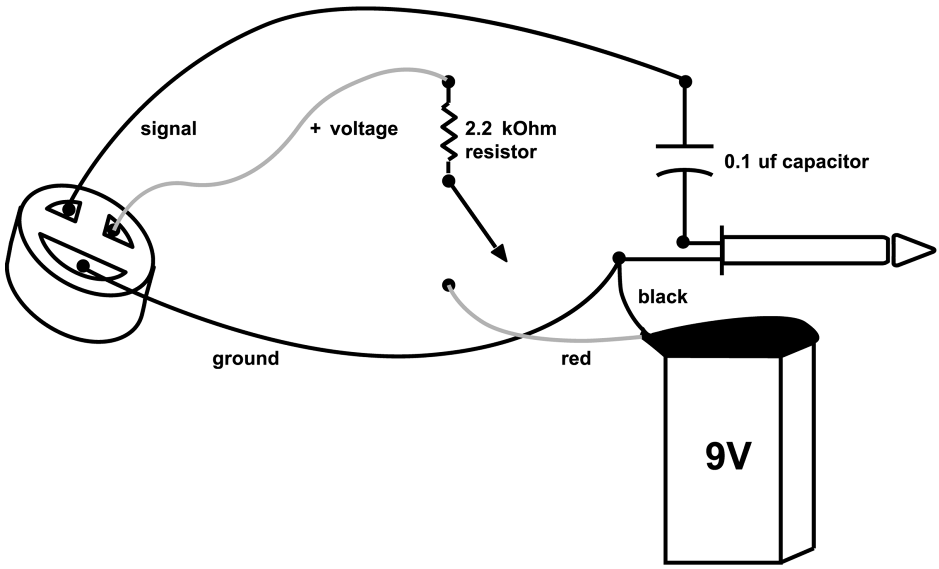


Figure 11.4 Wiring a three-wire electret microphone.

One solution is to run some shielded cable from the electret element to a remote power supply, in the form of a small box containing the battery and related components (another excellent use for abandoned Altoids tins), and then run another cable from this box to whatever plug you are using to connect to the rest of your audio system. You can simply pass the wires through holes in the box, or you can install an additional set of jacks (Figure 11.5).

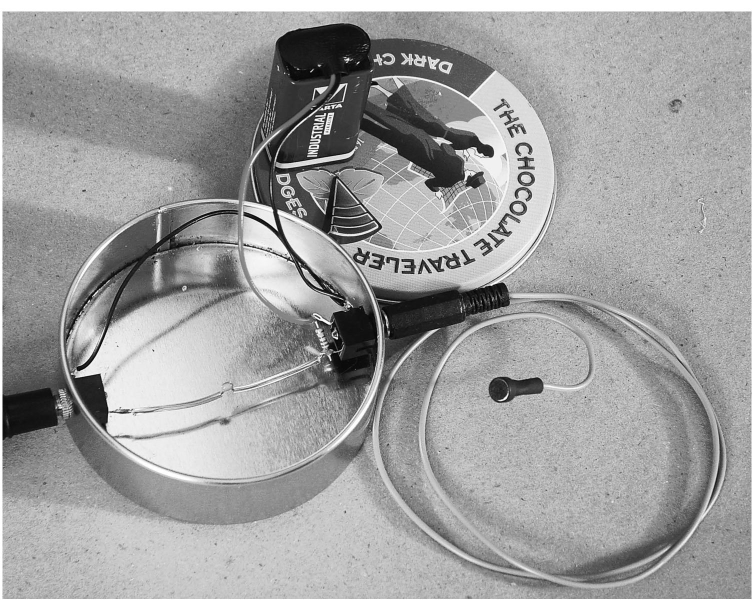


Figure 11.5
Remote power supply for electret mike.

The electret capsule itself can be packaged in a number of ways:

- Just squirt a blob of silicon caulking on the back side of the electret element; after it dries clip the electret to your shirt with a clothespin and use it as a lavalier microphone for interviews;
- Glue it into the end of a plastic drinking straw, with the cable running through the straw and out the other end;
- Remove the hood covering the alligator clip in a busted test lead (I'm sure you have a pile of them by now) and slip it over the electret element (Figure 11.6).
- Encase it in heat-shrink tubing (as shown in Figures 11.5 and 11.7). Heat shrink comes in a range of diameters and can usually be found at ordinary neighborhood hardware stores. Select a diameter a bit wider than the capsule, cut a piece an inch or so long, and slip it over the mike so that one end is flush with the front surface of the capsule. Aim a blow dryer or heat gun at the tubing and it should shrink down to fit as snugly as the sweater on a Littleneck clam (if it shrinks down over the front of the mike, you can trim it with a razor blade).
- Mount it in the same box as the power supply: drill a hole the diameter of the capsule, secure the capsule with epoxy or silicon caulking, build in the remaining components, and run a cable out of the box to a plug. This is a good way to approximate a pressure zone microphone (PZM).
- Use small PVC plumbing parts to make something that resembles a handheld microphone and build the power supply right inside.
- Use your imagination.

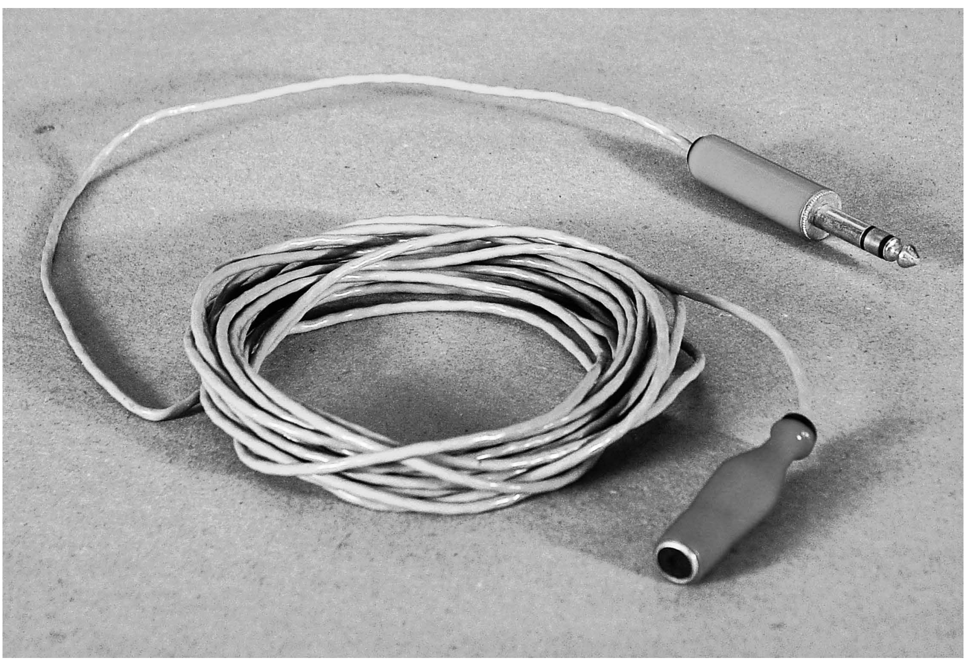


Figure 11.6 Electret capsule in clip lead hood.

The issue of packaging is not merely aesthetic: the shape and mass of the material can affect the frequency response and directionality of the microphone. You should "ear test" various options. The challenges of the mechanical design process may increase your tolerance of Neumann's mark-up.

"PLUG-IN-POWER" AND STEREO MICROPHONES

As mentioned prior, the 48-volt phantom power available on many audio mixers is too high for these little mikes to use directly, but a lot of semi-professional audio video equipment (such as flash recorders and video recorders) provide a form of 5-volt phantom power on the 1/8-inch (3.5 mm) stereo jacks used for connecting external stereo microphones. Plug-in-Power (PiP), as it is known, can be used in lieu of the remote power supply discussed prior—Figure 11.7 shows two electret elements connected to a single stereo plug to make an inexpensive stereo microphone for use with this low-voltage phantom power (refer to Figure 7.17 in Chapter 7 for a clear view of the wiring). The power supply shown in Figure 11.8 emulates Plug-in-Power if you want to use this mike design (or any commercially available stereo mike compatible with Plug-in-Power) with devices that do *not* provide the 5-volt phantom. Note that this power supply requires no on/off switch since the battery drains only when the mikes are plugged in. "+ voltage" is unspecified on the schematic: a 9-volt battery will work for most electret elements; you can use three AAs or AAAs if you prefer to conform to the 5-volt "industry standard."

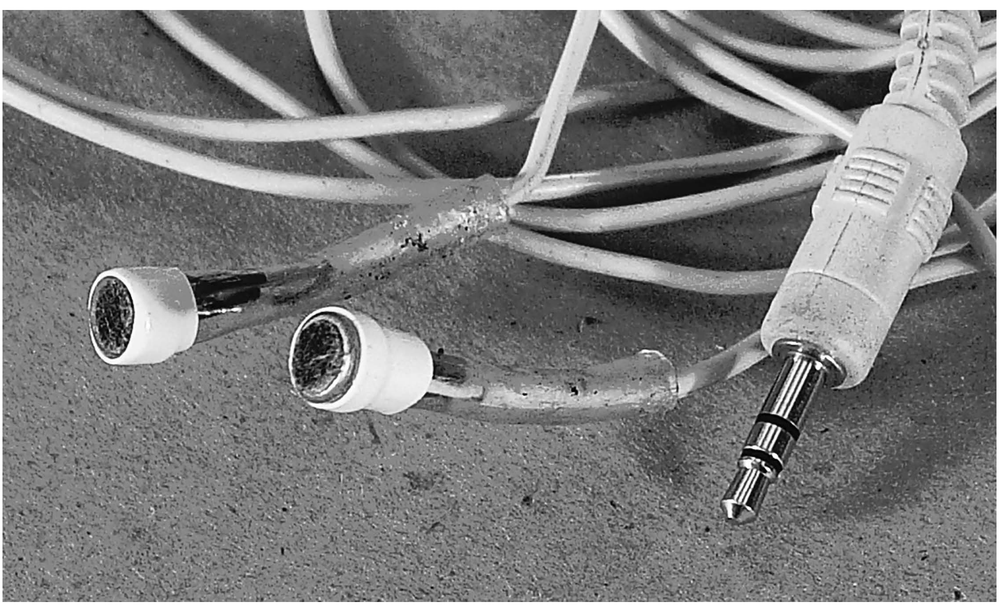


Figure 11.7 Two electret capsules wired as a stereo pair for use with Plug-in-Power.

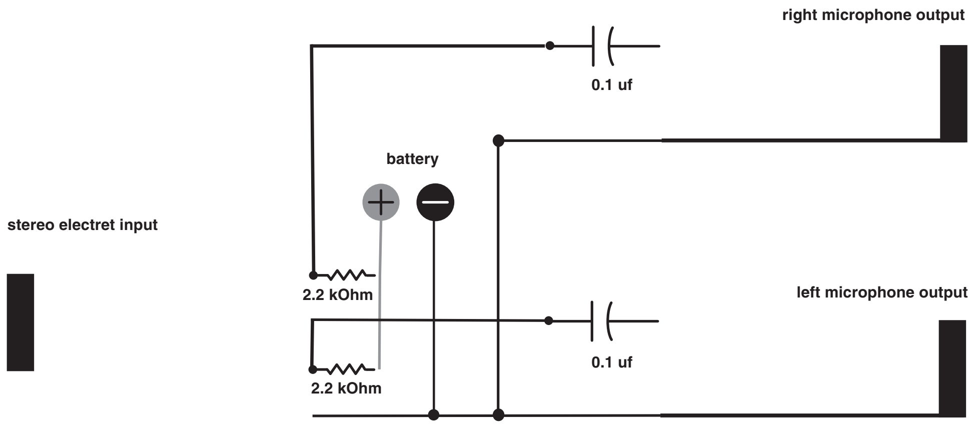


Figure 11.8 Power supply providing Plug-in-Power phantom voltage for any compatible microphone.

APPLICATIONS

"Binaural recording" is a microphone technique optimized for playback over headphones: two small microphones are mounted in the ear canals of a specially designed dummy head or worn like earbuds in the outer ear of the recordist; when the recording is played back over headphones, there is great 3D spatial realism because the headphones are in the same spatial relationship as the mikes were. Since more and more music is being heard through earbuds and headphones these days, consider experimenting with this recording technique. You can glue two electret mikes to a pair of ear plugs or headphones (but look out for feedback if you monitor through the headphones while recording). Alternatively, if you can find the head from a mannequin, drill holes where the ear canals would be and glue in two electrets (be prepared for odd looks when you carry the head into a concert hall or bird sanctuary).

You can mount a microphone at the focus of a parabolic reflector (a satellite dish or kid's snow saucer work fine) for a hyper-directional mike for wildlife sound recording. Embed the elements in musical instruments—they're great inside accordions and melodicas and in mutes for trumpets and trombones. New Zealand musician David Watson dropped one inside his bagpipes (see his video clip on the website). You don't even have to blow into the instrument: drop one into the mouth hole on a flute, aim it at a speaker, wiggle your fingers on the keys, and let feedback do the rest. Glue electrets and small speakers at opposite ends of PVC pipes of different lengths, patch them through small amplifiers, and build a feedback organ (see the video by Valve Membrane on the website); telescoping PVC tubing makes a feedback trombone. Hang one in your shower, put a speaker in the sink, and turn your bathroom into a reverb chamber (shades of Phil Spector).[1]

Electret mikes are cheap enough to use in situations where you wouldn't risk a Schoeps: sharing a beer bottle with a firecracker, for example.

Even the least expensive electret capsule has a surprising bass response—often extending to 20 Hz or lower. They are very susceptible to wind noise and breath pops;

you'll want a windscreen of some kind (scrap foam rubber will do), and you may need to roll off the bass to avoid overloading your recorder input under certain circumstances. But their low-frequency response also makes them excellent for picking up subsonic pressure waves, such as those produced by opening and closing doors, wind gusts, and some barometric changes—roll off everything *above* 30 Hz and you can record the weather. Felix Hess has made recordings of such very low-frequency barometric activity and sped them up into the audible range for our listening enjoyment.[2]

NOTES

1. An excellent guide to amplifying *any* instrument (on a small budget) is: Bart Hopkin, *Getting a Bigger Sound: Pickups and Microphones for Your Musical Instrument*. Tucson, AZ: Sharp Press, 2003. Hopkin covers making and using coil pickups, contact mikes, and air mikes and includes a very neat trick for rewiring electret elements to improve their ability to record very loud sounds without distortion.
2. Felix Hess's barometric recordings are included on a CD in *Light as Air* (see notes to Art & Music 5, "Drivers," Chapter 8).

CHAPTER 12

Laying of Hands
Transforming a Radio Into a Synthesizer by Making Your Skin Part of the Circuit

You will need:

- A battery-powered radio.
- Batteries for the radio.
- Small screwdrivers, flat and/or Phillips, as required to disassemble the radio.
- Plastic electrical tape and some stranded hookup wire may be needed.
- Optional: cigar box, double-stick foam tape.
- Your coil pickup from Chapter 6 and a monitor amplifier or mixer and speakers.

HOW TO CHOOSE A RADIO

It should be cheap enough that you won't be angry if it never works again. The AM band is more important than the FM, but it doesn't matter if the radio picks up both. It should have analog tuning (i.e., a dial) rather than digital presets or scan buttons. Older radios are usually better than newer ones—look for one built before 2000. Larger radios are easier to work with than tiny ones. Boom boxes are great, and you can use the tape head for other experiments (see Chapter 10). It's better if it has a built-in speaker, not just a headphone jack. And most importantly: **IT MUST BE BATTERY POWERED!** Beware: an alarm clock radio with a built-in "backup battery" is *not* suitable since it requires AC power to function as a radio.

LAYING OF HANDS

Install the batteries and confirm that the radio works prior to disassembly; if not functional, return it to the store. If it works, remove the batteries.

Remove the screws holding radio together. Put them somewhere safe (like a cup, not loose on top of your table), taking care to make a note of location if they are of different sizes. Some screws may be hidden beneath stickers or under the batteries. Gently separate the halves of the radio. Plastic wedge fasteners can be popped by twisting a thin flat screwdriver or clam shucker along the seams. Don't force it—check for hidden screws if it resists. Avoid tearing wires. Once open, make note of any wires connecting the two halves of the radio or the circuitry to the speaker, battery, antenna, etc. in case they break later (see the Fourth Rule of Hacking).

Remove the screws holding the circuit board to the radio housing. Carefully lift the circuit board from the case. Sometimes adhesive may be used as well as screws. Knobs and switches may intrude through openings in the case: knobs often pull straight off but may have set screws. You may need to cut some plastic to extract the circuit, but be careful: circuit boards, especially old ones, can be very brittle, so don't bend them! If the radio has a telescoping antenna, it may need to be disconnected in order to expose the circuit board—it's not needed for our purposes.

In an old-school circuit, the side of the board with most of the little bumpy colorful things (resistors, capacitors, chips, etc.) is called the "component side"; the side that consists mostly of wiggly lines (usually silver or copper colored, sometimes under a translucent green wash) is the "solder side." Turn the board so that the solder side is accessible. Remove the volume and tuning knobs if they are large enough to cover over parts of the circuit board, leaving short nubbins by which you should still be able to adjust settings—you want to expose as much of the board as possible. Replace the batteries—depending on the

Figure 12.1 Laying of hands.

construction of the case you may have to hold the batteries in place using plastic electrical tape or extend the battery leads from the discarded case with extra wire.

Turn on the radio and tune it to a "dead spot" between stations or at the end of the dial. Lick your fingertips, like a safecracker in a film noir. Touch the circuit board lightly in different places with several fingers at the same time until you find locations that affect the radio's sound (Figure 12.1). Search for touch points that cause the radio to start to whistle, squeal, or "motorboat." The moisture on your fingertips increases conductivity and makes your touch more sensitive, but I suggest you do not lick the circuit board directly. And, observing the Seventh Rule of Hacking, avoid shorting points on the circuit board with screwdriver tips, bare wire, or full immersion in drool.

(Modern radios, populated with tiny "surface mount components," have components and traces distributed on both sides. If this is the case with yours, try playing both sides of the board.)

Don't worry if you don't get new sounds immediately—it's a bit like making your first sound on a trumpet or flute or learning to ride a bike. Tune the radio to different locations across the band; rotate the board on your table and try from another angle, and experiment with different finger placement. The area near the volume control is often the most sensitive. Sometimes you have to work a while before you find a sweet spot, but then you'll lock in and form a tight feedback relationship with the instrument, and the sounds should pour forth. If you can't get anything after an hour, try another day or another radio.

WHAT'S HAPPENING?

As Ol' Sparky has demonstrated on far too many occasions, flesh is an excellent conductor of electricity. By bridging different locations on the board with your fingers, you are effectively—if haphazardly—adding free-range resistors and capacitors to the existing circuit. Your body literally becomes part of the circuit. Varying the pressure (or dampness) of your finger changes the values of these bio-components. Depending on the location and pressure, you might end up merely retuning the radio or affecting its loudness, but you could change the radio into a very different kind of circuit, like an oscillator. This happens when the output of an amplifier stage flows back through your skin into its input—voilà, feedback, the musician's friend!

A radio contains many of the basic modules of a classic analog music synthesizer: oscillators, noise generators, amplifiers, filters, ring modulators—in addition to the world's largest sample library, courtesy of your local radio stations. Your skin retunes these modules, patches them together in different ways, and adds feedback paths. Moving your fingers and changing the volume, tuning, and band selection reconfigure this synthesizer to make different sounds—a whistling oscillator, modulated white noise, a signal processor chopping fragments of radio broadcasts, etc.

You may not know exactly what you are doing (like plugging patchcords and twiddling knobs in the dark), but you'll soon acquire a sense of touch: what points work best, how pressing harder affects the sound, etc. This is a very direct, interactive sense of control similar to that which a "real" instrumentalist, such as a violinist, uses to articulate and intonate notes. This principle of direct contact with circuitry is relatively rare among commercial electronic instruments but has often been explored by experimentalists (and is the soul of the infamous Cracklebox—see Art & Music 7).

THE CRACKLEBOX

The inspiration for the Cracklebox (*Kraakdoos* in Dutch, see Figure 12.2) springs from the out-of-body experience of an adolescent Michel Waisvisz (1949–2008) playing his father's shortwave receiver by laying his hands on its 240-volt-powered circuit board. He recovered, the radio was nailed against the wall of their house in Delft to prevent further mishap, and Waisvisz's vision of an electronic instrument that could be played by intuitive touch was eventually safely realized in the late 1960s in collaboration with the engineer Geert Hamelberg. After building a number of different keyboard-sized instruments, in 1975 Waisvisz worked with engineers at STEIM, a music research foundation in Amsterdam, to design the very portable and affordable Cracklebox. Four thousand Crackleboxes were sold, and the original instrument was reissued in 1998 and can be bought online.[1]

Touch circuits had been employed in the expressive keyboard controllers of the maverick synthesizer designers Donald Buchla in 1965 and Serge Tcherepnin in the early 1970s, but the Cracklebox was the first mass-produced electronic musical instrument that incorporated the player's skin as the primary variable component in a sound-generating circuit.

Figure 12.2
The Cracklebox (Kraakdoos).

In the upcoming chapters, we will build circuits controlled more "rationally." Any time things start to sound *too* controlled, remember you can always add your body to the circuit. And if your eviscerated radio becomes too predictable, try a friend's or open another—different radios respond differently. Listen to Josh Winter's audio and the Bent Radio Orchestra in Duncan Chapman and Stewart Collinson's video on the website.

Older-style radios sometimes have tuning coils whose colorful slotted tops are just asking for the twist of a screwdriver. Doing so may interfere with the radio's ability to pick up stations but can add whooshy noise and rhythmic motorboating to your instrument's palette (see Michael Bullock's video on the website).

Generally speaking, the older the radio, the greater the range of sounds you can coax out of it. Modern radios cluster more functions onto each chip, minimizing the number of potential interconnections you can make with your skin. Older designs use more components on larger, finger-friendlier circuit boards; they often have bigger speakers with more extended bass response as well. When scavenging flea markets, thrift shops, and eBay, think 1960s (like the radio in the lower center of Figure 12.1), rather than the twenty-first century. The additional weight and mass in your touring luggage will be more than offset by its sonic splendor.

When you are through experimenting, you may want to reassemble the radio—this is the safest way to carry it around and to restore its functionality as a radio. But if you are so enamored of your electronic Ouija board that you cannot bear to seal it up again, welcome to the most hardware part of hardware hacking: finding a box. Cigar boxes work great: using double-stick tape, you can stick down the circuit board (solder side up), speaker, and related parts (Figure 12.3). Close the lid to transport, open it to play. Don't do this with metal boxes, as they may short out the circuit, but wood or plastic are fine.

A tickled radio often swoops over a very wide frequency range, but if the built-in speaker is small, you might never hear the bass end. Try sticking the coil pickup from Chapter 6 on the speaker's magnet and plug it into a better sound system. The coil will pick up lower frequencies than a small speaker will actually reproduce. Alternatively, drop

Figure 12.3 Nicolas Collins, Oliveros Radio-Cracklebox mounted inside a cigar box.

an amplified contact mike onto the speaker: listen as it bounces around, adding a percussive edge to the radio's squeals—like the bottle caps around the calabash of an mbira.

If your radio has a headphone jack, you can listen to it over headphones or connect it to a *battery-powered* amplifier—if this amp has a larger speaker than the radio, it should give you a louder, fuller range signal. Do *NOT* connect from the headphone jack into an amplifier, mixer, or recorder that connects to AC (Mains) power at the wall—use the coil-on-speaker cheap DI instead.

EXTREMELY IMPORTANT NOTE: DON'T EVEN *THINK* ABOUT "LAYING HANDS" ON ANYTHING THAT PLUGS INTO THE WALL!! AND NEVER PLUG YOUR RADIO'S HEADPHONE JACK INTO AN AC-POWERED MIXER OR AMPLIFIER UNLESS YOUR ARE 101% CERTAIN THAT THERE IS NO POSSIBLITY OF A GROUND FAULT (I.E., NEVER)!

VARIATIONS

The primary electronic function needed to transform the radio into an oscillator is *amplification*. Although the radio's filtering, frequency shifting, and chopped noise add considerable character, you can get many of the more oscillator-like effects by laying damp fingers upon a simple battery-powered amplifier circuit instead, such as those described in Chapter 1 or built in Chapter 24. Laying hands on a Walkman or boom box sometimes varies tape speed as well as circuit feedback (see Figure 12.4 and Seth Cluett's video on the website).

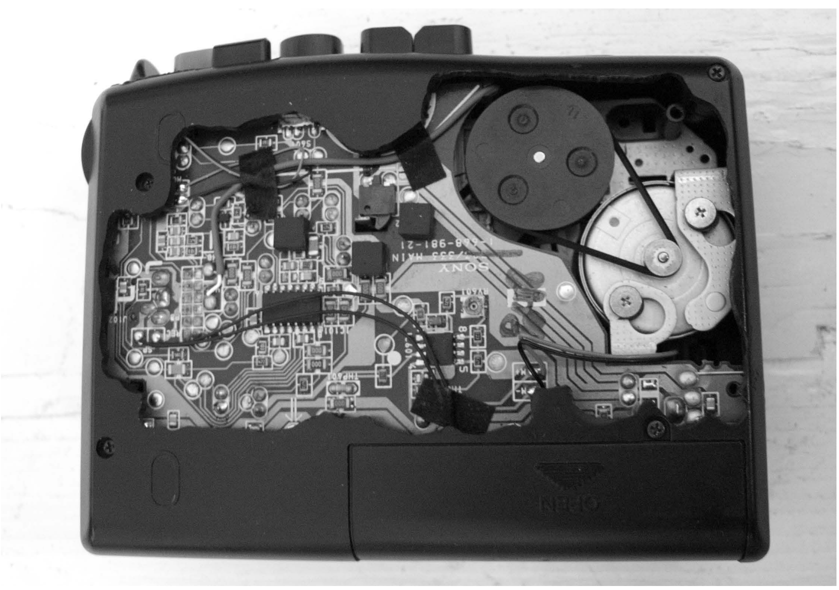


Figure 12.4 Seth Cluett, open-back cassette player, played with hands on the circuit board.

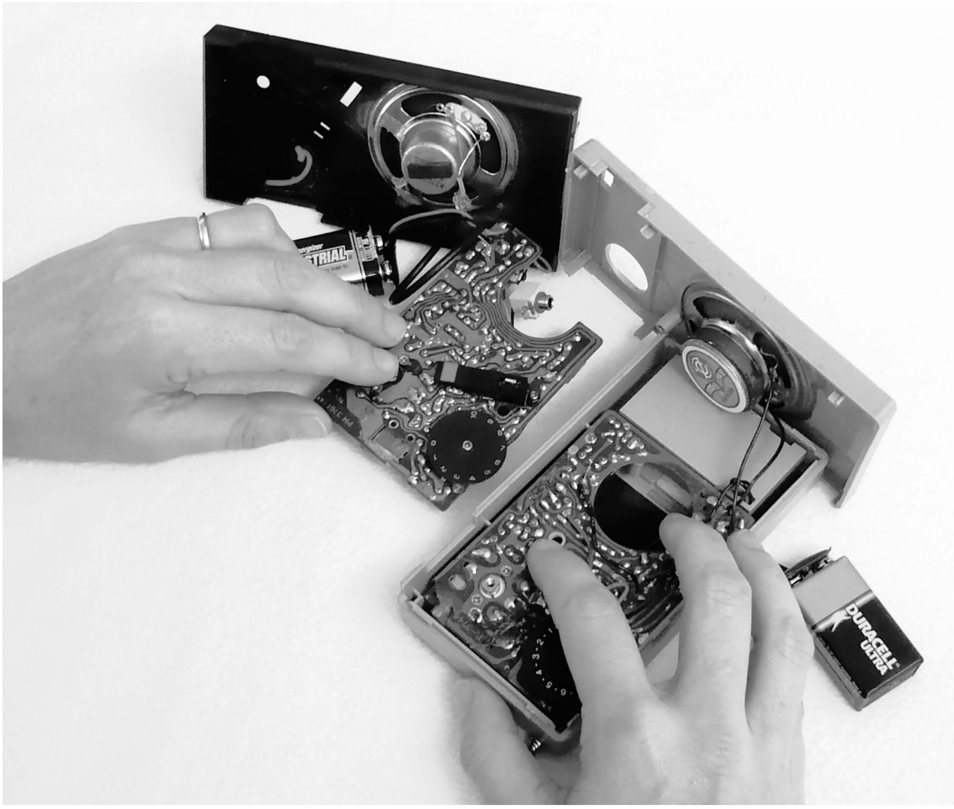


Figure 12.5 Susan Stenger does Rick Wakeman—spooky interaction between two radios.

Very spooky things start to happen when you link two or more separate radio circuits with your fingers—spread a few open radios in front of you like a set of Tarot cards and try it (Figure 12.5).

NOTE

1. www.mijnwebwinkel.nl/winkel/steim-webshop/a-45371328/products/cracklebox/

PART 3

Building

CHAPTER 13

My First Oscillator™
Six Oscillators on a Chip, Guaranteed to Work

You will need:

- A plastic prototyping board ("breadboard").
- One CMOS Hex Schmitt Trigger Inverter Integrated Circuit (74C14, CD4584, or CD40106).
- Assorted resistors and capacitors.
- One or more photoresistors.
- One or more potentiometers, 100 kOhm or larger.
- Some small signal diodes, such as 1N914.
- Anti-static foam from packaging integrated circuits.
- A lead from a mechanical pencil.
- Some paper and a soft pencil or graphite crayon.
- Some fruit and/or vegetables.
- Some solid hookup wire, 22–24 gauge.
- A monitor amplifier or mixer and speakers.
- A female audio jack of some kind.
- Plugs and cables to match your amp and the jack.
- A 9-volt battery and connector.
- Hand tools.

In the contrarian spirit of hacking, the first circuit we build from scratch is based on the misuse of an integrated circuit (IC) never intended for making sound. The "Hex Schmitt Trigger" is a CMOS digital logic chip consisting of six identical "inverters." An inverter takes a logical input, 1 or 0, and puts out its opposite (so 1 becomes 0, 0 becomes 1)—it is one of the fundamental Boolean building blocks that goes into the design of the computers we've come to depend on for so much of our work and play. This particular implementation of an inverter is useful to us because it runs for a long time on a 9-volt battery, it is very cheap, and it contains a circuit element known as a "Schmitt Trigger" whose fine points you don't need to understand right now but, trust me, transforms the chip from a simple digital no-man (as opposed to a yes-man) into a versatile sound generator.

The Hex Schmitt Trigger may be designated by the part numbers 4584, 40106, or 74C14. There may be prefixes, suffixes, or additional number strings that you can ignore, but chips with a different "innerfix" may not work: if labeled 74HC14 or 74AC14, it will not run on a 9-volt battery and so is less suitable for this project. If you purchase it online, sight unseen, make sure it is specified as having dual in-line (DIP) packaging and is not a surface mount device (SMD), as the latter format is infernally small and difficult for prototyping (this packaging requirement applies to most of the chips we will use in this book).

Figure 13.1 shows the 74C14's internal configuration and external connections. This is the information you need in order to make the correct connections between the chip, a battery, and the components needed to metamorphose a mute bug into a buzzing oscillator.

We will build our circuit on a solderless prototyping board, commonly referred to as a breadboard. On it you can assemble and rearrange components quickly, without solder. It consists of a plastic block with lots of little holes, beneath which are springy channels of metal arranged in a matrix (Figure 13.2). These strips, called "buses," run in one or two long horizontal strips along the top and bottom edges of the block and in numerous shorter vertical strips that extend above and below a central groove.

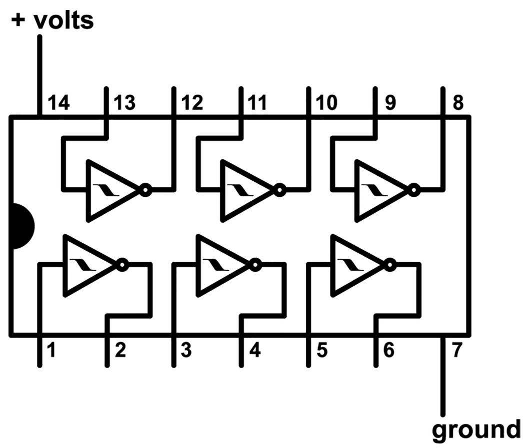


Figure 13.1
74C14 pinout.

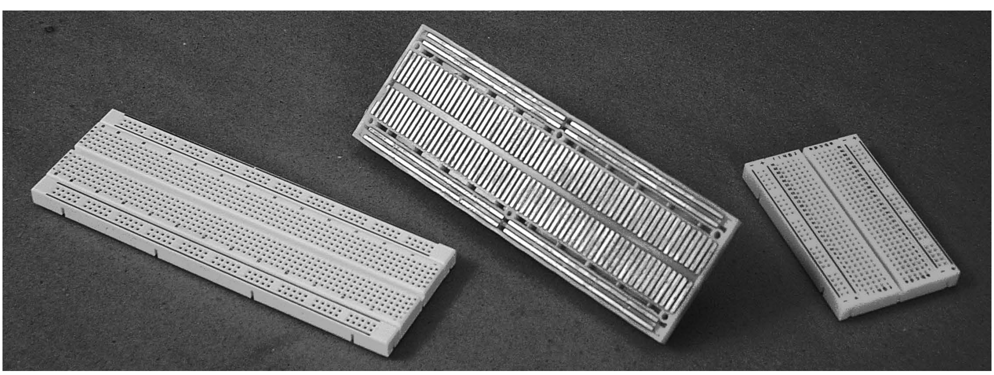


Figure 13.2 Solderless breadboards, showing bus strips on underside of board.

There are variations in breadboard design: some are longer, or have only one horizontal bus and the top and bottom instead of two, or consist of multiple plastic modules on a metal plate. But they all employ the same underlying bus and matrix system.

The holes are large enough to accept the leads of most electronic components (resistors, capacitors, integrated circuits, etc.) and solid hookup wire. Thanks to the metal channels underneath, any wire or component leg stuck down into a hole is connected electrically to anything else stuck into another hole in the same row or column of the matrix—as if you were clipping them all together with test leads, but much tidier. Circuits can be built up by inserting components into the holes on the board and linking rows and columns with short strips of wire (often called "jumpers").

YOUR FIRST BEEP

Place the breadboard on the table so the trough-like central groove runs horizontally, from left to right. If the breadboard is long, the upper and lower horizontal buses may be "broken" in the middle, rather than extending the full length of the board. If so, you will notice a slightly larger gap in the pattern of five holes—gap—five holes. Jump the gap with a small piece of wire, as shown in Figure 13.3.

Strip, twist, and tin 1/2 inch from the ends of each lead of a 9-volt-battery hookup clip so that the stranded wire can fit neatly into a hole on the breadboard. Push the end of the red wire into one end of a horizontal bus on the upper edge of the breadboard and the black wire into a horizontal bus along the lower edge. Don't snap the battery in yet; when you do, anything inserted in the upper bus will be connected to +9 volts while anything plugged into the lower one will be connected to ground (0 volts).

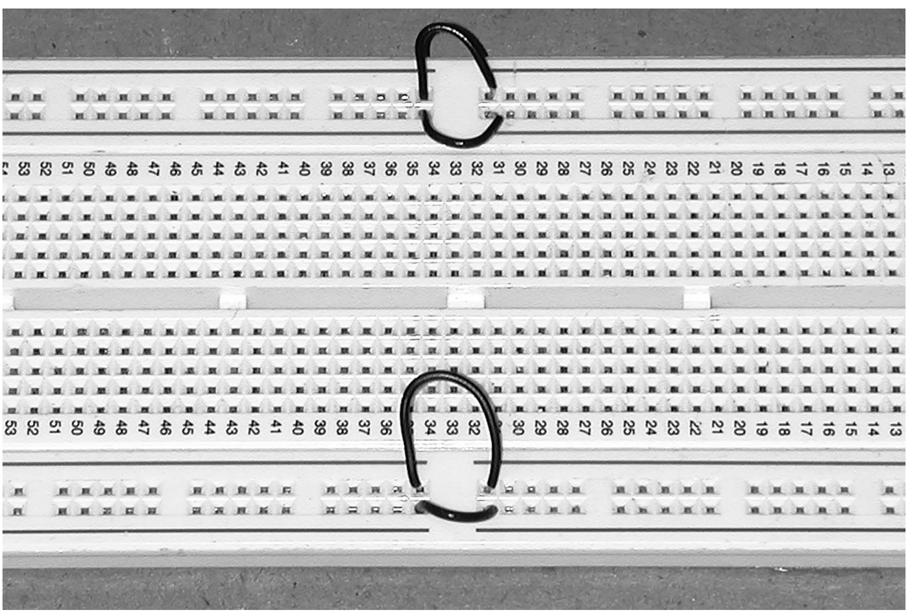


Figure 13.3 Using jumpers to join split power buses.

Press a Hex Schmitt Trigger IC into the breadboard, taking care not to bend over any pins. Be sure that the notch and/or small dot is at the left side—these markings help you orient the otherwise symmetrical chip. Strip 1/2 inch of insulation off either end of a short strip of solid wire and use it to connect between any free hole in the vertical matrix column containing pin 14 and any in the +9-volt horizontal bus at the top of the board. Use another wire to connect pin 7's column to the ground bus (Figure 13.4). If there are two horizontal buses at the top and bottom, be careful to link to the correct one and not its parallel twin (an easy mistake to make). You have now connected voltages from the battery clip to the chip.

The next component we add to the board is a capacitor. Capacitance is measured in farads, usually scaled down to microfarads (uf), nanofarads (nf), and picofarads (pf). We're looking for a 0.1 uf capacitor. Sometimes you'll get lucky and the component will be marked with exactly those numbers ("0.1"), but often the labeling follows a simple code: the first two most significant digits followed by a multiplier. A 0.1 uf capacitor would be indicated as "104," where one 10 times smaller—0.01 uf—would be "103." (The easiest way to familiarize yourself with this argot is to buy a few capacitors of known value and look for correspondences between the numbers printed on the capacitor and its actual value, as specified in the packaging or ordering information. You can also find handy charts online that lay out this system.)

Connect a 0.1 uf capacitor between pin 1 and the ground bus: push one lead of the capacitor into a hole in the same vertical bus into which pin 1 has been inserted and the other end into a hole in the horizontal ground bus. You can use any of holes in these buses, but with components with closely spaced leads (such as capacitors), it's easier to use holes that are close together (Figure 13.5).

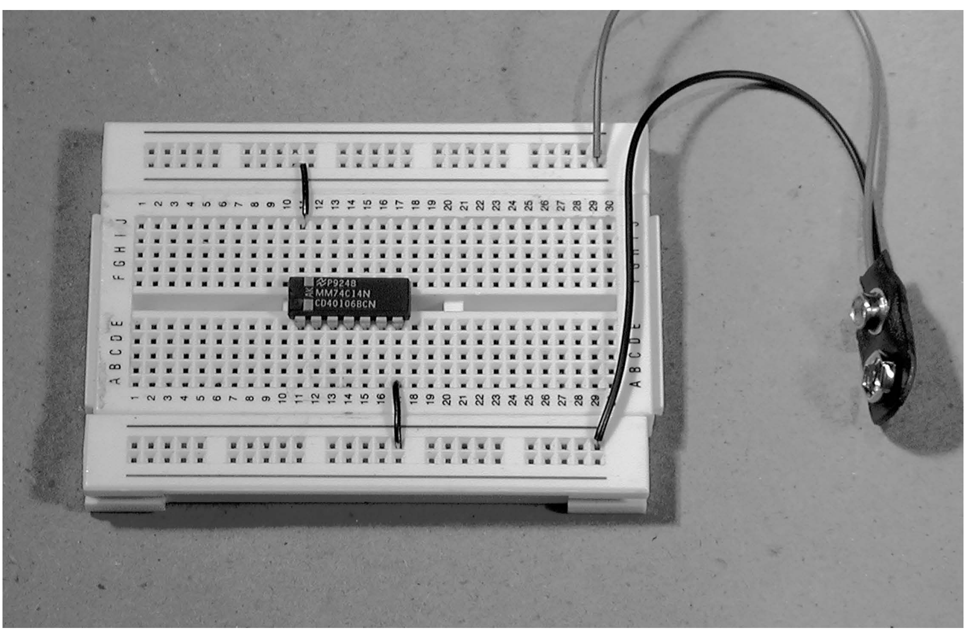


Figure 13.4 74C14 in place, with power connections.

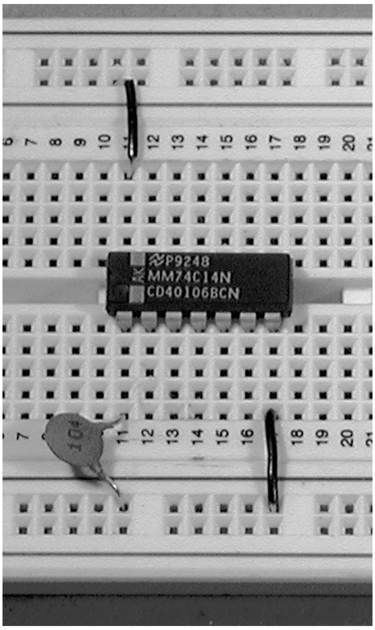


Figure 13.5
Capacitor added.

Next we need a resistor of around 100 kOhms. Resistor value is coded in colorful stripes (Table 13.1). Similar to capacitors, the first two stripes indicate the two most significant digits, the third one is the multiplier, and the last specifies the tolerance of the value. As per the table, a 100 k resistor would be marked:

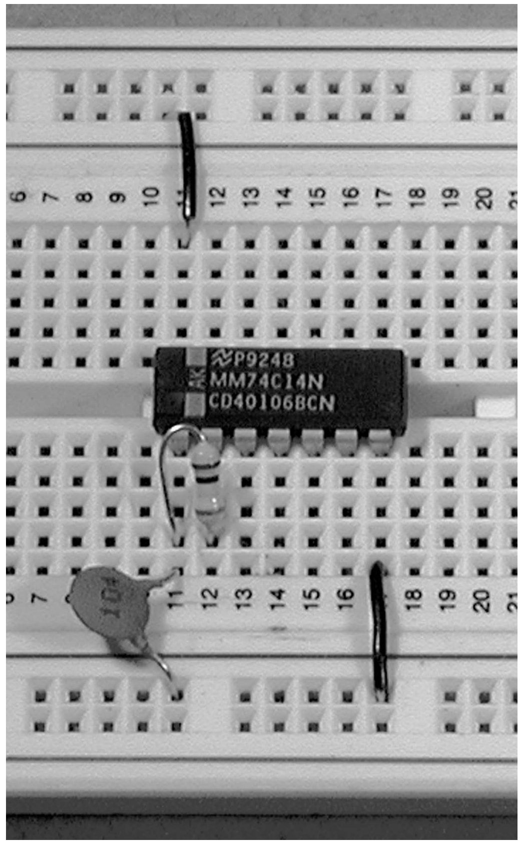
Brown = 1
Black = 0
Yellow = multiply by 10,000

The last stripe will probably be gold (5% tolerance) but is not critical at this point.

Table 13.1 **Resistor color code.**

Color	*Value*	*Multiplier*
black	0	1
brown	1	10
red	2	100
orange	3	1,000
yellow	4	10,000
green	5	100,000
blue	6	1,000,000
violet	7	10,000,000
gray	8	100,000,000
white	9	1,000,000,000

Figure 13.6
Resistor added.



Bend the leads on the resistor so it resembles a misshapen croquet wicket. Push one leg into the bus connected to pin 1 of the chip, which should already have one leg of the capacitor attached. Insert the other leg into the bus connected to pin 2 (Figure 13.6).

Solder short pieces of solid wire (the same kind you use for your jumpers) to the tip and sleeve connections on a female jack that mates with a cord you can connect to your amplifier. Connect the wire from the tip/hot of the jack to the pin 2 vertical bus and the wire from the sleeve of the jack to any convenient point along the horizontal ground bus (Figure 13.7). Insert one end of a cord into the jack on the breadboard and the other end into your amp. Alternatively, you can solder up a patchcord that goes from a plug that mates with your amplifier/mixer to two bare solid wires to plug into the breadboard. Or you can clip test leads to short pieces of bare wire inserted in the appropriate buses and clip the other ends of the leads to an open audio plug that matches the input of your amplifier.

Snap the battery into its hookup clip. Connect the jack/plug to your amplifier, turn it on, and listen (keep the volume low—this circuit is *loud*!) You should hear a strident, steady pitch—a square wave. Congratulations, you're an engineer!

No joy?

- Check your connections. It's very easy to be off by one hole to the left or right when you insert component leads and jumper wires.
- If there are double power buses at the top and bottom of the board, make sure you've connected the chip and the other components to the same bus and that the bus is connected to the correct lead from the battery clip (and bridge the gap if the buses are split, as shown in Figure 13.3).
- If the chip is HOT, disconnect the battery immediately and start checking your wiring. In particular make sure you haven't put the chip in the board backwards or reversed the battery connections. Hooking the voltage up backwards is one of the only things that can destroy this plucky little silicon warrior—if the chip was cooking for several minutes before you disconnected the battery, you may need to start over with a fresh chip, after a hasty burial for the fallen hero.
- Make sure the component values are correct: too small a capacitor or resistor will cause the circuit to oscillate at a frequency too high for you to hear (you might notice your dog complaining, though); very large values in turn produce sub-audio frequencies, a slow tick-tock that you might miss.

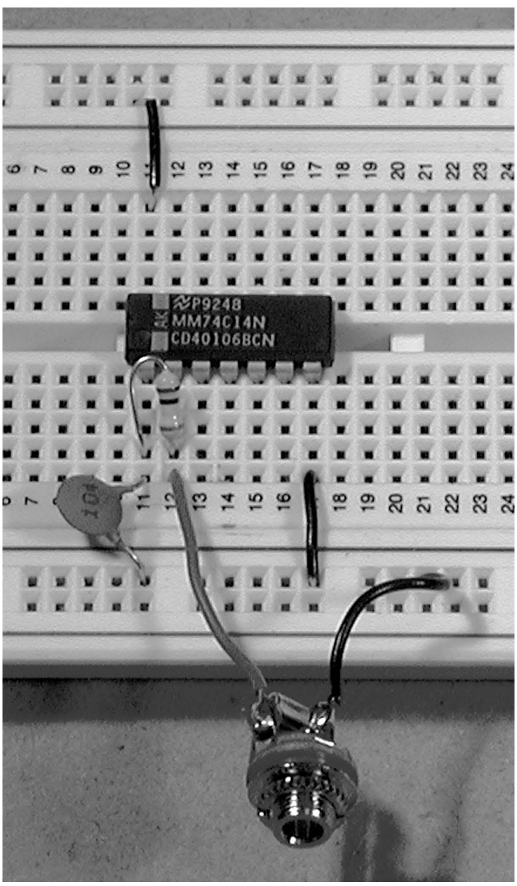


Figure 13.7
Jack added.

- Check that none of the chip's pins have been folded, yogi-like, under the chip instead of inserted into a hole.
- If the circuit oscillates, but erratically, and is sensitive to placing fingers on chip pins and component leads, then you may have forgotten to connect the + or − power to the chip, or connected them to the wrong pins, or left one leg of the capacitor or resistor unconnected. Sometimes the circuit *almost* works without a direct power hookup by sucking voltage through other connections you have made (spooky!).
- Confirm that the chip has none of the mysterious "innerfixes" I mentioned at the head of the chapter since those chips only run reliably on 5 volts and will behave very erratically (or blow up) with a 9-volt battery.
- If nothing else works, strip everything off the breadboard and start over, in a different location, with different components. This is one of the world's simplest circuit designs and usually works first time, every time. But with every connection comes the possibility of failure, and in rare circumstances it can take a second try to get things humming.
- Check all your connections again . . . and again . . . and again. . .

YOUR FIRST SCHEMATIC

This is as good a time as any to start getting familiar with *schematic* representation, which conveys a circuit design independently of the physical arrangement of its components on a breadboard (Figure 13.8).

The big triangle represents one inverter—any of the six in the 74C14 package; the flat left side is the input, which could be pin 1, 3, 5, etc., while the pointed right side is the output (2, 4, 6, etc.) We happen to be using the inverter that connects through pins 1 and 2 of the chip. The zigzag line symbolizes the resistor, between input and output. The two vaguely parallel lines (one straight, one curved) symbolize the capacitor,

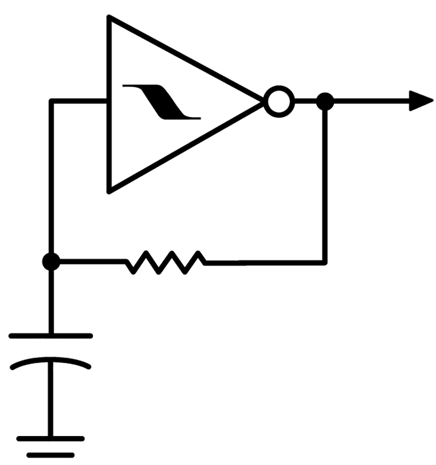


Figure 13.8
Schematic representation of our oscillator.

connected between the inverter's input and the ground bus (− voltage). Ground is represented by the weird runic arrangement of three lines. The output appears as a single line with an arrowhead at the end. This is the signal that goes to the tip of a plug; the ground half of the connection is implied, rather than drawn in (Rule #10). Likewise, the power connections to the chip (+9 volts to pin 14, ground/ −9 volts to pin 7) are implicit in this drawing.

The translation from this symbolic schematic to parts on a breadboard may not seem obvious at first, but as you get more familiar with electronic components, you'll see that the schematic is a useful way to represent the way a circuit functions, rather than just the way the parts fit together. As I introduce new designs, I'll present both a picture of a typical physical arrangement on a breadboard and a schematic representation. It's worth familiarizing yourself with the language of schematics since it opens up a huge world of circuit possibilities—on the web, in books, and in technical journals.

VARIATIONS

You are justified in taking great satisfaction in producing your first electronic tone, but after a while you may wish for a change of pitch. Try substituting different resistors and capacitors and listen to the effect. The wiggly resistor symbol in the prior schematic can be taken to mean any form of resistor, including variable ones such as pots or photoresistors.

A potentiometer (or pot) is a variable resistor that gives you direct, repeatable control of frequency. It has three terminals—two "ears" and one "nose"—that are labeled A, B, and C in Figure 13.9. The resistance between the outer two ears (A and C) is fixed at the designated value of the pot, which is the pot's absolute maximum resistance (i.e., 1 mOhm). As you rotate the shaft of the pot clockwise, the resistance between the center terminal (nose) B and the outer terminal A goes *up* from 0 Ohms to the maximum value while the resistance between B and the other outer terminal C goes *down* from the maximum to 0—the two values change in contrary motion, like a seesaw. Reversing the pot's rotation tips the seesaw back the other way.

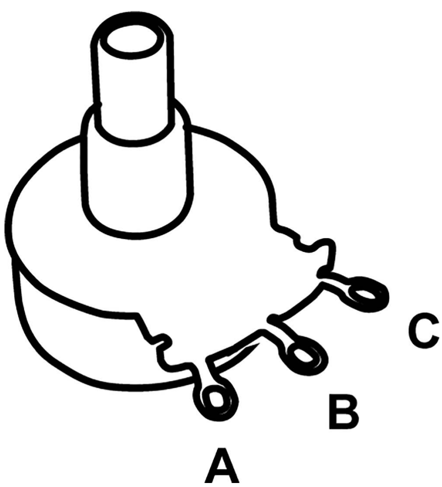


Figure 13.9
The three terminals of a potentiometer.

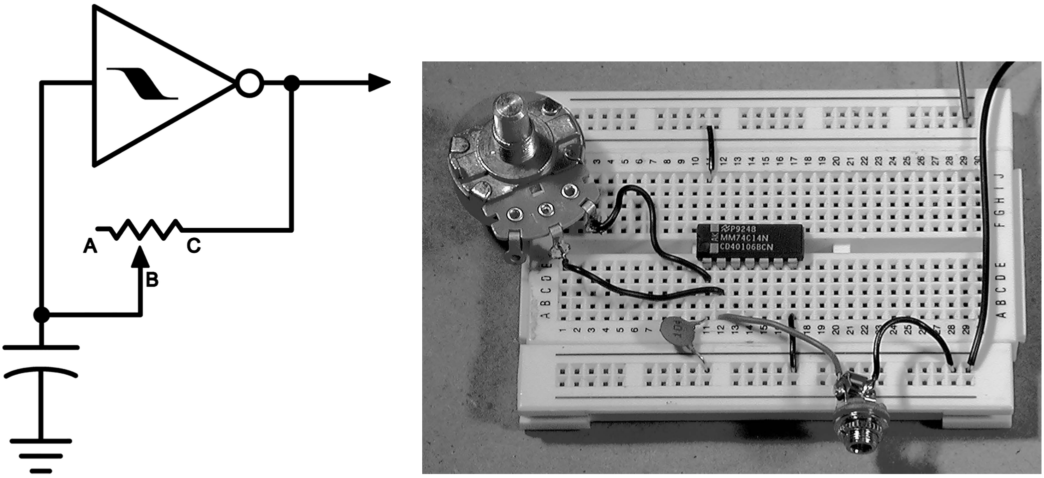


Figure 13.10 Potentiometer-controlled oscillator: schematic and photo.

In our schematic we can specify that the resistor is a potentiometer if we use the symbol in Figure 13.10.

A photoresistor (or photocell, as it is sometimes called) is a device that changes its value in response to light level: the resistance gets smaller when it is exposed to light and gets larger in the dark. Two wires come out of one side, and the other side displays a pleasing zigzag pattern of fine lines. The side with lines is more sensitive to light than the other, but the back is translucent enough that light striking the back will affect the resistance as well. The lowest resistance in bright light is anywhere from 100 to 2,000 Ohms while the "dark resistance" can be as great as 25 mOhms.

A photoresistor-controlled version of the oscillator is shown in Figure 13.11. A photoresistor turns this simple oscillator circuit into a Theremin-style instrument

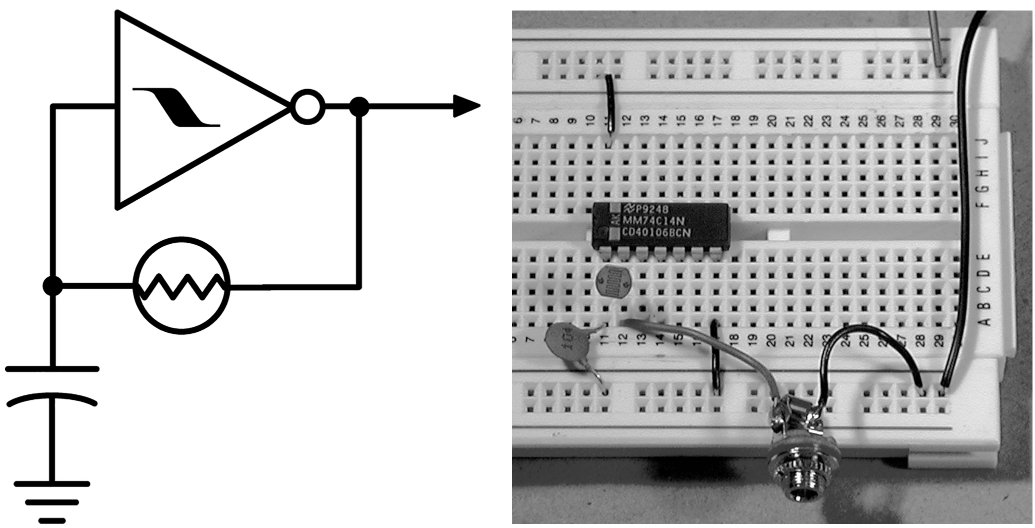


Figure 13.11 Photoresistor-controlled oscillator: schematic and photo.

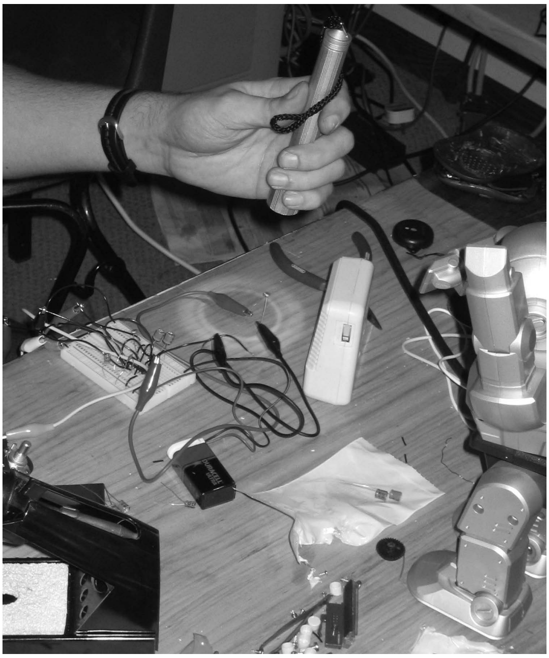


Figure 13.12
Playing a photoresistor-controlled oscillator with a flashlight.

controlled by light and shadow (Figure 13.12). You can wiggle your hands between the circuit and any ambient light, aim a flashlight, light a candle, interrupt a laser with a fan, etc. An excellent excuse to download a strobe app for your phone!

The capacitor determines the range through which the variable resistor will sweep the pitch. Larger values (greater than 2.2 uf or so) lower the frequency from the range of audible pitches to that of rhythm—you'll hear the oscillation as a tick-tock of the sharp edge of the square wave instead of a continuous buzz. With a very small capacitor (less than 0.001 uf), the sounds will be so high that only dogs and bats can hear them.

Capacitors 1.0 uf or larger are usually found in a style called "electrolytic" and take the shape of little plastic cylinders. Electrolytic capacitors have polarity, like a battery: one leg will be shorter and labeled "−" with a string of minus symbols or a black stripe along one side of the cylinder. When using electrolytic capacitors, be sure to observe the correct polarity and hook the "−" side to the ground bus (in a schematic the "−" side of the capacitor is indicated by the curved line). The circuit will work if the capacitor is connected backwards, but it may not be as stable or go as low as you expect. Electrolytic capacitors usually have their value printed quite clearly.

WHY? (IF YOU CARE)

Our circuit oscillates because of the principle of argumentation. Each inverter stage, represented by the big triangle in the schematic, puts out the opposite of whatever signal appears at the input: if a binary "1" is applied to the input, then a "0" appears as the output. In a circuit running on a 9-volt battery, that theoretical 1 is represented by a 9-volt signal, while logical 0 equals 0 volts. The 0 at the output flows through the resistor back to the input. When the 0 appears at the input, it now causes the output to switch to 1 (9 volts), which returns to the input and the whole process begins again, causing the circuit to flip back and forth between two states, generating a square wave that alternates between 0 and 9 volts. The speed of the flip-flopping is the pitch we hear, and it depends on the values of the resistor and capacitor.

It's like the Monty Python argument sketch, transferred to a bar: I disagree with everything you say, you disagree with every one of my replies, so our output keeps flipping between yes and no according to how fast each of us can reply. The resistance and capacitance act like beer: the more you have, the slower the argument goes, ergo the lower the pitch. The appropriately Germanic-sounding Schmitt Trigger part of the inverter prevents indecisiveness in the argument: the inverter snaps completely from one state to the other, from 0 to 1 and back, and never vacillates in between or proffers a "maybe" at the output.

Since I've already started down the slippery slope of liquid analogies, I can offer one more for those of you who want a deeper understanding of the interaction of the capacitor and resistor. Pretend for a moment that the inverter processes water rather than voltage (Figure 13.13). Water flows from the output back to the input through a pipe, which is our resistor. If the pipe is skinny (high resistance, left side of figure), it takes longer for the water to flow back than if it is fat (low resistance, right figure). At the input to the inverter is a basin, which represents the capacitor. Halfway up is our "logic line": if the water is below that line, we have a 0; when it rises above, we have 1. If the basin is a wide washtub, it takes longer to reach the line (big capacitor, left figure) than if it is a narrow, modern vase (small capacitor, right figure), since the pipe must top up a greater volume of water. Got it? Now commence the rinse cycle.

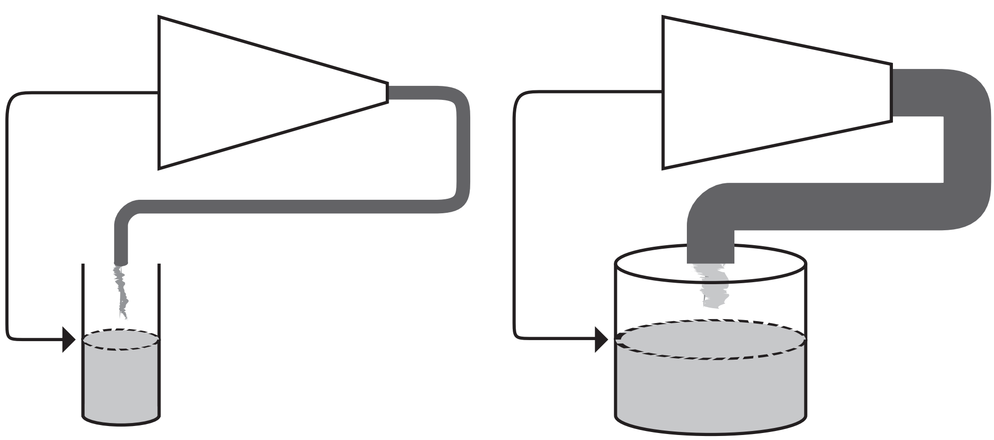


Figure 13.13 Pipe and bucket model of inverter signal flow.

MORE VARIATIONS

When you use a potentiometer, you might notice that as you turn toward one extreme of the rotation, the pitch will zip up too high to be heard, regardless of the size of the capacitor. The resistance at one end approaches 0 Ohms, which results in a very high pitch no matter how big the capacitor. This ultrasonic dead zone limits the useful range of the pot, forcing you to make all your musical decisions within a slightly constricted amount of rotation. You can put a fixed resistor of modest value in series with one leg of the pot (Figure 13.14) to set a maximum pitch that is within the range of hearing. Start with something around 10 kOhm, then substitute larger and smaller values until you find the one that yields the best balance of resolution and range.

Similarly, if you wire a pot in series with a photoresistor instead of a fixed resistor, the pot will set the upper pitch limit of a light-control instrument, combining the accuracy of the pot and the expressiveness of the photoresistor.

There is little significant variation among capacitors except for value (in farads), but resistors come in myriad forms. Other things to try inserting between the two pins of the inverter in lieu of a fixed resistor include:

- Photoresistor in mouth. A photoresistor in your mouth makes a very expressive controller that responds to changes in the light level as you open and close your mouth, as well in the conductivity of your saliva-laden tongue across the photoresistor's bare legs (a suggestively naughty extension of the wet-fingers-on-circuit-board effect). **ONLY DO THIS EXPERIMENT UNDER 100% BATTERY POWER. DON'T CONNECT TO ANYTHING THAT PLUGS INTO WALL CURRENT!**
- Electrodes. We can adapt the touch sensitivity of our radio-cracklebox (Chapter 12) to our oscillator by substituting two coins for the legs of a resistor (Figure 13.15). Good party fun can be had with "social electrodes": gather a group of friends and ask them to hold hands in a line, give one electrode to the person at either

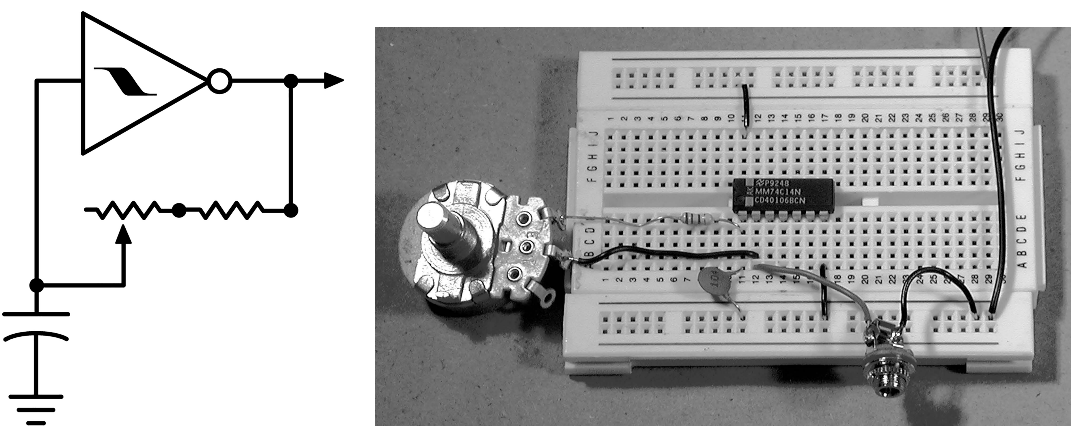


Figure 13.14 Potentiometer with resistor for setting upper frequency limit.

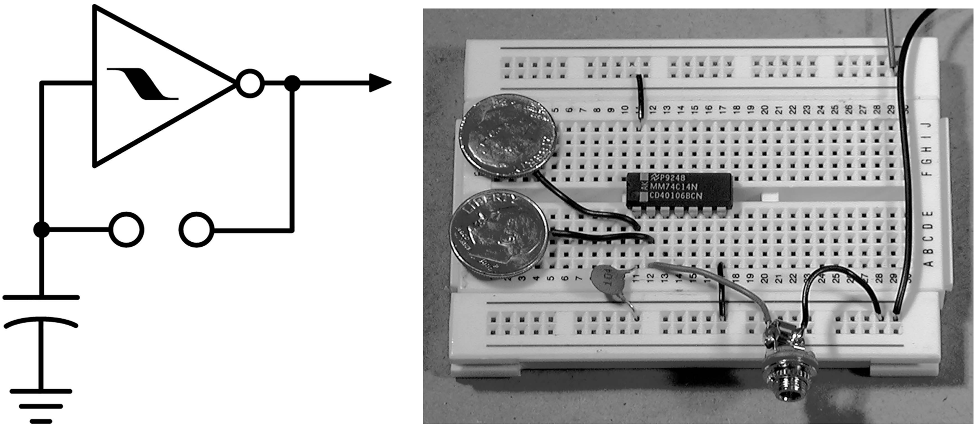


Figure 13.15 An electrode-controlled oscillator.

end, and listen to the group sweat (you'll need to use a rather small capacitor in the oscillators to get an audio pitch rather than rhythm since the net resistance of several bodies in series is very big). (See Lauren Carter and Joe Grimm's "Picnic" video on the website.)

- Potentiometer with electrodes in parallel. Wire a coin to each tab of the pot that connects to the inverter; turn the pot for direct control, touch the electrodes for gestural effects.
- Pressure-sensitive foam rubber. The squashy black "anti-static foam" in which integrated circuits are often packaged changes resistance with pressure. Put a piece between two coin-electrodes and wire them to the inverter's input and output pins (Figure 13.16).
- Corroded metal, like we used in our Victorian Synthesizer (Chapter 3).
- Pencil lead. The resistance increases with the distance separating the contact point, like a homemade slide pot (Figure 13.17).
- Graphite on paper. Draw two blots near the edge of a piece of paper. Clamp one end of a clip lead to the paper at each blot; connect the other ends to the two pins of the inverter. Draw a line between the two blots (Figure 13.18). As you widen or darken the drawing linking the blots, the pitch goes up since the more graphite between the clips, the lower the resistance. (Patrick McCarthy has made actual potentiometers using this technique—see video on the website).
- Magnetic tape. Tape is also conductive, and, as with the pencil lead, the resistance varies with distance. A job for all those VHS tapes in your attic.
- Vegetables and fruit. Produce has resistive value that changes as it dries out or is squished. (See Grégoire Lauvin's video on the website).
- Another circuit. Use clip leads to connect from the input and output of the inverter to any two points on another circuit board in lieu of a resistor. Try dead cell phones, computer motherboards, radios, electronic toys, etc. (See Nicolas Collins's *Salvage* and *The Royal Touch* videos on the website) Just make sure the appliance is switched off and **NOT PLUGGED INTO THE WALL!!**

Figure 13.16
Pressure sensor made from anti-static foam between two coins.

Figure 13.17 A pencil-lead resistor.

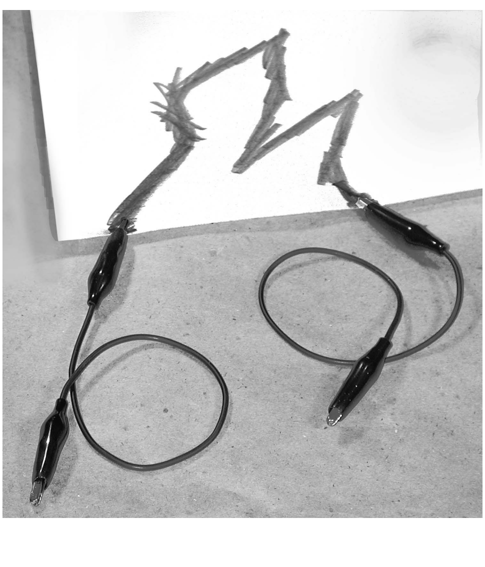


Figure 13.18
Drawing a resistor.

POLYPHONY

As you might be able to intuit from the pinout in Figure 13.1 or your knowledge of Greek, the *hex* inverter has six identical sections. You can make an additional oscillator with any of these sections: just duplicate the connections we made for our first oscillator with another set of components, attached to another set of pins. For example, insert a capacitor between pin 3 (input to inverter #2) and ground; insert a photoresistor between pin 3 and pin 4 (output); connect pin 4 (output) to a jack, and patch that jack into your amplifier. Confirm that it works.

Great, but we disconnected our first oscillator to hear the second. How can we listen to both? If you have a mixer, you can patch one cord between each oscillator and a separate input to the mixer. But if you don't have a mixer? It would be great if we could just clip the two oscillator outputs together and send them down one cord to your amp, but, sadly, that won't work. If you think back to my explanation of this circuit, you'll recall that each input is connected to an output that switches between 1 and 0. If we clip the outputs together, both appear at each input. If one output is 1 and the other 0, they combine to present 1/2 to each input. The chip only understands 1 (yes) or 0 (no): 1/2 (maybe) sends the chip into existential crisis and it stops oscillating. Silence, *quelle dommage*.

But don't despair. We can make our own simple (and cheap) mixer right on the breadboard. Take a fixed resistor in the range of 10 kOhms–100 kOhms. Insert one end in the row with pin 2, which is the output of our first oscillator. Stick the other end in an unoccupied row beyond the chip (i.e., not in a row shared with one of the chip legs)—we'll call this row our mix bus. Next, take another resistor of the same value and insert one end in the row with pin 4 (output of oscillator #2) and the other end in the same "mix bus" row you used for the resistor connected to your first oscillator. Finally, connect from this mix bus and ground to your amp (Figure 13.19).

You can continue to add voices until you are using all six stages (Figure 13.22). Refer to the pinout in Figure 13.1. Each inverter needs: a capacitor from the input to ground, a pitch-control resistor (potentiometer, photoresistor, etc.) from the output

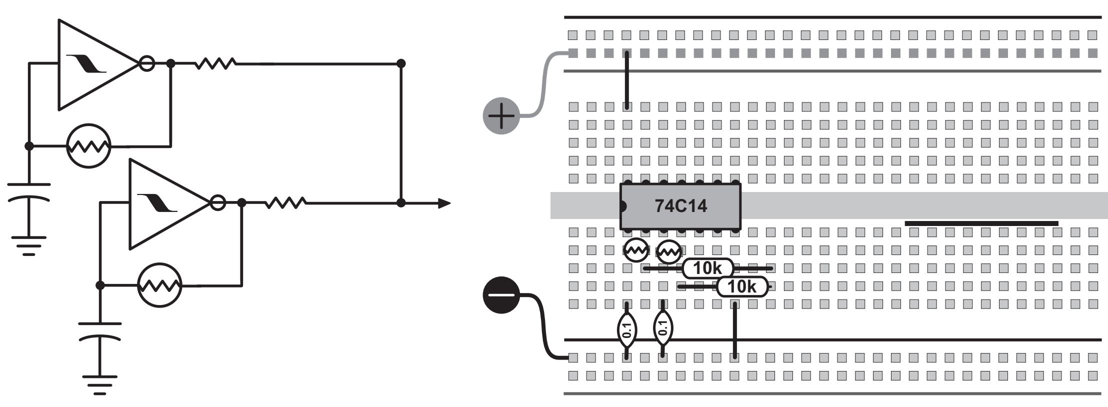


Figure 13.19 Two oscillators mixed through resistors.

back to the input, and a mixing resistor from the output to the mix bus row. There are two important things to bear in mind as you work:

- Make sure you can hear each new voice clearly before building the next. Don't assume after you get one oscillator buzzing that you can smite the remaining five in one blow. When you find yourself gazing down at a rat's nest of wires that makes neither sound nor sense, strip your board back to one oscillator and start over, one voice at a time, listening as you go. Which brings us to another valuable lesson enshrined in our Rules of Hacking:

Rule #17: Start simple and confirm that the circuit still works after every addition you make.[1]

- Remember when working on the "top" side of the chip (pins 8–14) that the capacitor must go between the chip and ground, not to the + supply that mirrors the ground bus on the upper side of the board. If there are two parallel horizontal buses at the top and bottom of the board, you can connect a wire jumper between the lower ground bus and the upper bus that is not being used for +9 volts; the second upper bus is now an additional ground bus into which you can insert the legs of the capacitors used on the upper half of the chip (Figure 13.20). Be careful to connect from the lower ground bus to the upper line designated for ground, not the 9-volt bus: if you accidentally link the ground and power buses together with a wire, you will soon have a very hot, very dead battery.

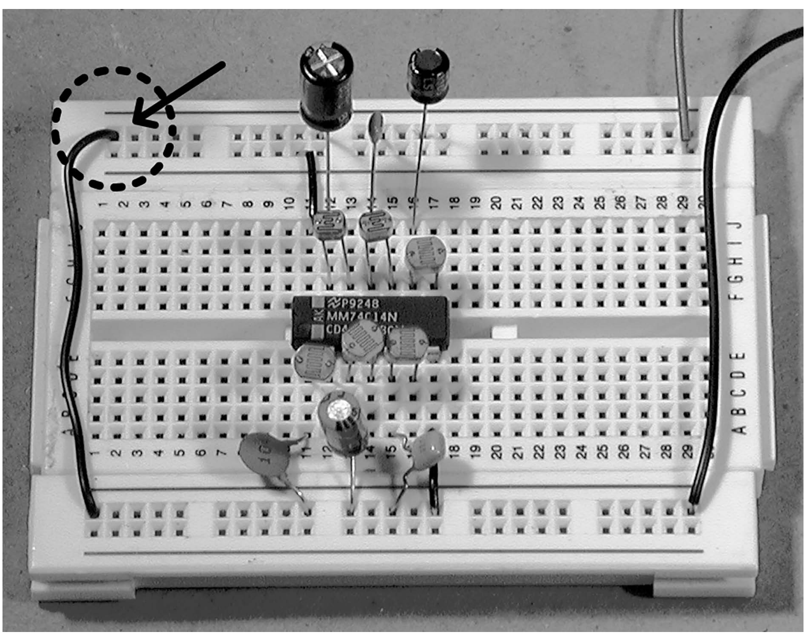


Figure 13.20 Six-voice oscillator, with capacitors connected to secondary ground bus parallel to + voltage bus along upper edge of breadboard.

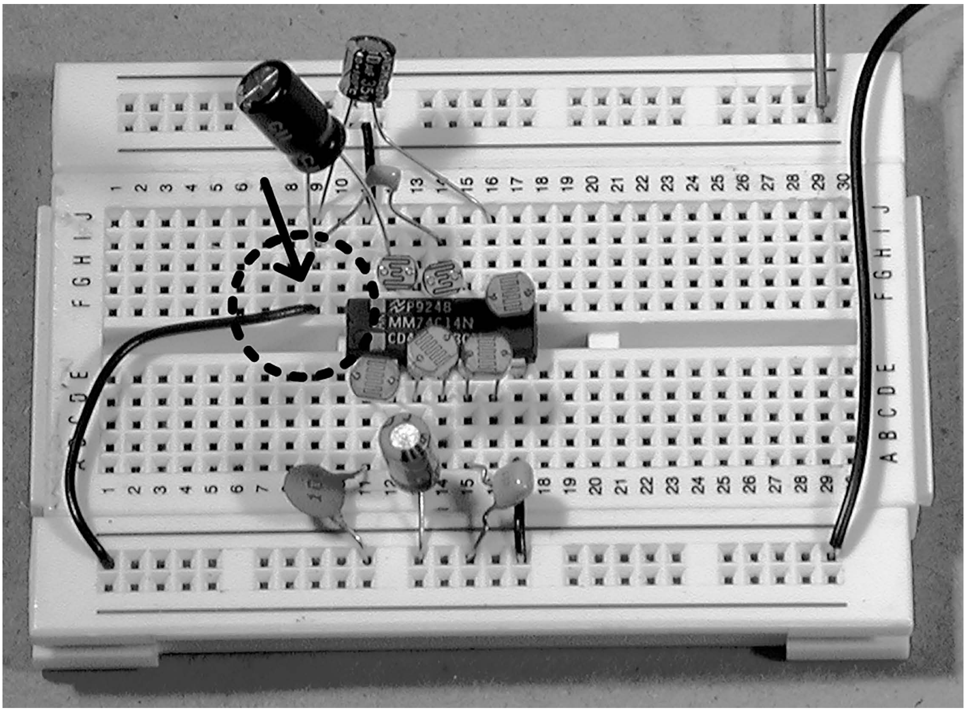


Figure 13.21 Six-voice oscillator, with capacitors connected to satellite mini ground bus.

- Alternatively, you can push the grounded legs of the capacitors into any free vertical bus on either side of the chip and jumper that vertical bus to the lower ground bus, creating a secondary, ground "satellite-bus" (Figure 13.21). Finally, if the breadboard only has one horizontal bus at the top and one at the bottom, you can use both for ground and connect the +9-volt (red) wire of the battery holder directly to pin 14, rather than via a long bus.

You can use a different size capacitor for each oscillator so that each of the six covers a different range, from low BPM to ultrasonic pitches. You can add a switch to select among different capacitors for each oscillator—a toggle or rotary switch gives you direct control while a multi-position tilt switch makes a nice wobbly interface (see the Technical Bootcamp section of the website for a guide to different kinds of switches). You can use a tilt switch to select different oscillator outputs as you tip and wobble the circuit.

A joystick can be an expressive device for controlling pairs of oscillators. You can salvage one from an unneeded game controller. A proper analog joystick consists of two potentiometers controlled by the *X*–*Y* movement of a shaft (some games just use four switches activated at the four extremes of movement). If you de-solder a joystick from its original circuit, you will notice the familiar three terminals on each of the small pots. Connect the nose and one ear of each pot to each oscillator. You may need

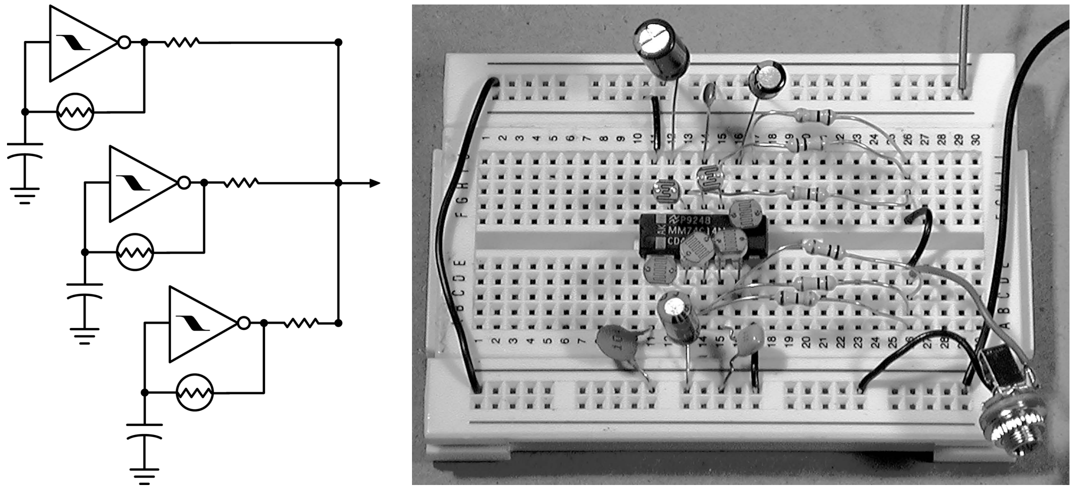


Figure 13.22 Six oscillators mixed through resistors (only three shown on schematic).

to try various connections before you arrive at the most satisfying interaction between the two oscillators, but when you get there, you'll be rewarded by square waves careening off each other like radio-controlled mosquitoes.

REAL ELECTRONIC MUSIC

The resistors form a simple linear mixer, through which one can hear each oscillator distinctly—like a recording mixer but with the knobs glued in place (we can't adjust the levels independently—that comes later in the book, in Chapter 18). If instead you mix the outputs together using a component called a diode, the individual oscillators will interact and distort in an archetypically "electronic music" way—similar to a ring modulator, in which sum and difference frequencies are exaggerated and the individual source pitches suppressed. Pull out the mixing resistors and substitute diodes as shown in Figures 13.23 and 13.24. Make sure the diodes all have the same orientation, with the stripe facing the output mixing bus. At the mixing bus, add a resistor of around 10 kOhms to ground (not shown in breadboard view). As per our new rule, it's advisable to start with two or three oscillators to evaluate the effect.

Diodes are odd little devices that only allow signals to pass in one direction. Current flows toward the leg indicated by the single stripe that appears at one end of the component. Exactly why they make such a cool sounding mixer for oscillators is a somewhat complicated story, and somewhat irrelevant to appreciating their sonic contribution, so I won't try to explain it. Which brings us neatly to Rule #18:

Rule #18: If it sounds good and doesn't smoke, don't worry if you don't understand it.

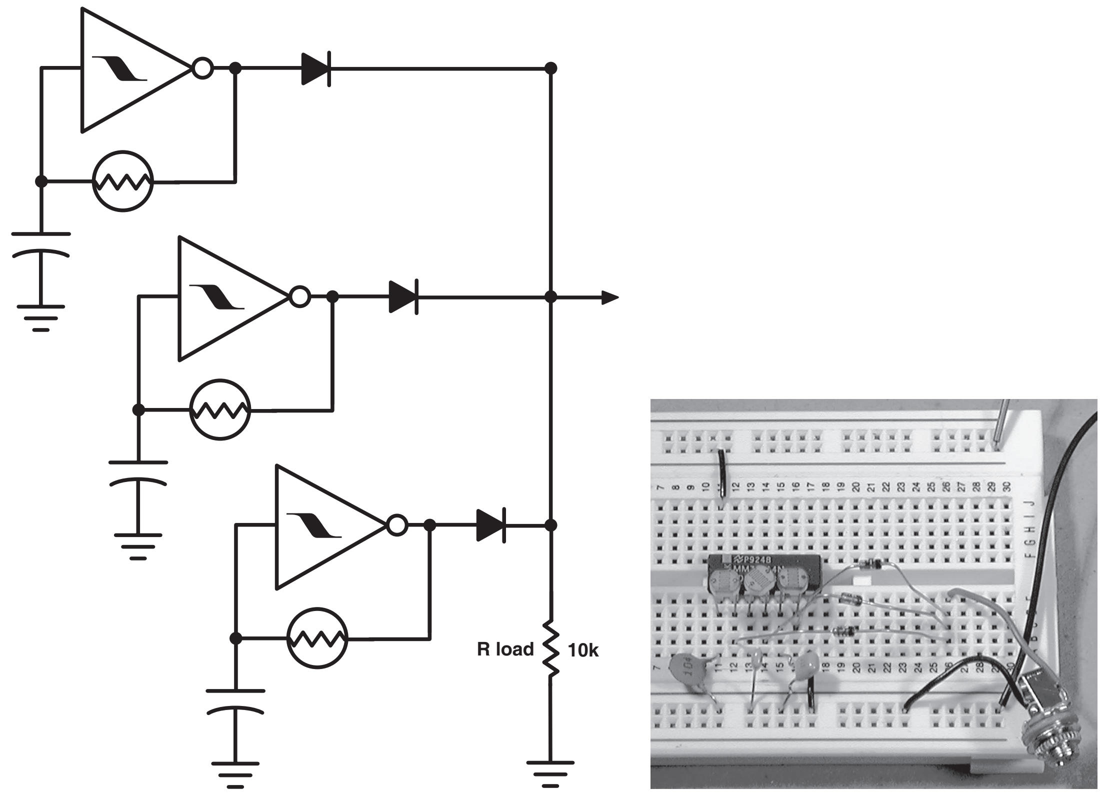
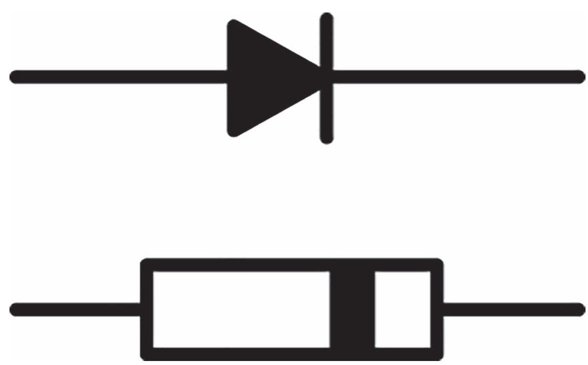


Figure 13.23 Three oscillators mixed with diodes.

Figure 13.24 Diode orientation.



If some of the oscillators are running at low frequencies (in the metronome range) while others are in the audio range, the low ones may gate the high ones on and off. See Chapter 15 for more advice on cross-modulating oscillators, and in Chapter 25 we'll build a proper stand-alone ring modulator.

Six square-wave oscillators make a glorious din. With six photoresistors or electrode pairs, control is unpredictable but nonetheless intuitive and playable. There's a glorious tradition of music made with masses of homemade oscillators, from David Behrman (see his audio track on the website) through Eliane Radigue and beyond. Indulge.

NOTE

1. Sharp-eyed readers will note a gap in the numbering of the Rules of Hacking: we jumped from Rule #11 in chapter 7 to #17 here—what happened to #12–16? Those missing rules arose from the art of circuit bending, and my chapters on that subject have been moved to the website for this book. The full, properly sequenced set are listed in Appendix B in this book, but refer to the chapters in the Circuit Bending section of the website for their context.

CHAPTER 14

Solder Up! From Breadboard to Circuit Board

You will need:

- Your breadboarded circuit from the previous chapter.
- A full duplicate set of parts used in the circuit.
- A prototyping circuit solder board (such as Datek #12-617).
- 14-pin IC socket.
- Solid and stranded hookup wire.
- Hand tools and soldering iron.

The breadboard is great for trying out designs: components can be easily swapped and mistakes undone without burning your fingers. But it's not very stable if you want to take your music on the road. At some point you may wish to solidify the circuit. One solution, often the simplest, is to solder the components down onto a generic printed circuit board (PCB) specifically intended for prototyping and developing new designs (as opposed to the board you find inside a radio, computer, or phone, which was designed and fabricated for one particular circuit). You don't *have* to plug in your iron now—you can skip over this chapter and continue breadboarding more circuit variations before settling on one to solder. But if you're happy with what you made, and want to preserve it before clearing off the breadboard, read on.

Prototyping circuit boards come in various styles, with patterns of individual copper pads that can be linked together with components and bits of wire any way you wish (Figure 14.1). Some designs, such as Datek #12-617 (available from Unicorn Electronics), mimic a typical breadboard almost exactly and make it much easier to transfer your circuit from breadboard to soldered board (Figure 14.2).[1]

If the board you've chosen is larger than needed for your one-chip circuit, you may want to cut it down before you start soldering on parts—scribe along the dotted line with a sharp knife, then snap the board over the edge of a table (very satisfying, in a karate-lite kind of way). You can use the vacant half later. Alternatively, you can build your first circuit at one end and add designs as you develop them.

Once you've got your circuit board cut down to size, pick up a 14-pin IC socket. The socket is like a tiny version of the breadboard. Since chips can be damaged by the excess heat of sluggish soldering, and are very difficult to de-solder if they have to be

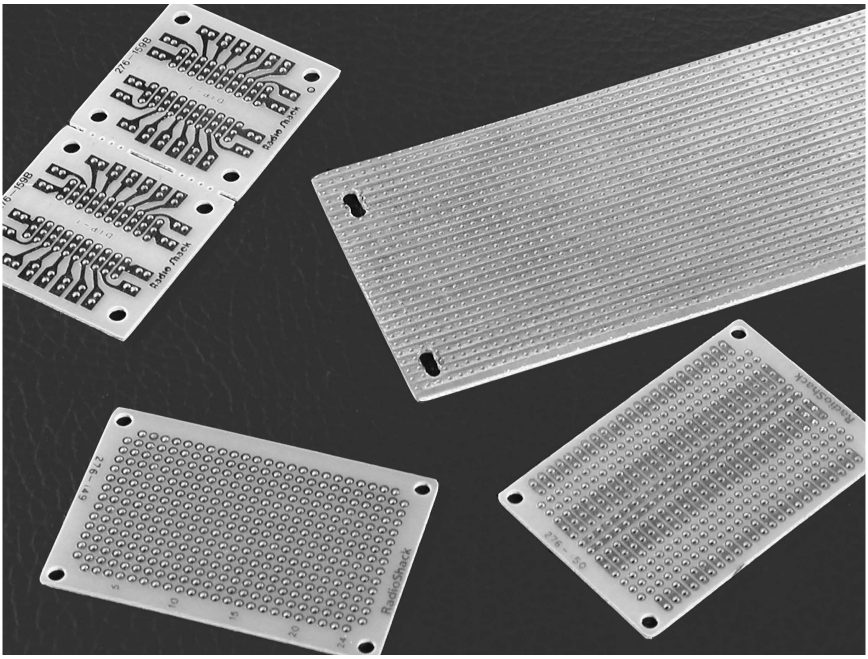


Figure 14.1 Assorted printed circuit boards, showing solder side.

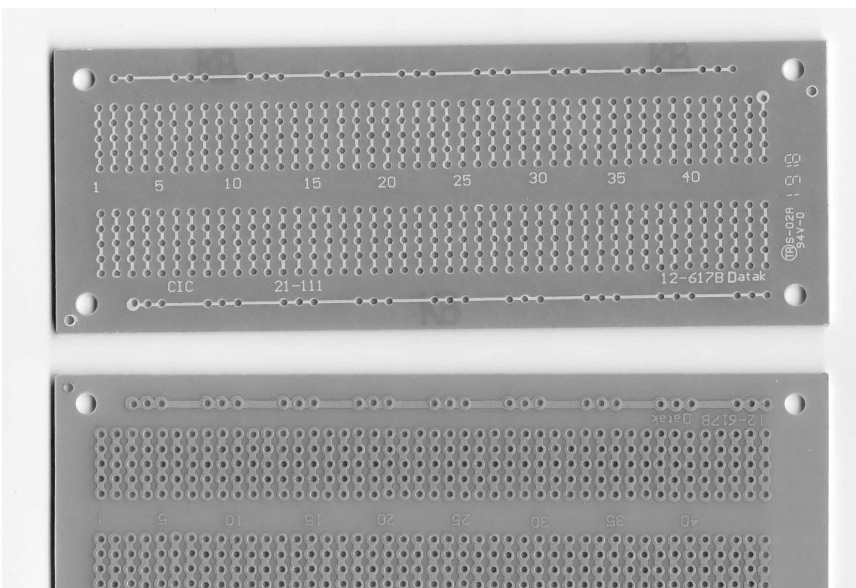


Figure 14.2 Printed circuit board mimicking breadboard, showing solder side and component side.

removed, we will solder in a socket instead and wait to insert the chip itself until after we have finished all our hot work. If for some reason the chip doesn't work (or fails later), we can remove and replace it as easily as on our plastic breadboard.

Place the socket on the side of the board that does *not* have the copper paths and pads (this is called the *component* side—the other side is the *solder* side). Push the pins gently through the holes, ensuring that the socket positioned like the chip is on the breadboard, straddling the middle, with a row leading out from each pin. Make sure all the pins go through fully and none are bent over between the socket and the board. Solder the socket pins carefully: avoid letting blobs of solder short together adjacent pins or copper traces (Figure 14.3).

Now collect a full identical duplicate set of resistors, capacitors, and other components that you used on your breadboard. Don't be cheap:

Rule # 19: Always leave your original breadboard design intact and functional until you can prove that the soldered-up version works.

This makes it much easier to debug any mistakes by comparing the working version on the breadboard with the miscreant on the circuit board.

Bending the leads as necessary, place the parts, one by one, on the component side of the board. Insert the leads into the appropriate holes from the component side of the board, solder them into place on the solder side, and clip off the excess wire before going on to the next part. Follow their placement on your breadboard *exactly*. You can press resistors, capacitors, etc. snugly down against the circuit board, rather than leaving them waving in the air like you probably did when breadboarding.

Use thin, insulated *solid* wire to interconnect components (strip insulation off the ends as you did to make the jumpers for the breadboard), or use bare wire if there is no danger of wires shorting against each other or any component leads. Link circuit traces by running the wire jumpers along the *component* side of the board, passing the ends through holes in the appropriate strips, and soldering them to the pads on the solder side (Figure 14.4). Make sure the uninsulated bits of wire and component legs do not

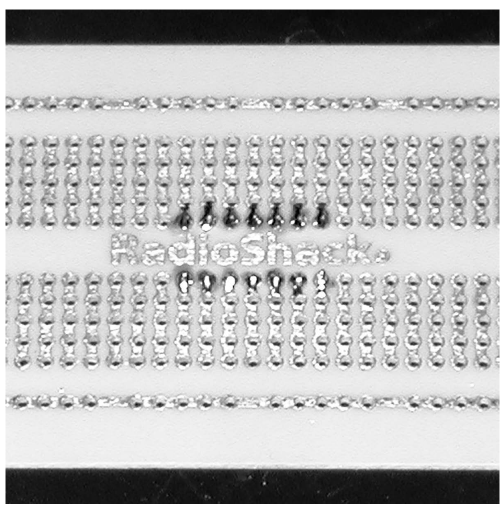


Figure 14.3
Socket soldered to PC board, solder-side view.

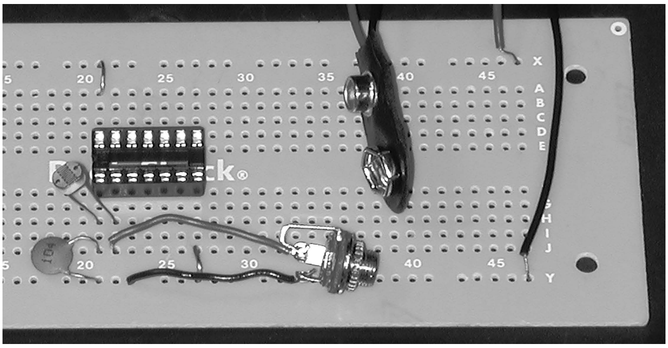


Figure 14.4
Completed circuit, waiting for the chip.

short against each other, and solder carefully to avoid blobs that make unintended connections. Use solid wire for the on-board jumpers (since it's easier to work with), but stranded wire is better for attaching the pots and jacks because the wire can flex more easily without breaking when you mount the circuit in a case. Don't forget that every jack needs signal connection and a ground wire (Rule #10). Depending on how you plan to package your circuit, you may want to extend the leads of any photoresistors with stranded wire so that you can mount them in holes through the box (wrap some electrical tape around the bare legs to keep them from shorting).

If you use a circuit board that mimics a breadboard closely (like the Datek board), it will have at least one horizontal bus at the top and bottom of the board that you can use just like on the breadboard. If you are working with non-clones, however, you may have to create your own satellite mini-buses for grounds, as explained in the previous chapter. If you made a multi-channel oscillator and the circuit board only has one horizontal bus along each edge, you may need to use *both* buses as grounds for the capacitors, in which case you should connect the +9 volts directly to pin 14.

As mentioned earlier, the Hex Schmitt Trigger has a very low current drain and will run a long time on a single 9-volt battery. But unless your uncle works for Union Carbide, you'll still want to turn the thing off if you're not using it for a while. You can always just disconnect the battery from the clip when you put the circuit to bed. But if you prefer a real on/off switch:

1. Pick up a toggle switch.
2. Solder the "Normally Open" terminal to the +9-volt line (red wire) of your battery clip.
3. Solder a wire from the "Common" terminal of the switch to the +9-volt bus or directly to pin 14.
4. Finish by soldering the ground wire (black) from the battery clip directly to the ground bus on the board.

After all soldering is finished, carefully insert a chip in the socket, and double-check to make sure its orientation is correct (i.e., pin 14 goes to +9 volts, pin 7 to ground, not vice versa). Look to see if you made any unintentional "solder bridges" between traces. (After transferring your first design from breadboard to circuit board, you will see how important it is to have a good soldering iron with a fine tip.) Compare your connections against the breadboard one more time before you connect the battery and turn on the circuit. If the battery or chip gets hot when the circuit is on, shut it off immediately and check again for mistakes.

If your soldered circuit doesn't seem to work, compare the placement of every part and wire between the breadboarded version and the soldered one—look sharp, since it's easy to miss a connection that's off by one hole. Check for wires you pressed though the board but forgot to solder, as well as blobby gray cold solder joints. Make sure there are signal and ground connections for the audio output. If it makes sound but is much quieter than the breadboarded version, or just acts weirder than it should, check to make sure you remembered to hook up the battery's "+" and "−" connections to pins 14 and 7 of the chip. Make sure that none of the copper traces have torn from repeated soldering and de-soldering or the strain of wires

A tip for keeping this (and any other) circuit running cleanly is to solder a 0.1 uf capacitor between the "+" pin (14) and the "−" pin (7), keeping it as close to the pins as possible (not at the other end of the board and linked by wires). This "decoupling" capacitor helps filter noise that can spread through power supply connections.

A circuit board that mirrors your breadboard exactly makes the transfer process much easier. If you can't get such a board, you must make adaptations carefully, checking your connections as you go (it helps to draw out your layout in advance of soldering). Once you have transferred a few designs to clone boards, and have gotten comfortable with the "topology of circuitry," you can advance to other patterns of circuit board that give you the freedom to rearrange your designs between breadboard and final version, sometimes shrinking them down to fit in smaller boxes (Figure 14.5).

When your circuits start to get complicated, you may find that regular hookup wire is too thick and messy on the board. To lighten up, and move from bucatini to capellini, buy yourself a roll of what is called "wire wrapping wire." At 30 gauge, it's

Figure 14.5 A boxed photoresistor-controlled hex oscillator.

real thin, stays in place when snaked around the board, and comes in nice, bright, child-friendly colors that will cheer you through the ordeal of soldering.

Once you've confirmed that everything works, you can move on to finding a box and drilling a mess of holes (see "Jack, Batt, and Pack" on website). That's the fun part (but remember to insulate the circuit board from any metal surfaces!).

A PROVISO

There is one potential complication to Rule #19: some electronic components—capacitors in particular—have sufficiently wide tolerances that two parts with identical markings can deviate in value enough that the breadboarded and soldered versions of your circuit could behave slightly differently. If this is unacceptable, you can transfer the original parts from the breadboard to the circuit board, one at a time, as you solder, rather than working with a duplicate set. In this case debugging is greatly helped by taking a few photos of the functional breadboard before you start the transfer (keep your cell phone handy). A detailed drawing is sometimes even more helpful: the act of putting pencil to paper often helps fix the design in your mind (like taking notes during boring lectures or meetings).

SOMETHING SPECIAL

Generic prototyping boards like the Datek we used in this chapter are not the only options for finalizing your design. You can design your own printed circuit board using software and either route it yourself in a fabrication lab or send the design out to a company that will produce a "professional" bespoke board, just like inside your phone or computer. In Chapter 32 you'll learn how to do this. Alternatively, in Chapter 16 you'll weave electronic components into cloth to produce a "soft" version of your favorite circuit.

NOTE

1. Adafruit sells a classy, Altoid-tin-size version of this design as well: https://www.adafruit.com/product/1609.

CHAPTER 15

Getting Messy
Modulation, Feedback, Instability, and Crickets

You will need:

- A breadboard.
- One CMOS Quad NAND Gate Schmitt Trigger Integrated Circuit (CD4093).
- A CMOS Hex Schmitt Trigger Inverter (74C14, CD4584, or CD40106).
- Assorted resistors, capacitors, pots, and photoresistors.
- A piezo disk.
- A matchbox or balsa airplane.
- Some solid hookup wire.
- A jack to match your amp.
- A 9-volt battery and connector.
- An amplifier.
- Hand tools.

The Hex Schmitt Trigger presents a great introduction to making musical circuitry: with just a handful of components and a few minutes of time, you can throw together reliable oscillators whose pitch can be easily swept over a wide range. It also opens the door to the slightly warped world of making sound with digital logic, rather than traditional analog circuitry. With another chip from the same CMOS family, we can implement some more advanced control functions commonly associated with classic analog synthesis.

GATED OSCILLATOR

The Schmitt Trigger circuit element that turns each inverter in the 74C14 into a snappy oscillator is also found in other CMOS chips. Most useful is the CD4093 Quad NAND Gate (see Figure 15.1)

This chip contains four identical NAND gates. There are two gates on each side of the chip. Each gate has two inputs (on the flat side) and one output (on the rounded end). Unlike the spawning salmon of the 74C14, these gates are arranged in mirror

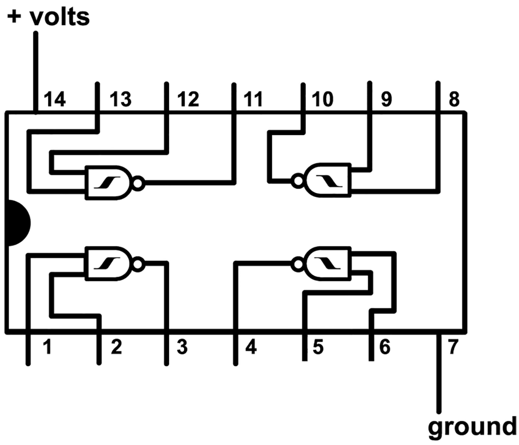


Figure 15.1
CD4093 Quad NAND Gate pinout.

symmetry, like rutting elks: the outputs of each gate face each other, rather than the same direction. Note that this chip has the same power connections as the hex inverter chip we used in the previous chapters: + voltage to pin 14, ground to pin 7. All of which brings us to an important new rule:

Rule #20: All chips may look alike on the outside without being the same on the inside—read the fine print!

A NAND gate is a variant of the basic binary function of an AND gate, whose "truth table" follows:

Input A	Input B	Output
0	0	0
1	0	0
0	1	0
1	1	1

You can see that the output only goes true/1 when both inputs are true/1—democracy in action: we go to the zoo because it's the place *both* kids agree would be fun.

A NAND (NOT+AND) gate adds an inverter stage after the AND logic to flip the output to the opposite state, like this:

Input A	Input B	Output
0	0	1
1	0	1
0	1	1
1	1	0

Democracy is replaced by contrarian despotism: dad *avoids* turning off the highway to visit Mammoth Caves specifically because both kids have been whining to see it for 200 miles.

This added inverter stage introduces the principle of knee-jerk denial (discussed in Chapter 13) that transforms this logic circuit into a gateable oscillator.

Breadboard the circuit shown in Figure 15.2. Note that the basic design has similarities to our earlier oscillator: a capacitor between an input and ground; a feedback resistor from the output back to that input. But where each stage in the hex inverter package had one input, each NAND gate has *two* inputs. Because of the combinatorial logic of the NAND gate (explained prior), the second input of the gate can be used as a *control* input to turn the oscillator on and off: the output of the circuit will only change state (i.e., oscillate) when the control input is held "high" (+9 volts). If you connect the second input of the gate to ground (0) the circuit stops oscillating and the output remains in a "1" state (Figure 15.3), silent.

Try both configurations to confirm the logic. Plug a bit of wire into the breadboard near the gate input (pin 1 in Figure 15.2 and Figure 15.3) and connect the other end first to the + bus and then to the ground. (If you leave the other end dangling in the

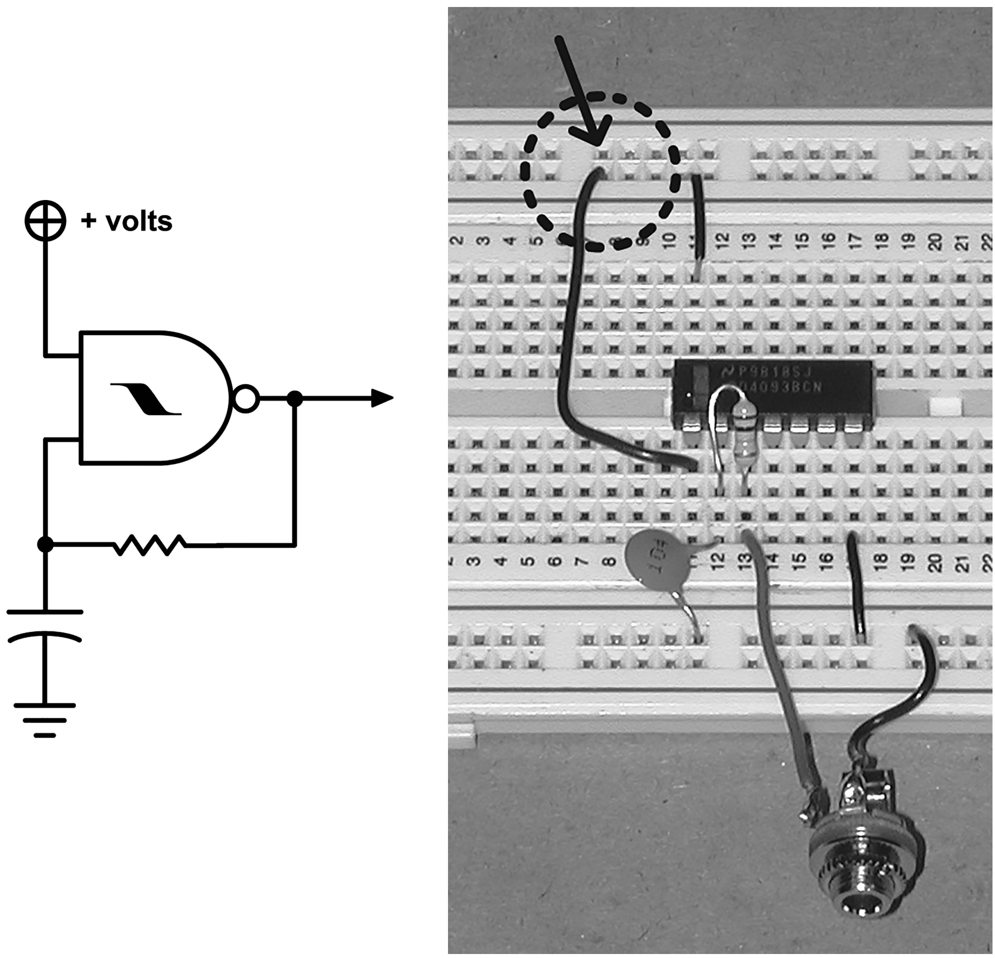


Figure 15.2 Basic NAND oscillator, enabled.

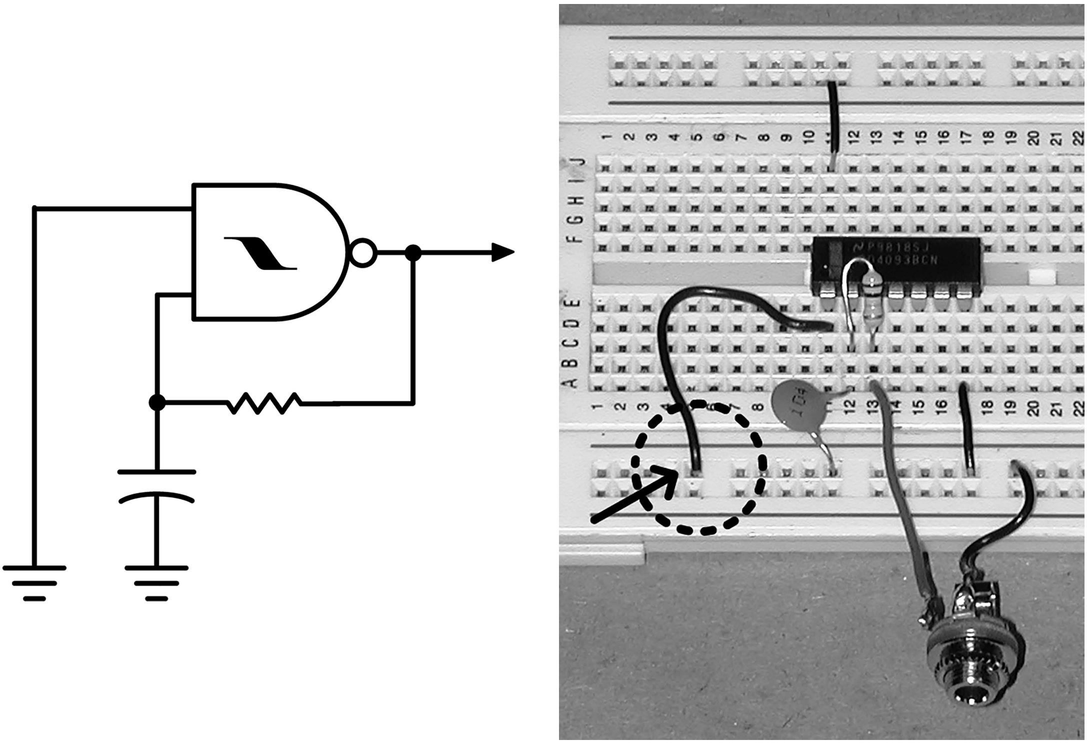


Figure 15.3 Basic NAND oscillator, disabled.

air the circuit may behave rather erratically—more on this later, but don't worry, it's not dangerous; go ahead and try this.) The two inputs to each gate are identical—it doesn't matter which you use for the control input (the one with the wire moved between + and ground) and which for the feedback input (the one connected to the capacitor and feedback resistor), as long as you don't mix up the two and connect the capacitor to one and the feedback resistor to the other. And you can use any of the four NAND gates on the chip—they function identically—although the figures show just one of four possible hookups.

Note that even though a control input (pin 1 in the figures) might be connected to +9 volts, we still need to connect +9 volts to pin 14 and ground (− voltage) to pin 7. The connections to the supply voltages have two distinct functions in our new circuit: through pins 14 and 7, they provide *power* to the chip, needed to run its internal operations (the gas); but + and − voltage also have *logical* value and are evaluated as part of the (admittedly simple) mathematical calculations that the circuit performs in order to oscillate—that's the connection to pin 1. As with the hex inverter in Chapter 13, sometimes this chip will make sound without proper power connections, but it will be "coasting" and probably will not perform reliably or predictably. Which brings us to Rule #21:

Rule #21: All chips expect "+" and "−" power connections to their designated power supply pins, even if these voltages are also connected to other pins for other reasons—withhold them at your own risk (or entertainment).

MODULATION

The oscillator oscillates when the control input is connected to +9 volts; it turns off when the control input is linked to ground. Big deal, you say, we can do this by simply connecting and disconnecting the battery. But, because an oscillator's output consists of a square wave swinging between +9 volts and ground, we can also use the output of one oscillator to switch *another* oscillator on and off. Breadboard the circuit shown in Figure 15.4. Use a large capacitor (2.2–10 uf) and 100 kOhm–1 mOhm pot for Oscillator 1 (shown here using pins 1, 2, and 3) and a 0.1 uf capacitor and a pot or photoresistor (as here) for Oscillator 2 (using pins 4, 5, and 6).

The control input on Oscillator 1 (pin 1) is tied directly to +9 volts, so it runs all the time, as we demonstrated earlier in Figure 15.2. But the control input of Oscillator 2 (pin 6) is connected to the output of Oscillator 1 (pin 3), which gates Oscillator 2 on and off as it swings between 9 volts and ground. If Oscillator 1 (the control oscillator) has a large capacitor and runs slowly (like a metronome), you can hear Oscillator 2 (the modulated oscillator) switch on and off at a regular tempo. As we tune Oscillator 1 higher and higher, this obvious on/off function transforms into a kind of modulation that is heard as a change in the timbre rather than tempo.

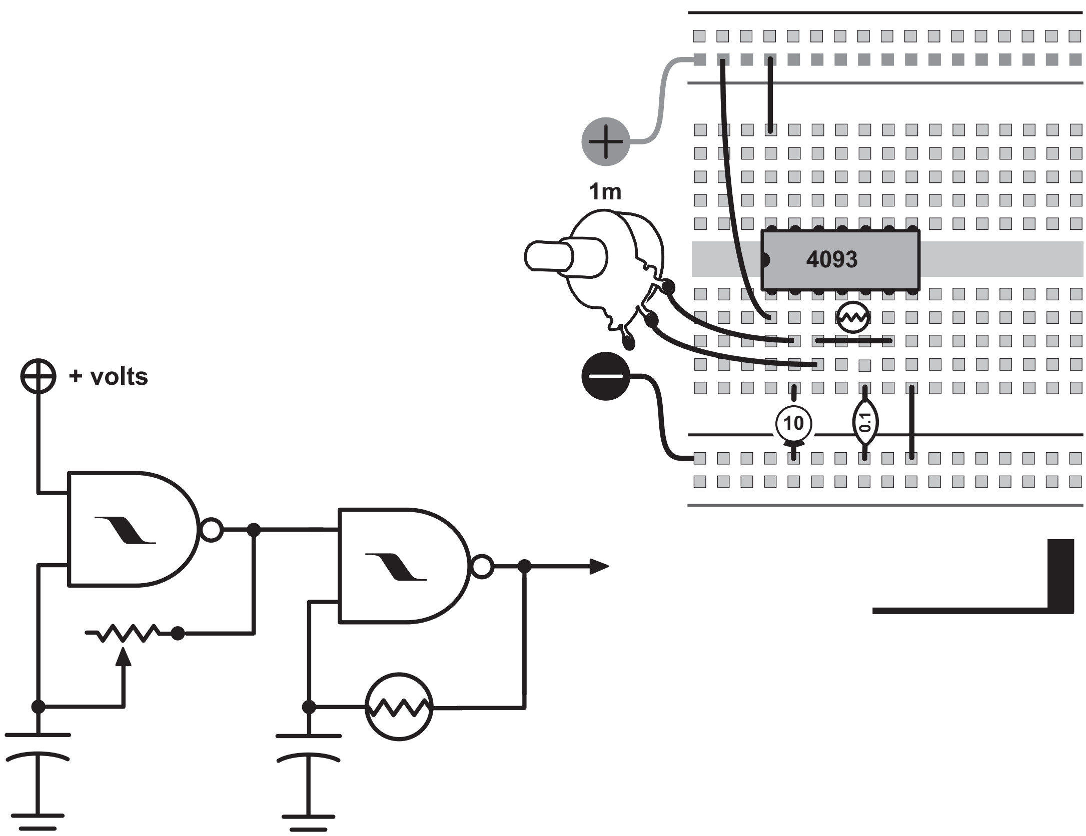


Figure 15.4 A gated oscillator.

Very cool flangey/wah-wah noises occur when the control oscillator and the modulated oscillator are both in the audio range and are close to one another in frequency—try using identical 0.1 or 1.0 uf capacitors and photoresistors for both stages. Now substitute 10 uf capacitors and 1 mOhm pots for both stages: careful tuning of the pots results in interesting polyrhythms.

You can cascade three or four oscillators (see Figure 15.5) to create tone clusters or rhythmic patterns, depending on capacitor sizes. The timbres and rhythms get more complex with each stage you add, but with diminishing returns: you can insert a second 4093 chip into your breadboard to add four more stages, but it might only yield a minor change in the timbre. You'll have to configure parallel or satellite ground buses for the gates on the top half of the chip, as we did for the multi-voice oscillators in Chapter 13 (Figures 13.20 and 13.21).

Experiment with different value capacitors and pots for the different stages. You can use photoresistors, electrodes, or any of the other alternative resistors discussed in Chapter 13.

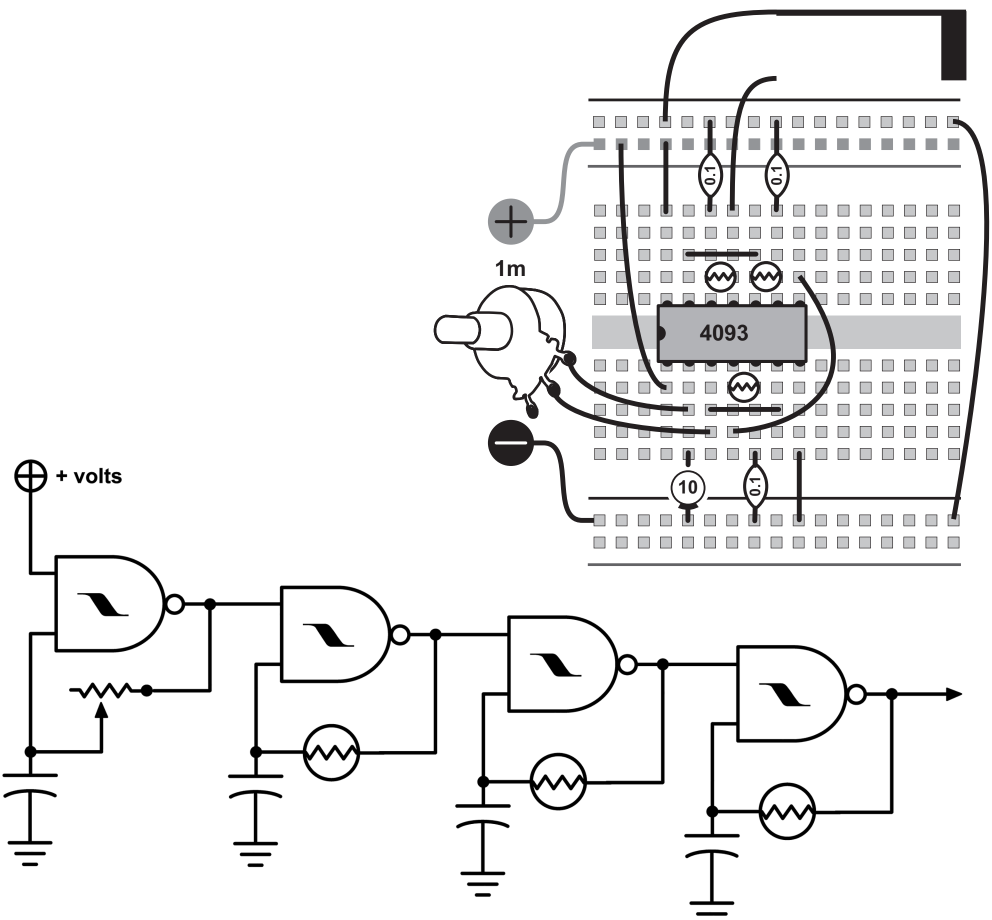


Figure 15.5 Four cascaded gated oscillators.

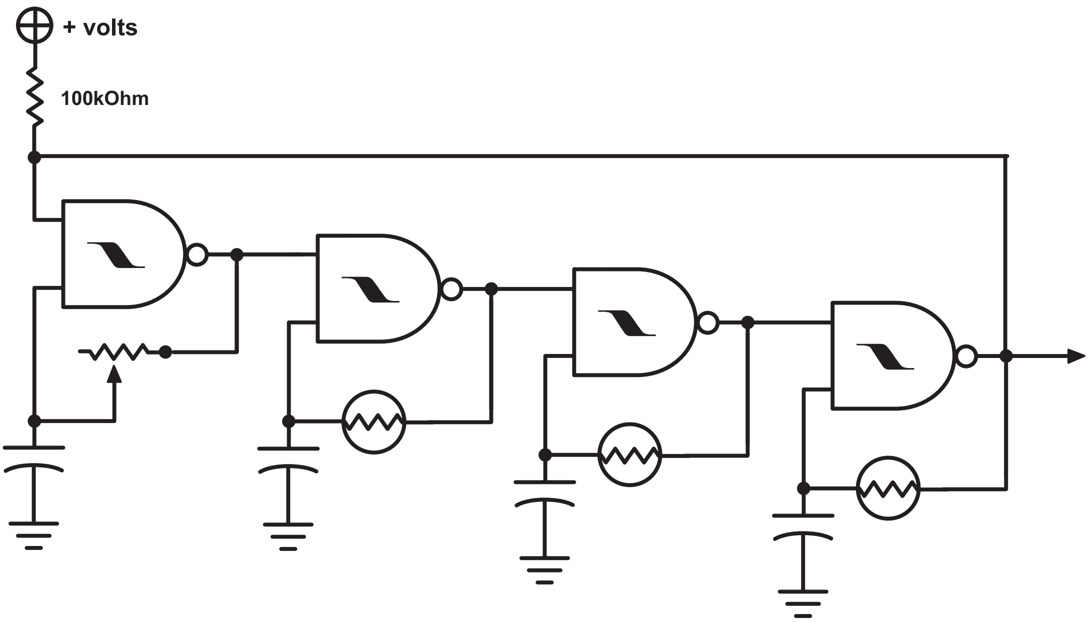


Figure 15.6 Four NAND gates with feedback connection from output of last stage to control input of first (note pullup resistor).

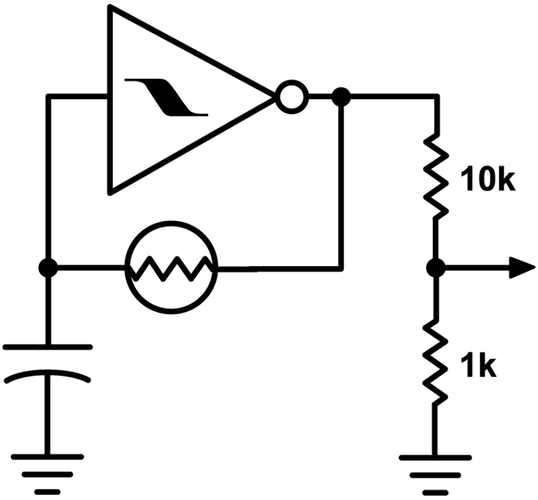
You can ratchet up the complexity of the sound by adding feedback from the output of the last stage to the control input of the first (Figure 15.6). This works best if you observe the following guidelines:

- You must "bootstrap" the control input (pin 1 in the figure) by connecting a large resistor (around 100 kOhm is good) to +9 volts—without this the oscillator might never start running.
- The feedback only works around an *even* number of cascaded stages—i.e., after two NAND gates, four gates, six gates, etc.
- Experiment with different values for the capacitors and resistors.

VOLUME CONTROL

Because these oscillator circuits put out such a hot signal (both the 74C14/40106/4584 and 4093 designs), they may overload the input to your amplifier or mixer, causing distortion (not always a bad thing—see Chapter 19) or limiting the useful range of your faders. If you want to drop the level down to a volume that matches your other line-level equipment (like your computer interface), try adding the circuit shown in Figure 15.7 to the output of each oscillator or, if you are mixing multiple signals, after the summing resistors. The volume is reduced by the ratio of the two resistors: 10,000/1,000 = 10, so the volume is divided by 10.

This deceptively simple add-on to any oscillator forms the basis for more dynamic circuits.



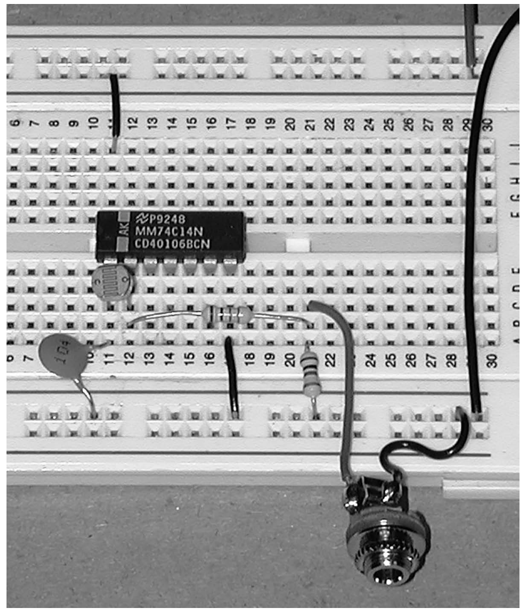
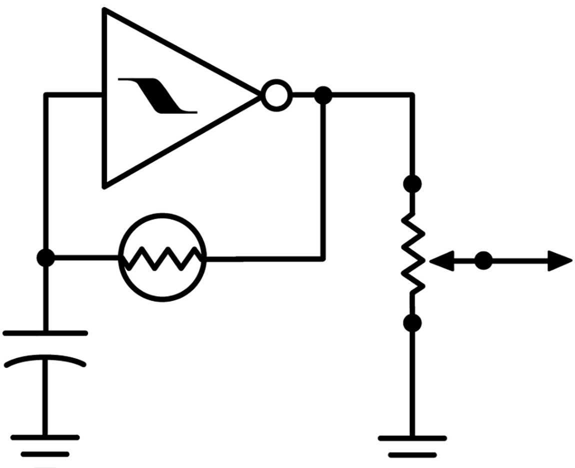


Figure 15.7
A volume-dropping circuit.



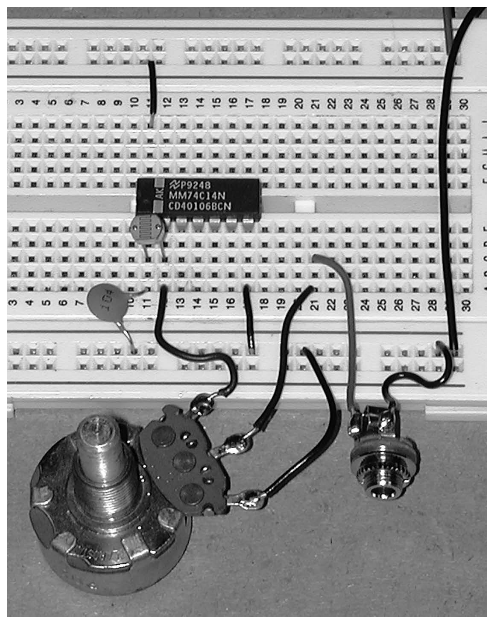


Figure 15.8
Output volume control.

For example, if you want to fade the volume up and down, rather than just drop it down to normal line level, you can substitute a pot for the resistors, as shown in Figure 15.8 (the two fixed resistors in Figure 15.7 can be thought of as a pot frozen in one setting). Any pot whose value is 10 kOhm or greater will do. Connect one ear of the pot to the oscillator's output, connect the other ear to ground, and the nose becomes your new output, with variable volume.

You may notice with this volume control, or when feeding your oscillator directly into some mixers or amplifiers, that at one extreme of the volume the pitch changes slightly. This is a drawback of the very simple design of the oscillator and a function of the loading effect of the circuit that follows it. You can usually fix this problem by putting a reasonably large fixed resistor (say 10 kOhm) between the oscillator output and the ear of the pot, rather than connecting the pot directly to the inverter. Your maximum loudness will be decreased slightly, but the pitch should be more stable.

A simple Theremin-like instrument can be made using photoresistors to vary both the *pitch* and the *level* of your oscillator. As you can see in Figure 15.9, we've replaced the 10 kOhm resistor in the volume-drop circuit in Figure 15.7 with a photoresistor

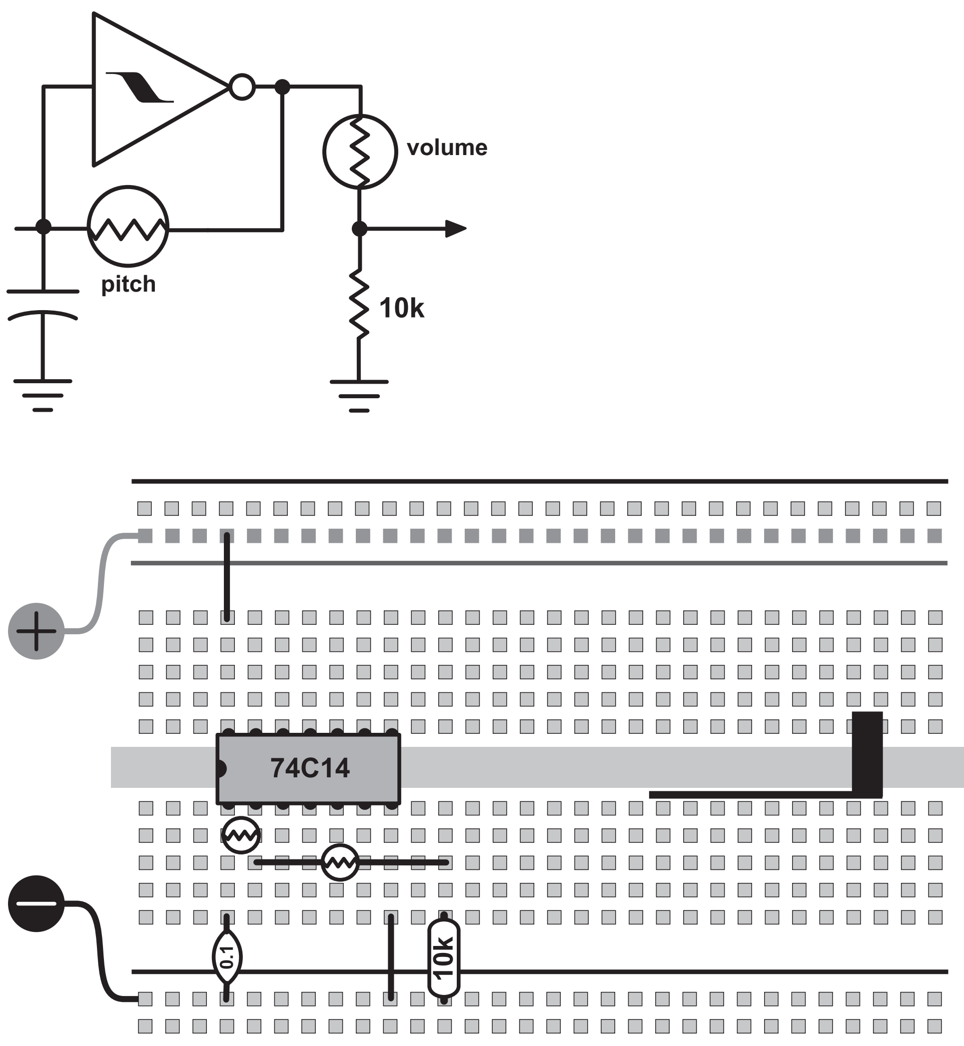


Figure 15.9 Theremin-esque circuit, with photoresistors for control of pitch and volume.

and upped the value of the 1 kOhm fixed resistor to 10 kOhm. The value of the fixed resistor determines the dynamic range—experiment with various values between 1 kOhm and 10 kOhm. When less light strikes the output photoresistor (i.e., when you move your hand closer), the volume gets quieter. This simple circuit will never mute the sound completely, but it can provide expressive control of dynamics. The physical arrangement of the two photoresistors is critical to the effectiveness of this circuit: try gluing the two down, about a hand-span apart, on the top surface of an opaque plastic or wooden box holding the circuit; make sure you have a direct, stable source of light from above.

FILTERING

These oscillator circuits aren't just loud: they are very bright—as befits the squarish waves they generate. (I confess that the signal isn't a perfect square wave—if you were to look at the waveform on an oscilloscope you'd notice that it does not have the requisite perfect 50% duty cycle but—thanks to Herr Schmitt—is slightly asymmetrical, which is what gives these circuits their distinctive sonic charm—however, it's close enough that we can use the term "square" to save a few keystrokes.) You can always throw in an external device to tweak the timbre: as mentioned earlier in the book, a 10-band graphic EQ footpedal is an excellent accessory for any hacker; you can also use the equalization on your amplifier or mixer. But with the addition of a few more common parts, you can modify the timbre before the signal leaves your circuit board.

The waveform exiting an inverter or NAND gate may be pretty square, but after feeding back through a resistor and capacitor to the input, it sounds (and looks, on an oscilloscope) astonishingly much like another synthesizer classic: the triangle wave. The overtones of a triangle wave are considerably quieter than those of its rectilinear brother, and consequently it has a much mellower tone. Unfortunately, connecting an output jack directly to this input point interferes with the delicate balance that keeps the oscillator swinging (think Heisenberg Uncertainty Principle) and can mute the circuit or shift the pitch. So we must isolate the circuit from the outside world with a big resistor, such as the 100 kOhm shown in Figure 15.10. Moreover, since the triangle wave "floats" on a DC bias equal to one-half the supply voltage, we add a capacitor (here 10 uf, but the value is not very critical) to block the voltage from reaching your amp or mixer (like we did to block the electret power supply from reaching the mike preamp back in Chapter 11).

We can use the square wave and triangle wave outputs separately and simultaneously, but when you hook up this circuit, you'll notice that the triangle wave is considerable quieter than the square wave if the latter is taken directly from the output of the inverter: the combined effect of the isolation resistor and the inherent behavior of the chip's input stage drops the triangle wave level close to the typical line level. So we'll use the same level-dropping circuit from Figure 15.7 to balance the loudness of the two waveforms (you can eliminate this if you don't mind having the square wave be much louder than the triangle).

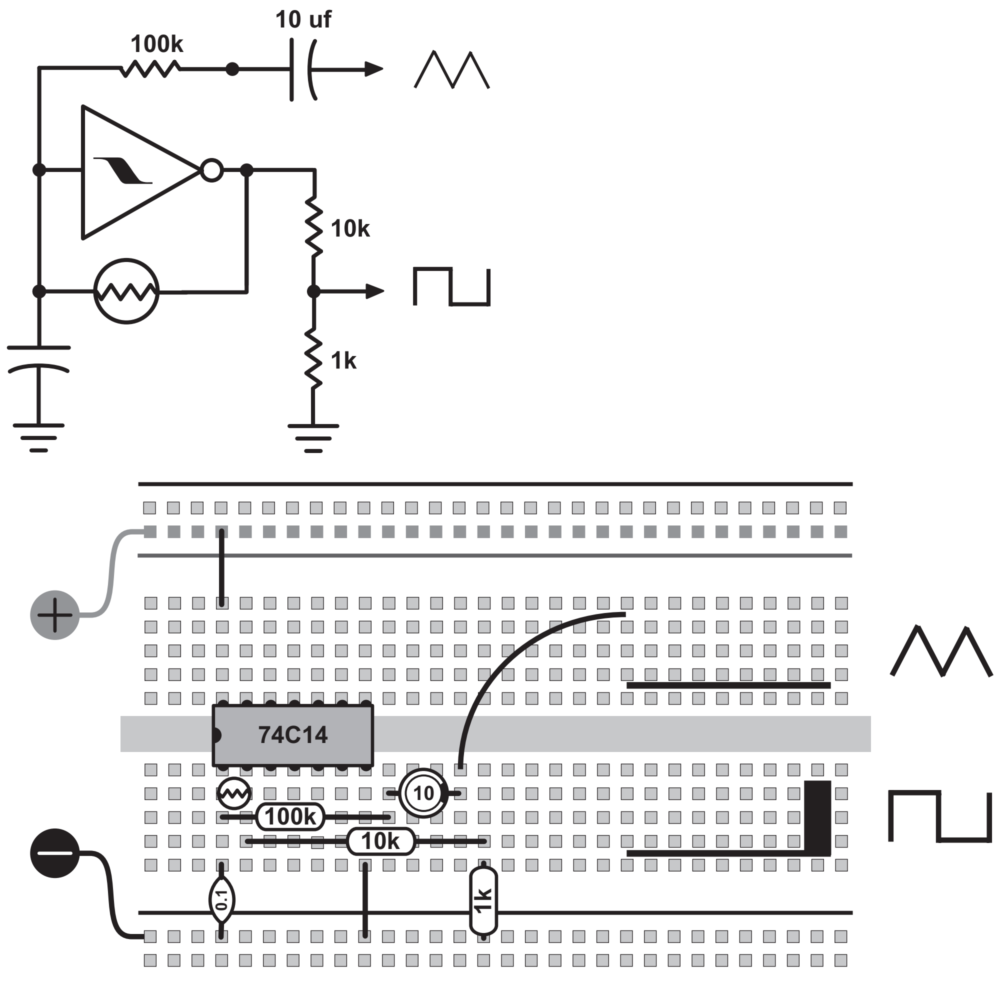


Figure 15.10 Adding a triangle wave output to a basic oscillator (with square wave level dropped to match triangle wave).

Another solution to the excessive brightness of the square wave is to *filter* it, instead of taking the triangle wave output. The circuit in Figure 15.11 is a simple fixed frequency "low-pass" filter: it rolls off all the overtones above its cut-off frequency, which is determined mostly by the values of Cf and Rf. Good starting values are 0.1 uf for Cf and 10 kOhm for Rf. Substituting a smaller capacitor makes the circuit brighter while a larger one will roll off more highs. You can also change the value of Rf: as with the capacitor, a smaller value raises the cut-off frequency while a larger one lowers it. This brings us to our next circuit.

If we substitute a photoresistor for the fixed resistor Rf of the previous circuit, we can use light to control the filter cut-off frequency (Figure 15.12). Start by setting Cf equal to the capacitor value used to set the range of the oscillator (i.e., 0.1 uf), and experiment with smaller values for a brighter tone, larger ones for a darker one. If both

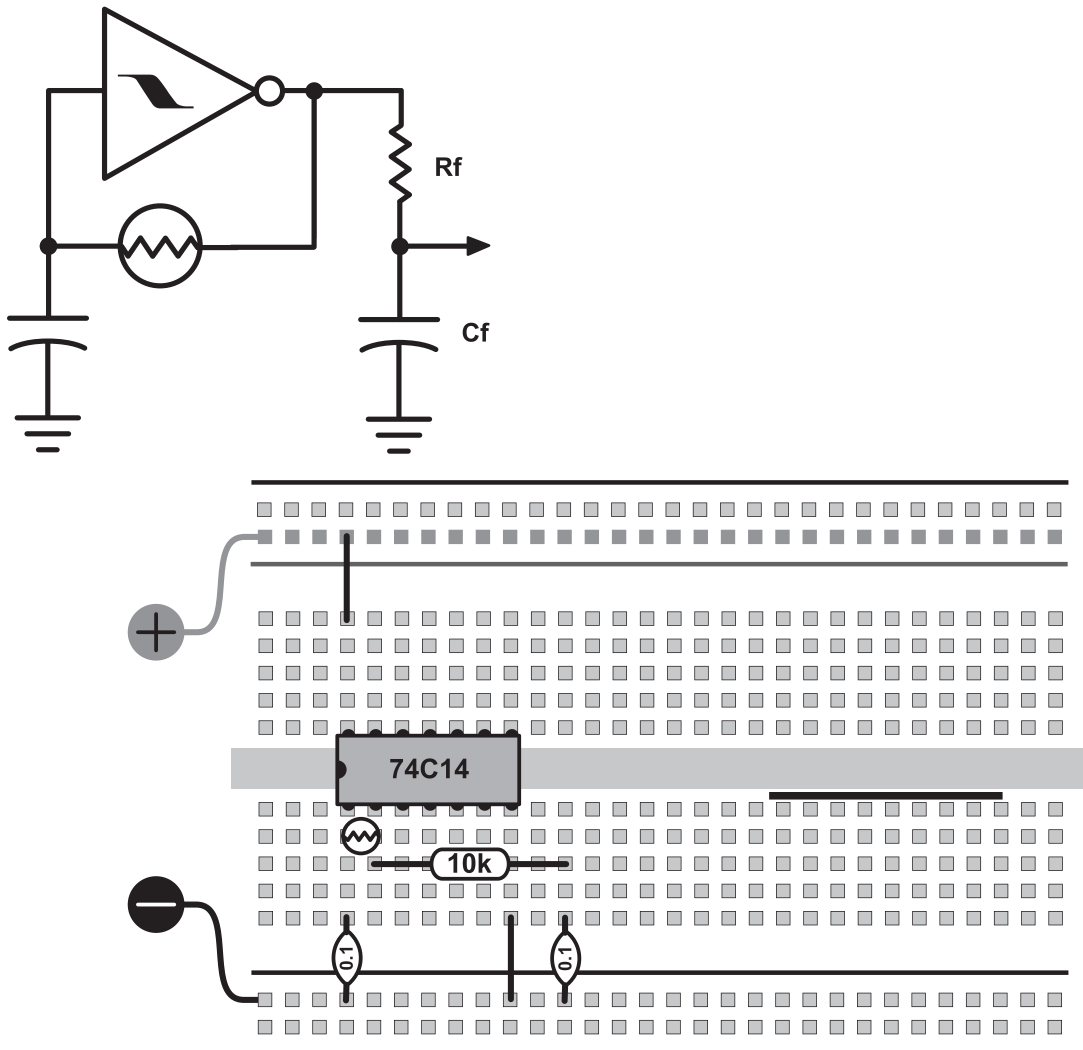


Figure 15.11 Simple, fixed frequency low-pass filter.

photoresistors receive the same amount of light (if they are immediately next to each other, for example), the filter will "track" the frequency of the oscillator—i.e., the filter will go up when the oscillator sweeps up and down when the oscillator goes lower. If the photoresistors are placed far enough apart, on the other hand, you can play the pitch and filter separately. As with our pseudo-Theremin, you need to think about the physical layout of your instrument. You can substitute a potentiometer for the photoresistor that controls the filter in this circuit if you want more direct control—100 kOhm–1 mOhm is a good starting value.

Before any "real" engineers start accusing me of leading my hackers astray, let me explain that most filters and equalizers are dauntingly complex circuits. I have simplified them greatly for inclusion here. You might notice that altering the cut-off

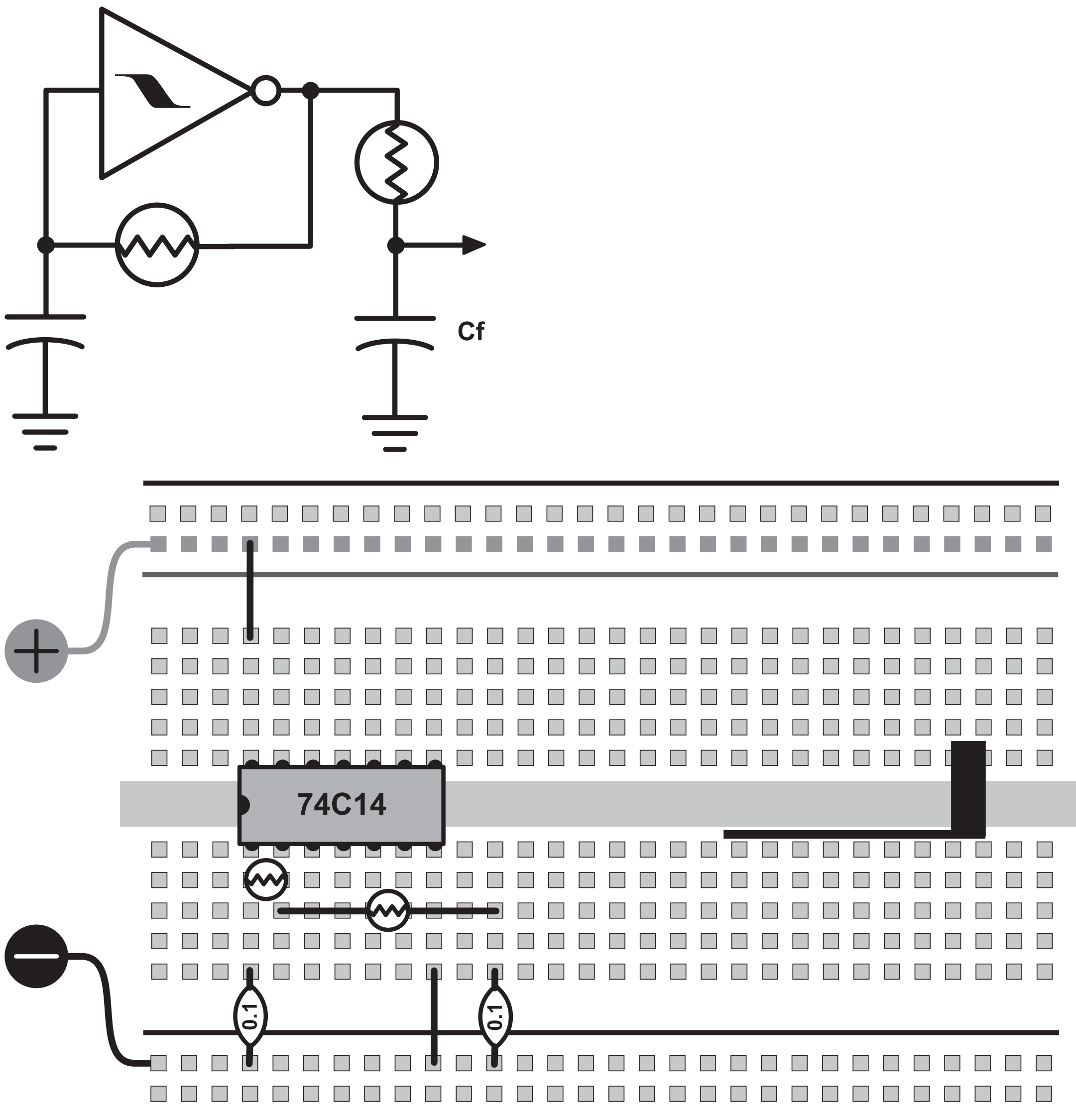


Figure 15.12 Low-pass filter with photoresistor-controlled cut-off frequency.

frequency can affect volume as well. Sometimes the behavior of the filter is affected by the "load" of the circuit that follows it (i.e., your amplifier or mixer). But they sound pretty good nonetheless. Trust me, a mongrel EQ with a slight limp can prove as true a companion as a purebred Moog filter.

I've shown the volume control and filter circuits as applied to the hex inverter oscillator, but these methods can be added to the output of the 4093 NAND circuits as well. The Theremin and variable low-pass circuits are especially effective on cascaded 4093 oscillators.

RANDOM WALKS

Once you get the hang of these basic circuits and the way they are *supposed* to work, don't be afraid to experiment with alternate configurations, even if you have no idea what you're doing (see Rule #18). It is almost impossible to destroy the chip by making "wrong" connections between its various pins (unless you connect the battery backwards); at worst, certain configurations will be mute.

Try random connections and component substitutions. Create multiple feedback paths by linking the outputs of some oscillators to the inputs of others—you can do this with plain wire, resistors, diodes, capacitors, pots, photoresistors, etc. Add electrode touch contacts to set up feedback paths through your skin. You may need to add pullup resistors to the various inputs of the circuits to start them oscillating, as we did in Figure 15.6: connect a resistor in the 100 kOhm–1 mOhm range between the input pin and +9 volts where needed. Arranged in matrices, the oscillators produce unstable, complicated patterns of pitch and rhythm not displeasing to the ear (or brain). When you hear something good, stop and make very careful notes of what is wired to what because you may never find it again. And probably no one will ever be able to explain why it sounds the way it does.

An excellent route to instability is "voltage starving" your circuit. Instead of hooking the 9-volt output of the battery directly to the circuit, connect it through a low value pot, as shown in Figure 15.13. As you increase the resistance of the pot by turning it, you diminish the voltage reaching the chip. At one extreme of the rotation (fully CCW as shown in the figure), the circuit gets full voltage and should operate "normally." At the other extreme (fully CW), there should be so little voltage reaching the chip that it falls completely silent. In between the two limits you should find a range of settings at which the circuit "almost works" and might produce curious sounds. It's important to find the value of pot that gives you a decent range of behavior. Start with 1 kOhm. If this has no effect no matter what the setting of the pot, then go up to 5 kOhm; if, on the other hand, the 1 kOhm pot is too sensitive, try 500 Ohm.

Voltage starving is especially effective on more complex circuits, such as the cascaded 4093 chains, with feedback, since different elements of the circuit often collapse at different voltages (it's also very useful in circuit bent toys—see chapters on website).

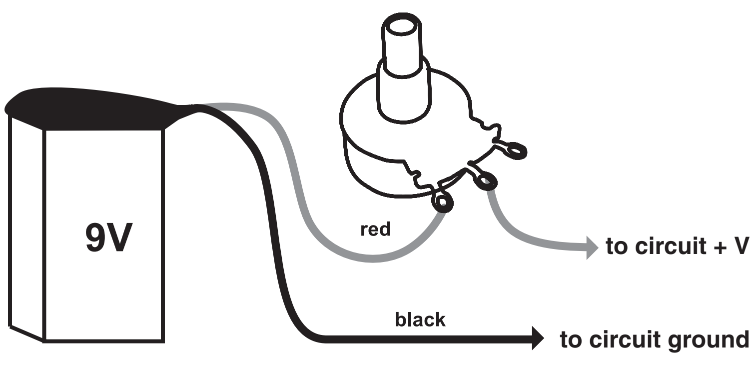


Figure 15.13 Voltage starve circuit.

POWER STRUGGLES WITH CRICKETS

As I mentioned earlier, CMOS Integrated Circuits consume tiny amounts of power, which makes them well suited to batteries. Power consumption is directly proportional to the frequency of the oscillator: the chip only uses power when it changes state from low to high or back again. The lower the frequency, the less power used in a given period of time—a metronome built with this chip can run for a year or more on one battery while a high-pitch audio squealer will go through batteries faster.

The miserly power consumption of these chips manifests itself in their output signal: although the output is very "hot" (swinging 9 volts peak-to-peak, compared to 0.7 volts of a typical mixer output), it carries very little current. This means you have to plug the circuit into a power amplifier in order to drive a speaker—if you wire the oscillator output directly to an ordinary speaker, you will probably hear nothing since it does not produce enough current to move the coil. Piezo disks, on the other hand, require very little current to function as (admittedly low fidelity) speakers, and these CMOS chips can drive piezos directly, without need for an additional amplifier. Simply connect the oscillator's ground bus to the metal substrate of the disk and connect its output to the white center of a disk. Hook up the battery and sweep the oscillator through its range.

Low-frequency pulses yield a pleasing clicking sound while audio pitches buzz like insects and frogs on a sultry summer evening. If it's too quiet for your taste, clamp the piezo disk to a cookie sheet or Styrofoam cup, or place it on a wooden matchbox with a stone on top, or glue it to the wing of a balsa glider. Dutch sound artist Felix Hess has made beautiful large-scale installations with multiple small circuits pinging piezo disks (see Art & Music 5, "Drivers," Chapter 8). Laurie Anderson embedded a piezo in molded bamboo fiber to create a portable sound generator for her *Walk* installation (see "Video Music/Music Video" essay on the website). If the disk came inside one of those plastic lollipop packages, this resonator will also increase the loudness of the chirping.

COMPANION @ WEBSITE

CONTEXT

This is as good a place as any to review why I've chosen the CMOS family of integrated circuits (of which the 74C14 and the CD4093 are members) for our projects:

- They consume very little current and can run on a wide range of voltages, which makes them ideal for battery operation.
- They are rather difficult to blow up.
- They are cheap.
- They were never intended for audio applications, which adds pleasurable frisson to our experiments.

I'm not the first, nor the only, hacker to take advantage of the clandestine audio affordances of CMOS. Stanley Lunetta (1937–2016) was an early adopter, and his designs are still revered, are passed through the internet, and have influenced several companies producing modules for Eurorack synthesizers.[1] ADACHI Tomomi "Tomomiboxes" make extensive use of CMOS circuitry (see his video on the website). And much of the soft-circuitry movement has taken advantage of CMOS's battery friendliness in crafting wearable electronics (see Chapter 16).

COMPANION @ WEBSITE

ART & MUSIC 8

COMPOSING INSIDE ELECTRONICS

The 1970s were a pivotal time in the evolution of the technology and culture of electronic music. Synthesizers were still impractically expensive for young musicians, but integrated circuits—the guts of those costly machines—were getting cheaper in inverse proportion to their sophistication. New chips contained 90% of a functional circuit designed by someone who really knew what he was doing; the remaining 10% could be filled in by someone relatively clueless. The trick was finding the right chips: in the days before the World Wide Web, information was much more segregated, with precious few leaks. When data did trickle down from engineers to amateurs, through magazines with titles like *Popular Electronics* and *Wireless World*, it was often passed from hand to hand like *samizdat* literature.

A musical community formed around this exchange of information. It included the Composers Inside Electronics who worked with David Tudor (see Art & Music 4, "David Tudor and *Rainforest*," Chapter 8); students of David Behrman and Robert Ashley at Mills College in Oakland, California (including Kenneth Atchley, Ben Azarm, John Bischoff, Chris Brown, Laetitia de Compiegne Sonami, Scot Gresham-Lancaster, Frankie Mann, Tim Perkis, Brian Reinbolt, and Mark Trayle); students of Alvin Lucier at Wesleyan University in Middletown, Connecticut (Nicolas Collins and Ron Kuivila); students of Serge Tcherepnin at California Institute of The Arts in Valencia, California (Rich Gold); and other musicians and artists scattered throughout the United States and (more thinly) Europe. Some participants were mere muddlers, who built beautiful, oddball circuits seemingly out of pure ignorance and good luck. Others became astonishingly talented, if idiosyncratic, designers. The prolific Paul DeMarinis included bits of vegetables as electrical components so his circuits would undergo a natural aging process (*CKT*, 1974), incorporated sensors that responded to the weak electronic field emanated by the human body (*Pygmy Gamelan*, 1973, Figure 15.14), and built automatic music composing circuits that anticipated later trends in computer music (*Great Masters of Melody*, 1975)—one of which could be played by a bird (*Parrot Pleaser*, 1974).

The European electronic music scene of the time was much more stratified—there was a well-established state-funded tradition of collaboration between composers and professional engineers, and homemade music circuitry didn't catch on there to the degree that it did in the United States (I have never seen a photograph of Stockhausen holding a soldering iron). There were notable exceptions, however. Andy Guhl and Norbert Möslang (Switzerland; Figure 15.15) formed Voice Crack in 1972 and over the next 30 years honed their skills at "cracking" everyday

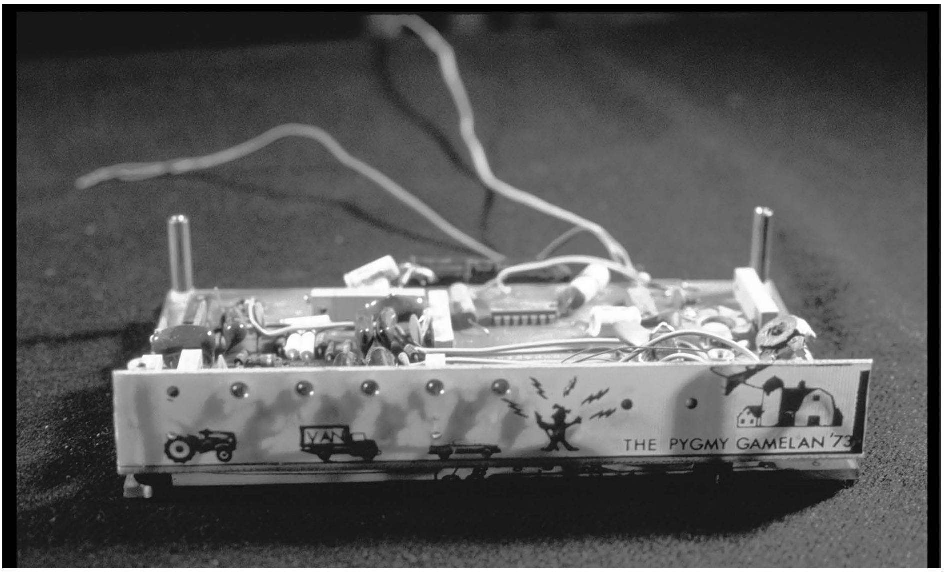


Figure 15.14 Paul DeMarinis, *Pygmy Gamelan*, 1973.

Figure 15.15 Circuitry by Norbert Möslang (Voice Crack).

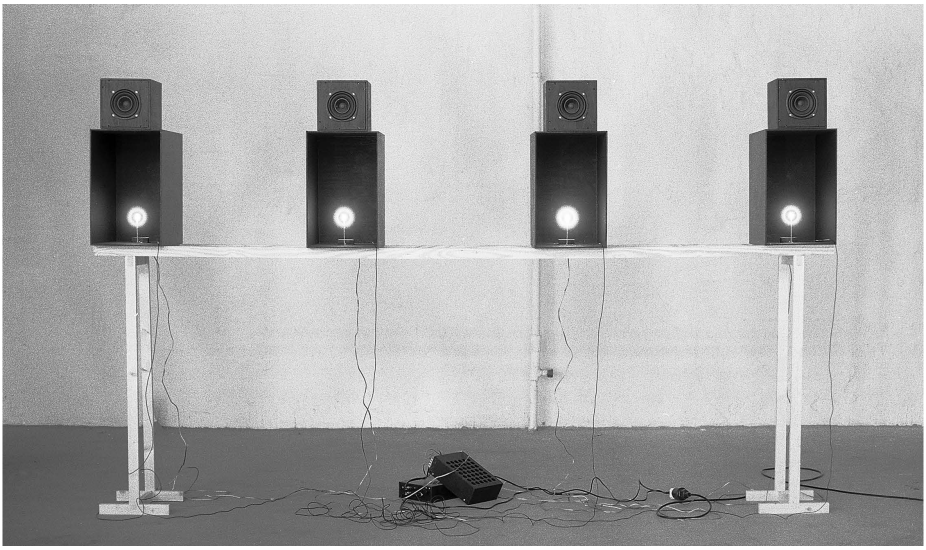


Figure 15.16 Christian Terstegge, *Ohrenbrennen*, 1986.

electronics; they became virtuoso performers with their new instruments, including circuits for extracting sound from blinking lights (see audio and videos on website), radio-controlled cars, radio interference, and obsolete Dictaphones. Christian Terstegge (Germany) has been making elegant sound installations and performances with homemade circuitry since the early 1980s. In his 1986 work, *Ohrenbrennen* ("Ear-burn") (Figure 15.16 and audio on website), four oscillators are controlled by photocells inside small altar-like boxes containing candles; the pitches of the oscillators rise in imperfect unison, punctuated by swoops that trace the sputtering of the candles as they burn down over the course of a dozen minutes.

Toward the end of the 1970s, the first affordable microcomputers came on the market. Cajoled by the visionary Jim Horton (US), a handful of musicians invested in the Kim-1: a single A4-sized circuit board that resembled an autoharp with a calculator glued on top. Programming this thing in machine language, and storing the program as fax-like tones on a finicky cassette tape recorder, was arduous, counterintuitive, and headache inducing, but coding offered one great advantage over building circuits: it was easier to correct mistakes by reprogramming than by resoldering. Over the next 10 years, Apple, Commodore, Atari, Radio Shack, and others introduced increasingly sophisticated machines (and eventually disk drives!) that gradually reduced the angst of programming, and homemade circuits faded into anachronism—until the anti-computer backlash of circuit bending, as proselytized by Reed Ghazala, brought "chipetry" back into fashion.

NOTE

1. For more information on Lunetta's circuits, start on his website (www.moosack.net) and move on to one of these sites, https://strangenessandcharm.wordpress.com/tag/lunetta, www.sdiy.org/rfeng/, http://electro-mngusic.com/forum/topic-30479.html

CHAPTER 16

Soft Circuitry
An Introduction to E-Textile Interfaces

LARA AND SARAH GRANT

INTRODUCTION

Materials have a way of informing and influencing our actions and emotions. Like when you feel the cool smoothness of a solid metal item versus holding a light and airy object made from plastic. You may prefer one over the other for reasons you can't quite put your finger on, but the tangible sensations will no doubt leave an impression. "Electronic textiles" (e-textiles) is the practice of building electronic circuits and components from thread, yarn, fabric, and other soft and flexible materials like plastics and rubbers. This practice explores tangible materials and their visceral properties, as well as their impact on communities, technology, and our environment. E-textiles welcomes and utilizes a variety of skills, making it a great space for skill sharing and collaboration. It also appeals to beginners who wouldn't normally pick up a soldering iron or sewing needle.

There are many exciting materials and techniques for building e-textile interfaces. In this chapter, we will introduce techniques that are beginner friendly, using materials that are easily obtainable and relatively inexpensive. We will build two circuits using traditional components on a breadboard, then we will learn how to make textile-based components and substitute them for standard hardware. We will move the breadboard to the body, then finally move the circuit from the breadboard onto a textile.

This chapter depends on supplies a little different from those in the rest of this book. You can find some online sources listed in Appendix A (Resources).

SOFT ELECTRICAL ART

The Art of Building Soft Electrical Components

Although a switch may be a simple concept, it can yield a lot of expressive power. There are a million and one designs you can create when you know how to build switches from scratch (see Technical Bootcamp on the website for information on the different kinds of switches). For example, the metal slider of a zipper can be used to close a switch created with conductive thread or can function as a linear potentiometer that changes resistance as the zipper is pulled up and down (Figure 16.1).

Figure 16.1 Becky Stern, zipper switch.

It's also possible to build *gesture* sensors. Two examples are a stroke sensor built by Hannah Perner-Wilson, which detects the direction of the stroke, and a soft tilt switch, which uses a metal bead and six conductive fabric pads to sense the direction in which the fabric is tilted. Kobakant, run by Hannah Perner-Wilson with Mika Satomi, presents a wealth of information and examples of DIY e-textiles (Figures 16.2 and 16.3).

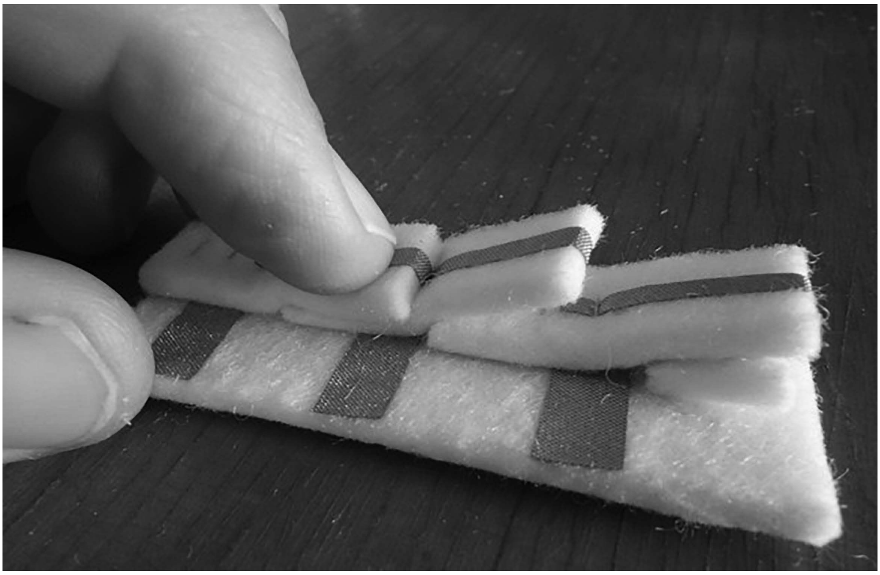


Figure 16.2 Hannah Perner-Wilson, stroke sensor that detects stroking direction.

Figure 16.3 Hannah Perner-Wilson, tilt switch.

Creative freedom also comes with being able to fabricate your own sensors from raw materials. There are many ways to make thin, soft, flexible resistive sensors using yarns, threads, fibers, and even plastics and rubber. Almost any existing component you can think of that is made from resistive and conductive materials can be remade into a soft sensor. Liza Stark created a soft linear potentiometer by combining conductive and resistive fabric together with a long metal bead that rolls along the strips to lower and raise the resistance (Figure 16.4).

Figure 16.4 Liza Stark, linear potentiometer.

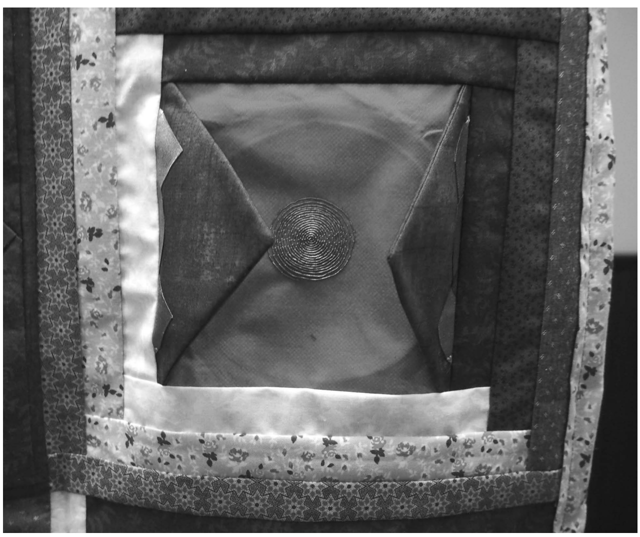


Figure 16.5 Liza Stark, conductive thread speaker as part of *The Tell Tale Quilt*.

Speakers can be made by hand, as well, using a combination of strong magnets and conductive materials. In *The Tell Tale Quilt*, Liza Stark created several speakers with conductive thread. Every square in the quilt contains two soft sensors that detect when someone has folded over a flap of fabric (Figure 16.5). This folding action both reveals a metallic thread speaker and triggers an audio clip of a memory or narrative that plays through the speaker. (See Chapter 9 for advice on constructing paper and fabric speakers.)

Antennas can also be built from electrical fibers, taking advantage of the lack of insulation in e-textiles. Afroditi Psarra's piece *The {Fractal Kimono}* features a fractal antenna made from conductive fabric for picking up electromagnetic signals that are amplified to an audible level (Figure 16.6).

Different kinds of fabrics and fibers present different material properties for interface design, depending on the stretch, drape, or fuzziness of a fabric. Adrian Freed uses a technique called "drape sensing" in his instrument Tablo (Figure 16.7). The drape sensing picks up light touches, as well as forceful or fast gestures, when the conductive fabric, draped on top, touches the piezoelectric strips below it. This makes it possible to detect a wide dynamic range of actions, which is essential for a compelling hand drum controller.

Figure 16.6 Afroditi Psarra, *The {Fractal Kimono}*.

Figure 16.7 Adrian Freed, Tablo.

Other techniques include felting, knitting, crocheting, lace making, and cross-stitching. Afroditi Psarra's LilyKorg, one piece of her Lilytronica series (inspired by the MicroKorg synthesizer and the LilyPad microcontroller), is an embroidered keyboard using conductive thread and a light sensor to generate sounds (Figure 16.8).

Felting is a sculptural technique that produces a textile by entangling wool fibers. Fine-grade steel and bronze wool can be integrated with the wool to make pressure-sensitive sensors. Metallic wool can also be used on its own as traces and electrodes. The Mini Soft Synth is a musical interface that uses conductive fabric, bronze wool, and resistive and conductive thread to create a e-textile interface (Figure 16.9). Two

Figure 16.8 Afroditi Psarra, Lilytronica.

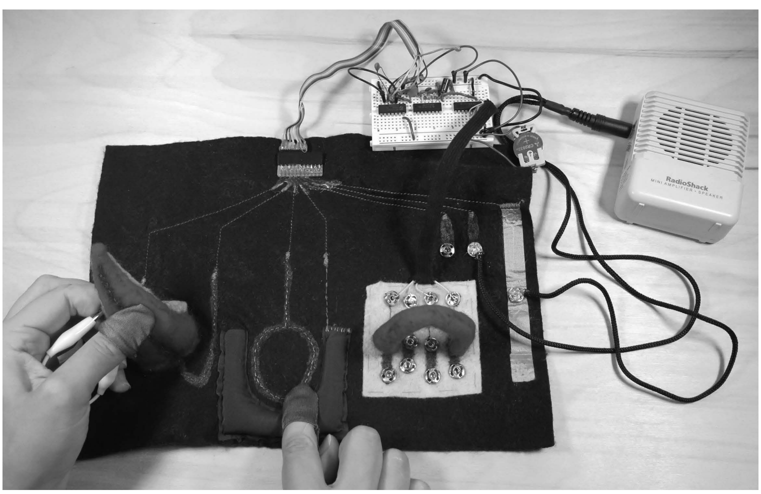


Figure 16.9 Lara Grant, Mini Soft Synth interface.

Figure 16.10
Sarah Grant, *Fortune Teller.*

handmade sensors modulate two oscillators made from a 4093 IC. The sound then goes through a distortion filter built with a 4049 IC and out through a 4040 IC to a patch bay made with snaps and a felted patchcord.

Fortune Teller by Sarah Grant is another experiment with conductive felt (Figure 16.10). This piece is reminiscent of a children's paper fortune teller origami toy, here transformed into a felted controller that sends serial data to a Max/MSP patch through an Arduino board (see Chapter 33). The controller is divided into four quadrants, which move together and apart in sets of halves when manipulated by the player's hands inside. As the controller is opened and closed along two axes, the resistance sweeps over a range of values. This change in resistance across the felt is mapped to a change in parameter values in a Max patch, triggering different note sequences.

Creating the Tools for the Practice

The practice of e-textiles takes all kinds of skills, and it borrows its material and its fabrication methods from the fields of electronics, fashion, and textile design. Not only are breadboards found next to sewing needles on the same workbench, but so are new tools, created to both help fabricate and measure the electrical properties of e-textile components. Irene Posch created her needlework multimeter to read resistance while crocheting a resistor (Figure 16.11).

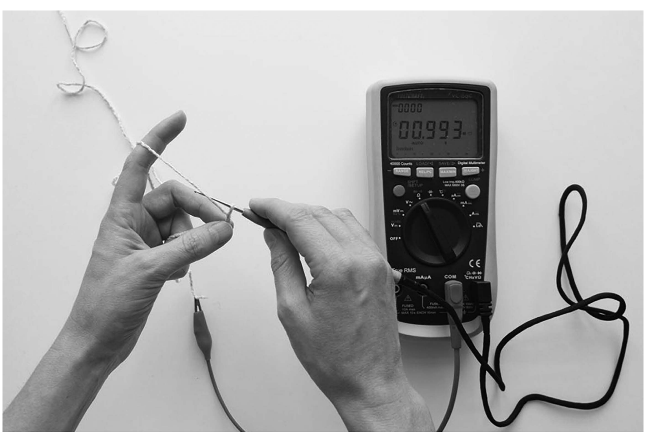


Figure 16.11
Irene Posch, needlework multimeter probes.

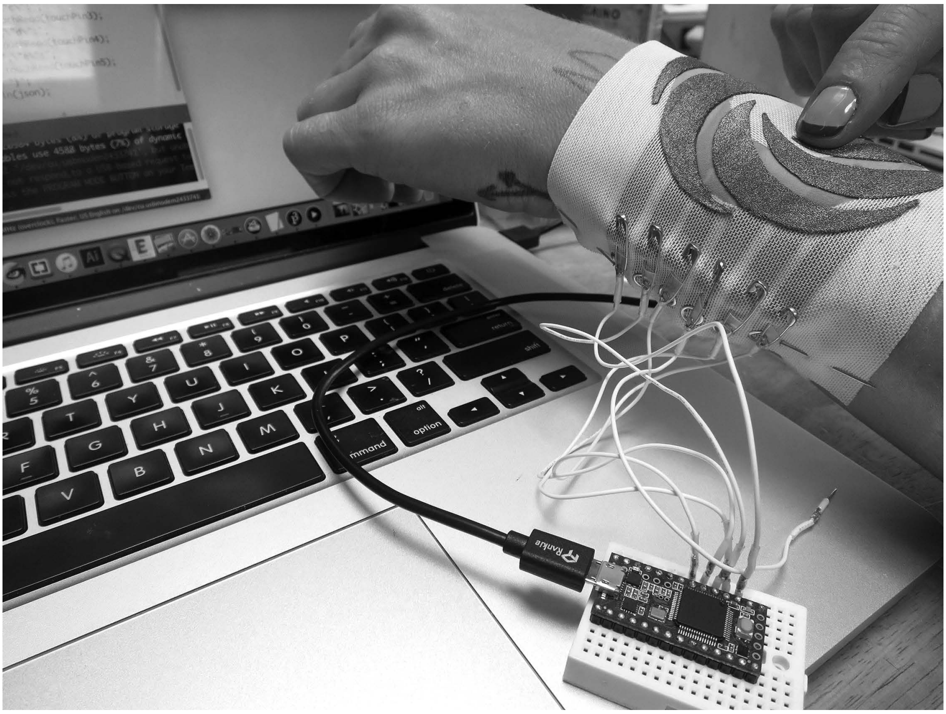


Figure 16.12 Testing a conductive fabric capacitive volume wheel for the Embodisuit by Rachel Freire and Sophia Brueckner. A safety pin to male header connector is used between a conductive fabric trace and a breadboard.

Connection techniques are major concerns in the world of e-textiles. Whether prototyping or designing for manufacturing, practitioners are always creating new ways of connecting hard and soft components, depending on their needs. Rachel Freire developed a safety-pin-to-breadboard connector to test a capacitive volume wheel made of conductive fabric for the Embodisuit, a wearable platform made by her and Sophia Brueckner that allows the wearer to map signals to modular haptics actuators on the body (Figure 16.12).

The Joining of Craft and Electronics

The joining of electronics and traditional crafts, such as hand sewing, weaving, felting, and embroidery, has demystified electronics and made it accessible to people who might not normally approach the field. Being able to sew a circuit has introduced kids and novices to a new path toward circuit design and programing. The LilyPad Arduino was a milestone in the history of e-textiles, created by Leah Buechley and released in 2007 by SparkFun (Figure 16.13). While getting her PhD at the University of Colorado Boulder, she created a sewable electronics kit to get kids into STEM (Figure 16.14). It was made of individual components, such as a photocell and LED, mounted onto fabric modules with conductive fabric pads that could be connected by conductive thread. It also included a programmable Atmel AVR chip and was the early prototype of the LilyPad Arduino board sold today.

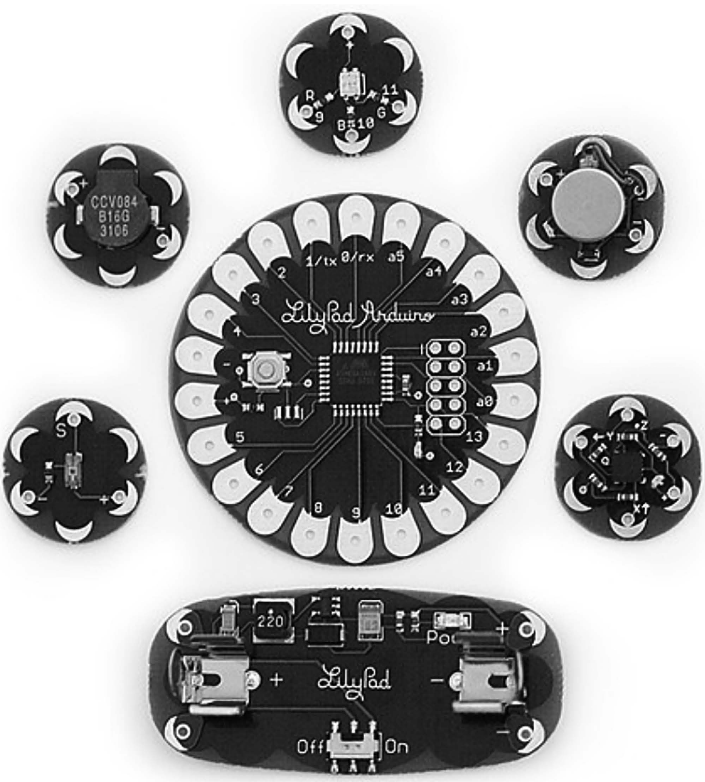


Figure 16.13 Leah Buechley, fabric LilyPad microcontroller.

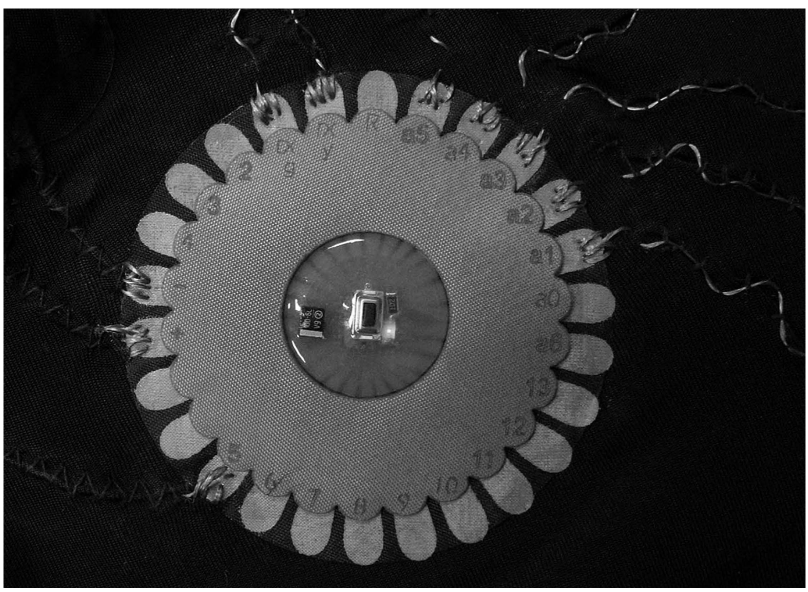


Figure 16.14 Leah Buechley and SparkFun, LilyPad Arduino e-textile kit.

The fusion of craft and electronics has led to the breakdown of some stereotypes. Workshops that teach circuit construction through sewing have created a safe space for people of different age groups and gender to gather, learn, and skill share. Creating these accessible and alternative paths can reveal skills and personal stories that may have previously been overlooked or undervalued (Figure 16.15).

Musical ensembles built through social interaction are possible using instruments such as Sam Topley's pompompot (Figure 16.16). Pompompot is a 555 timer instrument with an interface made from two pom-poms. By touching the pom-poms, people put themselves in series with a circuit generating a square wave. Squeezing the pom-poms varies the resistance, which changes the pitch of the oscillator. The pom-poms offer a playful and inviting way to connect to others and the circuit.

Figure 16.15 E-textile-based prototypes are part of a research project using e-textile objects for touch-based interaction for visual impairment by Emilie Giles and Janet van der Linden.

Figure 16.16 A bridge between two conductive pom-poms of a pompompot instrument created by Sam Topley.

MODULATED FREQUENCY CIRCUIT

Materials

- One CMOS Hex Schmitt Trigger (74C14, 40106, 4584).
- A 47 uf polarized capacitor (C1).
- A 0.1 uf ceramic capacitor (C2).
- Three 100 kOhm potentiometers (R1, R2, R3).
- A 3-volt 2032 type coin-cell battery and battery holder.
- A breadboard.
- Solid-core hookup wire.

BUILDING THE CIRCUIT

Before we start swapping hard components for soft ones, it's important to have a working circuit prototyped using the traditional components. Testing and adding soft components one by one will minimize frustration and potential for failure (Figure 16.17).

All of our circuits will run off of a 3-volt 2032 type coin-cell battery, often used in low-voltage soft circuits. Insert the hex inverter into the breadboard with the notch facing to the left. Connect the positive lead from the battery clip to pin 14 and the negative lead to pin 7. You will need ground on both sides of the chip, so jump the ground from one part of the board to the other (Figure 16.18) (you can also prototype this circuit on a larger breadboard that includes power buses on both sides). As a best practice, ground any unused input pins on our IC. Connect the 47 uf capacitor, labeled C1 in our schematic, pushing the long leg into the same row as pin 1 and the shorter leg into the ground bus. Connect C2, the 0.1 uf capacitor, between pin 5 and ground. Connect the potentiometers (pots)—later

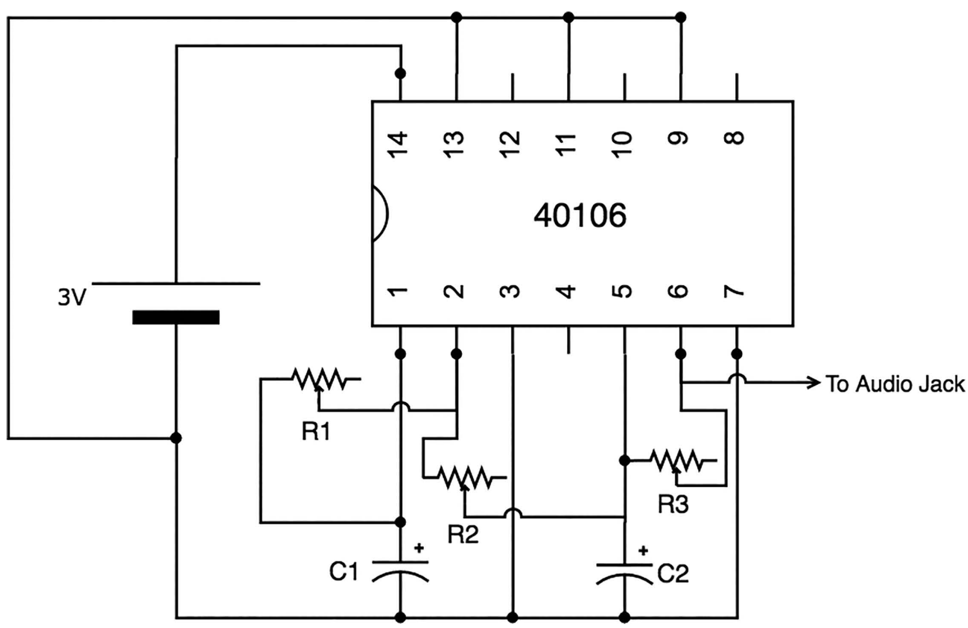


Figure 16.17 Modulated frequency schematic.

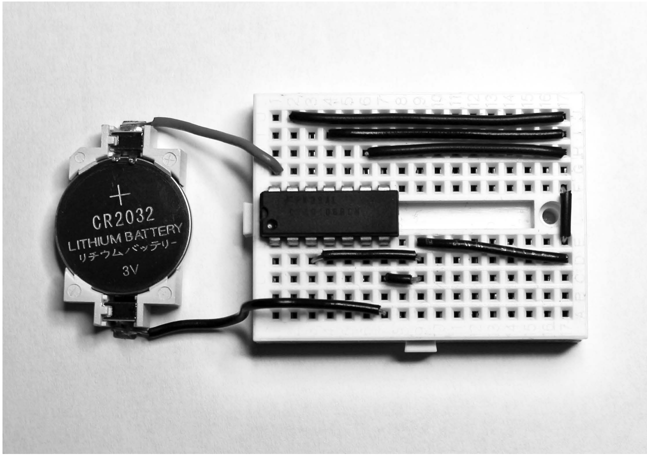


Figure 16.18 Breadboard with 74C14 IC, wires, and battery clip.

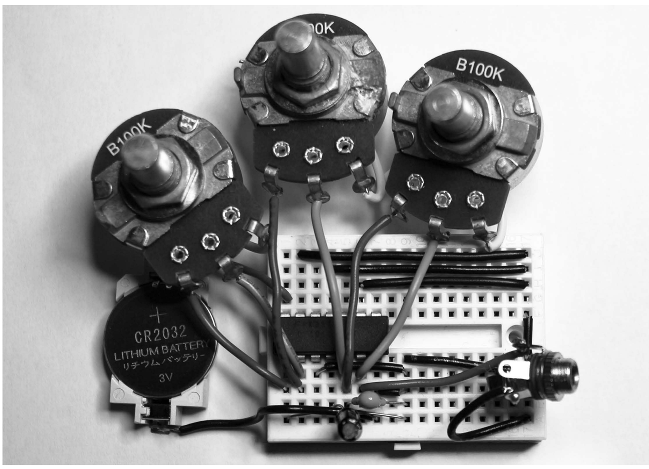


Figure 16.19 Completed circuit with audio out socket.

these will be replaced with soft components. Solder short wires to one outer tab (ear) and the center one (nose) of a 100 kOhm pot, labeled as R1 in our schematic, and connect them via the breadboard to pins 1 and 2 (it doesn't matter which tab goes to which pin). Do the same thing with the second 100 kOhm pot (R3), linking it to pins 5 and 6. Finally, connect pins 2 and 5 to the tabs of the third 100 kOhm pot (R2) (Figure 16.19).

Time to hear some oscillations! Connect the tip of a 1/8-inch (3.5 mm) mono audio socket to pin 6, the sleeve to ground, and run a cable to your amp and speaker. You should hear a square wave tone, pitched high or low depending on how your pots are set. If not, check your wiring. Once you have confirmed that the circuit is working, it's time to make some soft components.

POM-POM PRESSURE SENSOR

Our pom-pom pressure sensor uses conductive yarn. The yarn is a mixture of polyester (80%) and AISI 316L stainless steel (20%). The stainless steel will not corrode, and the yarn can be washed. When the pom-pom is squeezed, the stainless steel threads compress together. The compressed threads give electricity a more direct path, lowering the resistance in relation to the force applied.

Materials

- One bobbin of conductive yarn.[1]
- Wool yarn, worsted or chunky weight.
- Scissors.
- A scrap of thick cardboard.

Cut a rectangular cardboard with a slot cut from it (Figure 16.20). The sturdier the cardboard, the better. Depending on the size of your cardboard pom-pom maker and the thickness of your yarn, the number of wraps needed may be more or less. You may want to make an all-wool test pom-pom before adding the conductive yarn. For chunky weight yarn, do about 15 wraps of wool, then 15 wraps of conductive yarn, 15 of wool, 15 of conductive, and 15 more of wool. Take a short piece of wool yarn, slip it through the slot in the cardboard, and wrap it around the middle of the bundle (Figure 16.21). Tie one half of a square knot to hold the yarn in place and slide it off of the cardboard. Tighten the yarn around the middle and finish tying the knot. Cut all of the loops in half and fluff it into a pom-pom (Figure 16.22). If you like, you can trim the yarn to round the pom-pom out. Clip one alligator lead to a group of conductive yarn strands and a second lead to a group of conductive strands on the opposite side of the pom-pom (Figure 16.23). Remove the R2 pot and clip the pom-pom to the jumper wires. Squeeze to change the sound.

Figure 16.20 Wool yarn, cardboard with slot, and bobbin of conductive yarn.

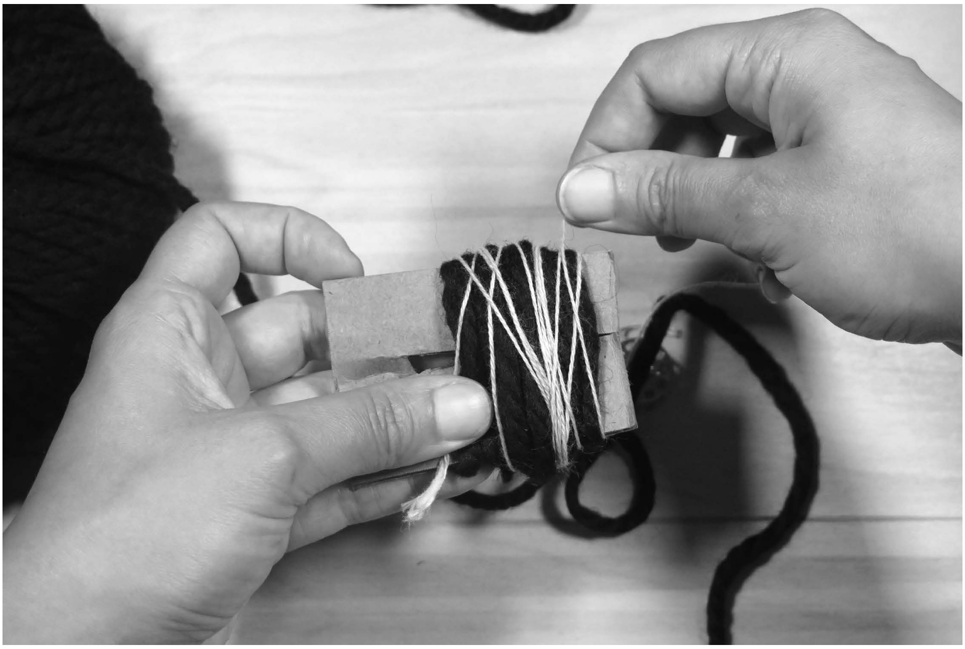


Figure 16.21 Wrapping conductive yarn over the wool wraps.

Figure 16.22 Cutting the loops.

Figure 16.23 Alligator leads clipped to conductive yarn in the pom-pom.

A STRETCH SENSOR

Conductive yarn can be knitted into a sensor without additional tools using a technique called "finger knitting" (Figure 16.24). A knitted sensor can act as a stretch sensor, or you can squeeze or twist it. Finger knitting produces a large and rather loopy knit stitch. To make the sensor thicker, knit the conductive yarn with a worsted, or heavier, weight yarn. This will help it retain its shape after deformation. To add even more elasticity, try knitting conductive yarn, regular yarn, and elastic thread together.

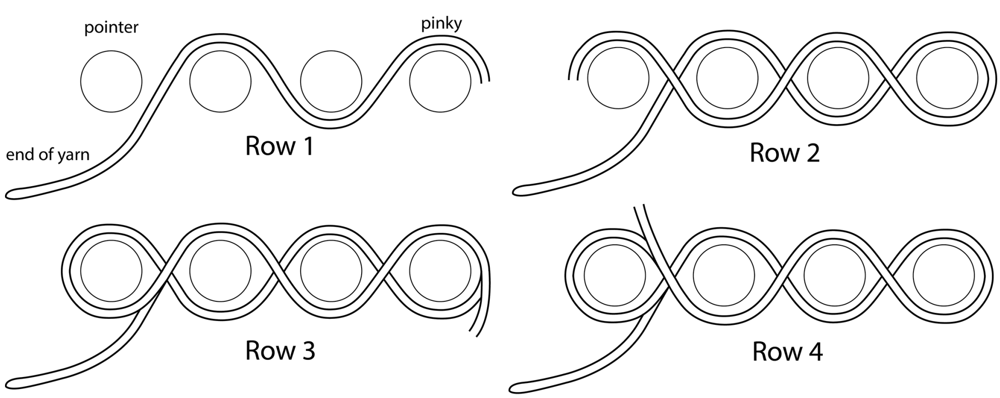


Figure 16.24 Guide for finger knitting.

Materials

- One bobbin of conductive yarn (as used in previous pom-pom sensor).
- Wool yarn, worsted or chunky weight.
- Scissors.

Start by draping the conductive and regular yarn over your pointer finger, leaving a 6-inch tail (Figure 16.25). Weave the yarn under and over the fingers to create four rows. This is called an e-wrap cast on.

Each finger should now have two strands of yarn. On the pinky, pull the bottom strand over the top strand and the tip of the finger. Do this to the next two fingers. When you get to the pointer finger, take the end of the yarn and place it in between the pointer and middle finger.

After moving the yarn end, the cast on is complete and you are ready to start knitting. Wrap the yarn around the pointer and weave the yarn through your fingers again to create two rows. You will end up with two strands on each finger. To knit a row,

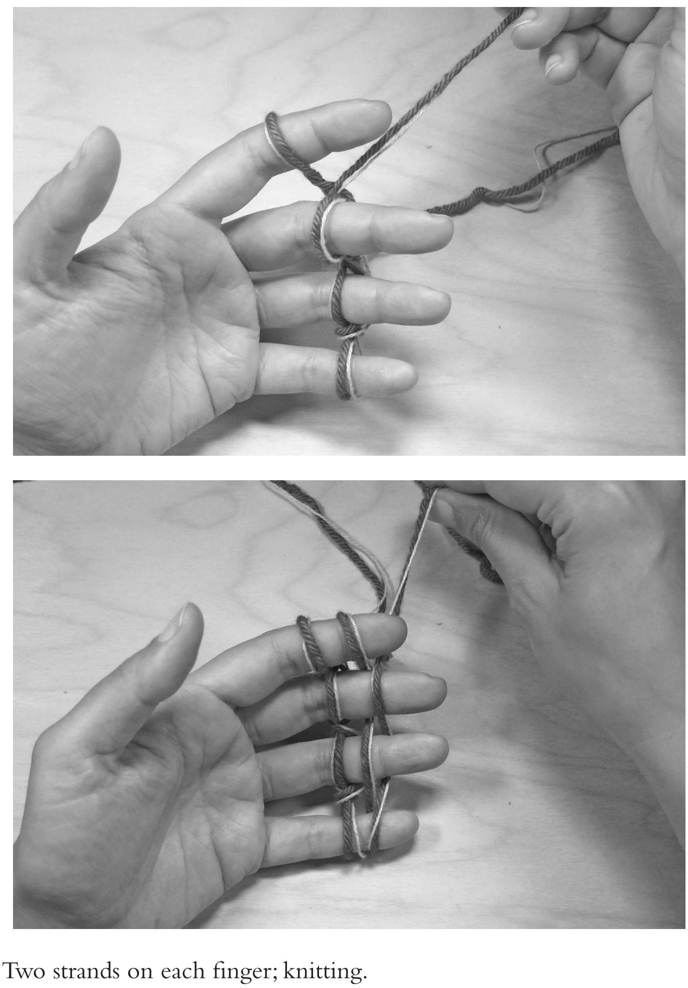


Figure 16.25 Two strands on each finger; knitting.

pull the bottom strand over the top strand and the tip of your finger again. Do this for each finger to complete the row. Continue to weave two rows, and weave stitches in this manner until you get the length of sensor you want.

Finishing is called binding off. To bind off, cut the yarn, leaving a 6-inch tail. Thread the end of the tail under the loop on the pinky, then the ring finger, index finger, and last the middle finger. Take the yarn off of the fingers and pull the tail, gathering and securing the end (Figure 16.27).

Grab two alligator leads and clip one to each end of the sensor, making sure the alligator teeth clip onto the conductive yarn (Figure 16.28). You now have two soft sensors to play with. Replace R1 with the stretch sensor and stretch to lower the resistance and change the sound.

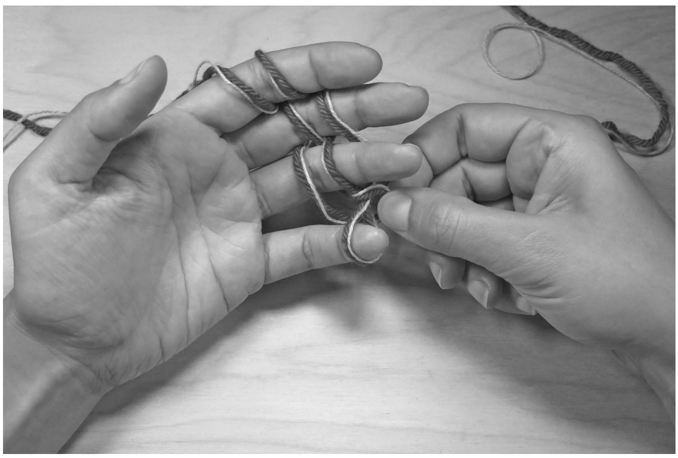


Figure 16.26 Pulling the bottom strand over the pinky; placing the end of the yarn between the pointer and middle finger.

Figure 16.27 Pulling the tail.

Figure 16.27 (Continued)

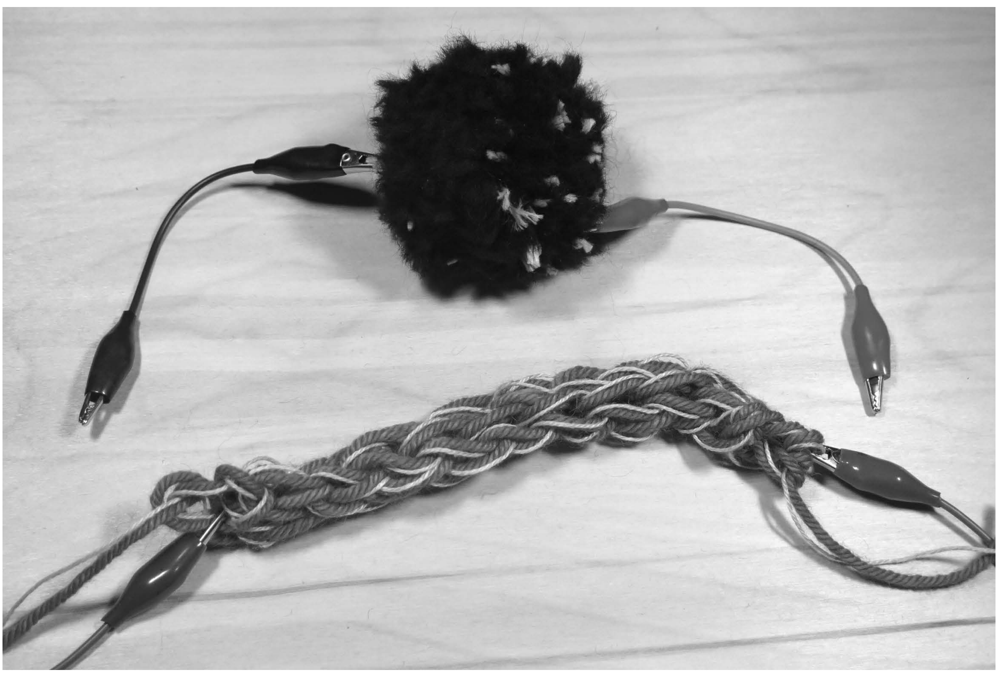


Figure 16.28 Completed pom-pom squeeze sensor and knitted stretch sensor.

RESISTOR LADDER CIRCUIT

Materials

- Modulated Frequency circuit.
- One 100 kOhm potentiometer (R1).
- One 100 kOhm resistor (R2).
- One 47 kOhm resistor (R3).
- Three 10 kOhm resistors (R4, R5, R6).

A resistor ladder is simply a chain of resistors connected together in line to achieve effects like frequency division to generate stepped pitches. To make the resistor ladder, you'll take your Modulated Frequency circuit, swap a component, and add some resistors and switches (Figure 16.29).

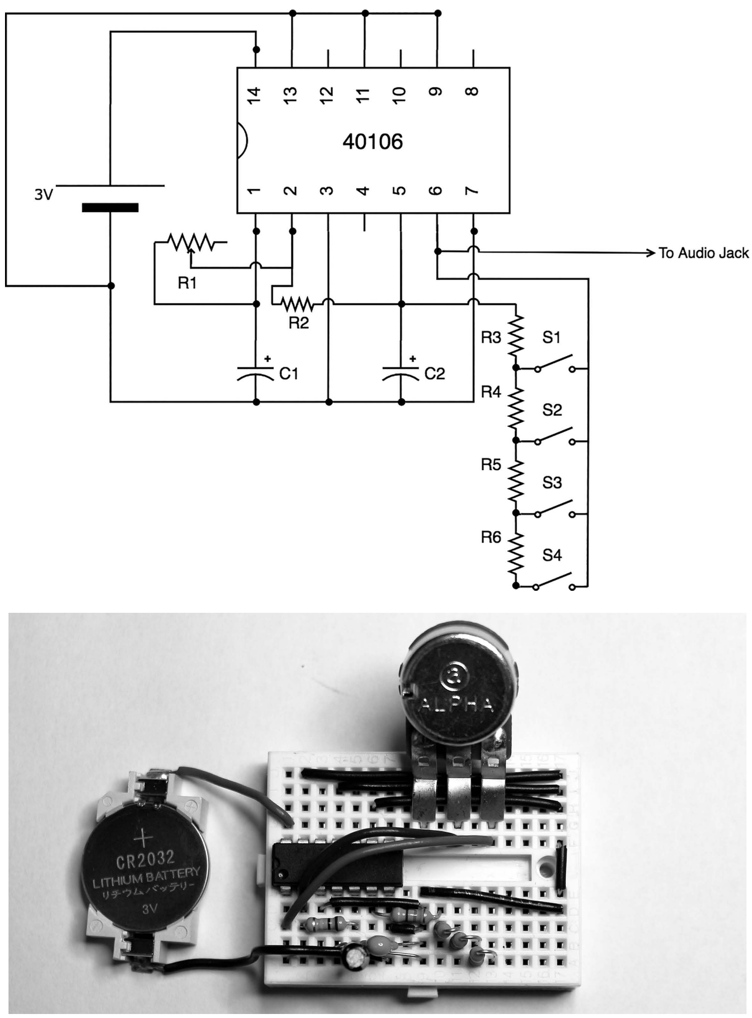


Figure 16.29 Resistor ladder with resistors R3, R4, R5, and R6 (schematic and breadboard view).

First, move R1 over to the other side of the board and link some wires over from pins 1 and 2 of the chip to the center and one outer tab of the pot. We are going to make this breadboard wearable, so swapping R1 for a panel-mount pot that can plug directly into the breadboard isn't necessary but makes things tidy and compact. Next, replace the 100 kOhm pot we used for R2 with a 100 kOhm resistor.

Start building the resistor ladder by pushing one leg of the 47 kOhm resistor in line with pin 5 of the IC in our breadboard and the other leg in another (open) part of the breadboard. In order to fit the rest of our resistors on the mini breadboard, we'll need to bend the legs over so that the resistor takes on a U shape and can fit into two adjacent holes. Then take our three 10 kOhm resistors and connect them in a line, one after another on the breadboard, with the first leg of each resistor in the same row as the last leg of the previous resistor.

In order to turn notes on and off, we will need several switches placed in between each stage of the resistor ladder and pin 6 of the IC. These switches could be regular momentary pushbuttons, but we will be making them from conductive fabric attached to our fingers.

SOFT SWITCHES

An advantage of a soft switch is that it can be comfortably placed anywhere on the body. The hand is convenient and expressive. Five switch contacts will be plugged into the resistor ladder and put on the tips of each finger and one on the thumb. A note can then be played by touching the thumb contact to any of the fingertips.

Materials

- Sheet of 100% sheep's wool felt.
- HeatnBond Ultrahold.
- Iron-on conductive fabric.
- Straight pins.
- 22-gauge stranded wire, preferably silicone coated.
- Sewing needle.
- Thread.
- Iron.
- Scissors.

There are many kinds of conductive fabric created from different kinds of fibers and metals. There are ones that stretch, ones that are antimicrobial, and ones that change resistance when pressed. Appendix A lists some good sources for these materials. In this chapter we are using a highly conductive copper-plated polyester with a tarnish-resistant finish. To secure it to our felt, we could sew it down, but that can be tricky and time consuming. A quick and stable solution is to make the fabric "iron-on." To prepare for use, iron a hot-melt adhesive specifically for making appliques, like HeatnBond Ultrahold, to one side of the fabric before any cutting is done.

Cut five strips of wool felt long enough to wrap around each fingertip and the thumb on one hand (Figure 16.30). Cut five pieces of conductive fabric slightly thinner and the same length (Figure 16.31).

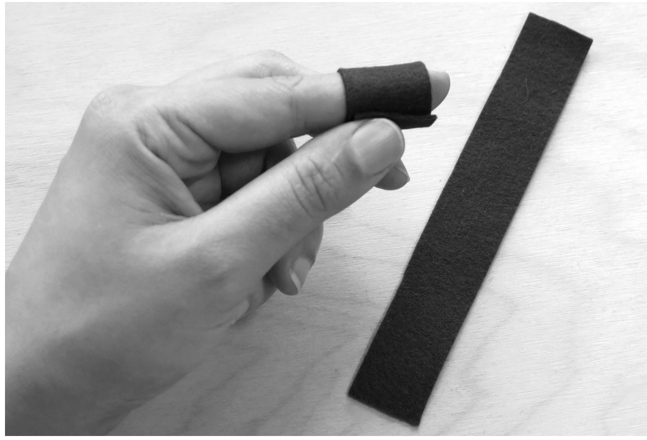


Figure 16.30
Wrapped felt around fingertip.

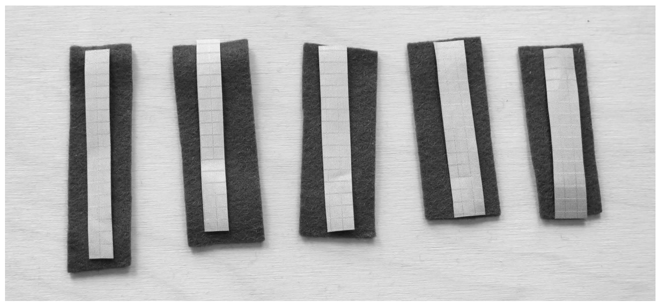


Figure 16.31
Conductive fabric and felt strips.

Figure 16.32
Ironing down the conductive fabric onto the felt.

Iron the conductive fabric onto the felt with the adhesive side down using an iron set to medium heat (Figure 16.32). Before you wrap them around your finger, take a moment to test the functionality of the soft switch. Grab two of the contacts and plug them in series with R1 where S1 is in Figure 16.33. Clip one side to pin 1 and the other to R1. Touch the contacts together to hear sound. You have a basic soft switch: two low-profile, soft, and flexible contacts that can reside in places that traditional switches can't.

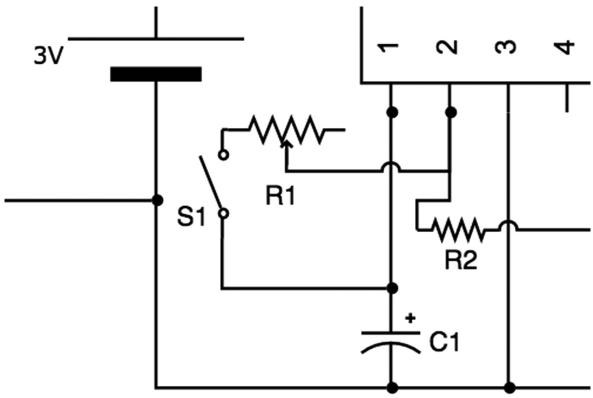


Figure 16.33
Soft switch.

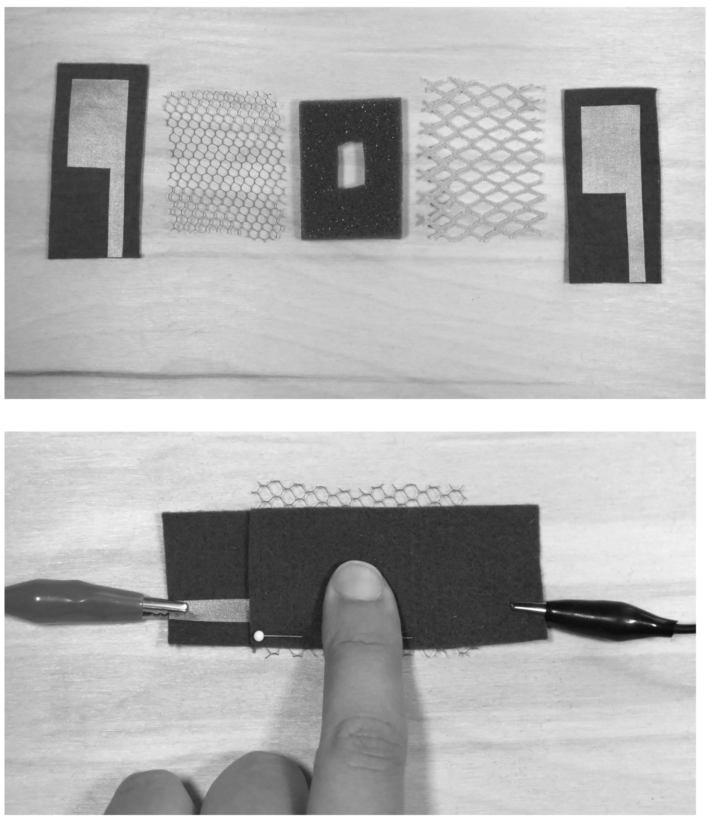


Figure 16.34 Top: switch contact with lead, netting, foam, thicker netting, another contact with lead. Bottom: pressing a soft switch with netting in between.

To create a kind of momentary pushbutton, put some netting fabric (or a piece of foam with a hole(s) cut from it) in between the two contacts and pin it together (Figure 16.34). The pins are conductive, so do not pin through the contacts, which will create a short. The netting creates a space between them. When pressed, the contacts touch through the gaps in the net, closing the switch.

After you are done experimenting, move all five contacts to the hand. Fit the strips onto each fingertip and pin the ends into place (Figure 16.35).

Pinning is a temporary and potentially prickly solution. The next step is to secure each ring with a common sewing stitch called the running stitch.

Push a threaded needle through the fabric and then back through at about the same spot, making a tiny stitch (Figure 16.36). Go back and forth to make at least two

Figure 16.35 Pinned switches.

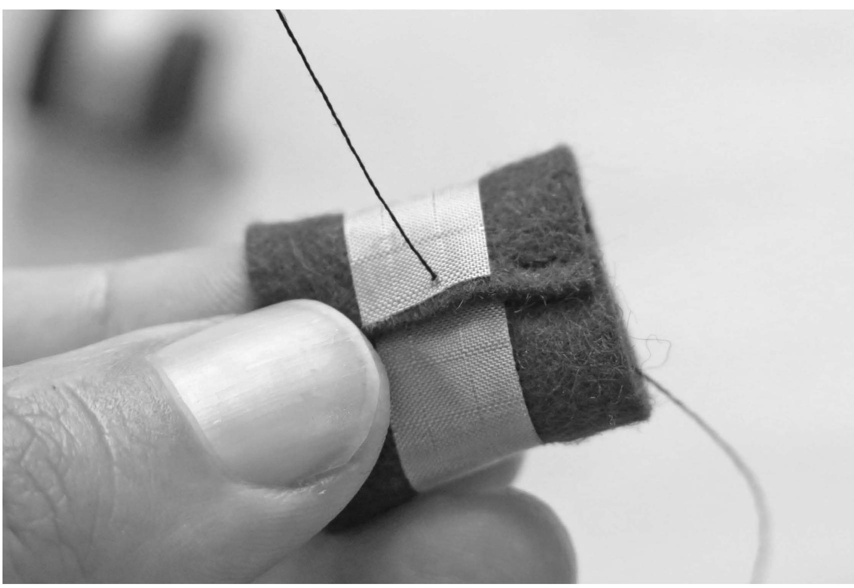


Figure 16.36 Sewing a felt ring closed.

tiny stitches to prevent the thread from being pulled out. This eliminates the need for a knot, which will keep things less bulky. This becomes especially helpful when sewing with conductive thread. Make a few stitches until you get to the end and secure the thread as you did in the beginning. Cut off the hanging tails. Do this for all five felt rings.

FABRIC TO BREADBOARD CONNECTIONS

Alligator leads are bulky and can easily get disconnected. Straight pins soldered to stranded wire make good alternative connectors between a breadboard and a soft switch while experimenting (Figure 16.37). Use silicone-coated stranded wire when possible: it is heat resistant and very flexible. To connect your five contacts to the breadboarded circuit, cut five pieces of wire about 8 inches long and solder a straight pin to one end of each wire. Stranded wire is more flexible than solid wire, but it can be difficult to plug into a breadboard and can slip out. You can tin the stripped ends to stiffen them or solder them onto a male header. Once done, pin one end to a finger contact and plug the other into the breadboard (Figure 16.38).

For the four fingers, plug one between R3 and R4, R4 and R5, R5 and R6, and the fourth at the end to R6. The fifth thumb contact gets connected to pin 6 and acts as the other side for all four switches, S2, S3, S4, and S5 in Figure 16.29.

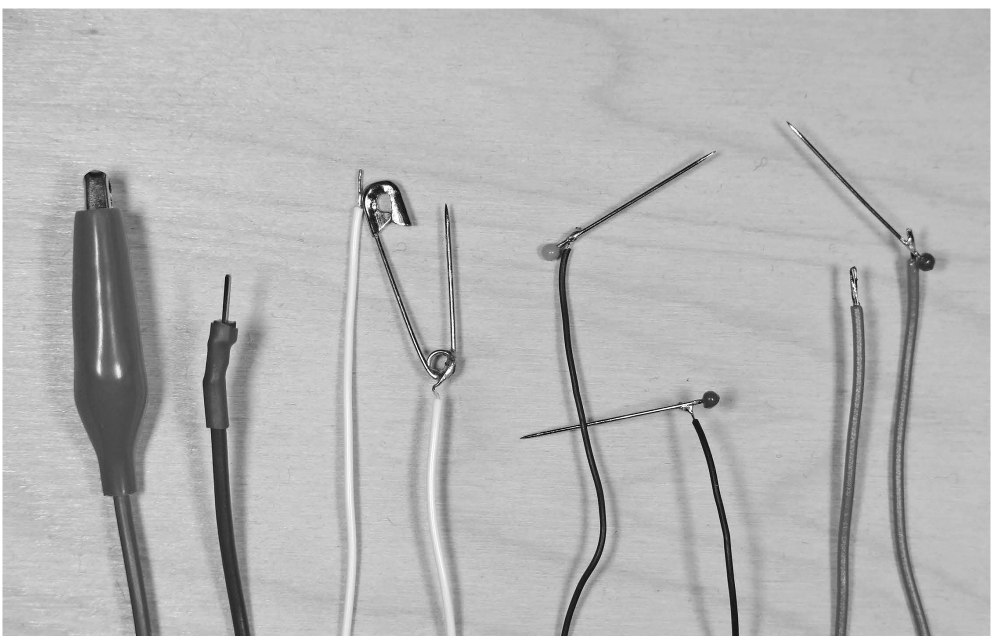


Figure 16.37 Connectors.

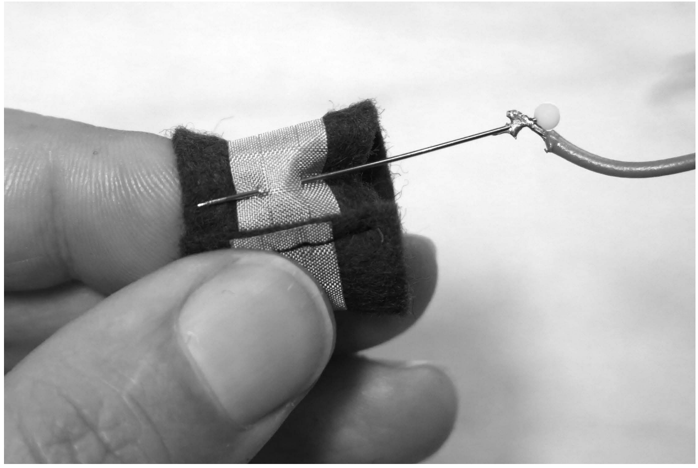


Figure 16.38
Pin connected to a contact.

MAKING THE BREADBOARD WEARABLE

Materials

- Double-sided Velcro.
- Hot glue gun and glue sticks.

Hook and loop (Velcro) is handy for quickly attaching sensors and circuits to the body. With double-sided hook and loop you can mount the breadboarded circuit onto your wrist. Size some hook and loop to your wrist with the hooks facing out. Hot glue a piece of loop to the bottom of your breadboard and stick it to the wrist strap. You now have a wearable prototyping platform.

Use hot glue or hook and loop, as needed, to secure the jack, battery, and pot to the strap. Wrap the breadboard-bracelet around your wrist and slip the rings onto your fingertips. Touch each finger with your thumb to play different pitches. If you hear no sounds when S5 or S4 are closed, turn R1 to a higher resistance. Playing single notes can be fun, but things become a lot more interesting once sensors are added to the mix. Let's learn how to put a handmade force sensing resistor onto the finger to modulate frequencies with the bend of a finger.

A WEARABLE FORCE SENSOR

Materials

- Velostat.
- Sheet of 100% sheep's wool felt.
- Jersey knit (T-shirt material).
- Fabric pen.
- Iron-on conductive fabric.
- Sewing needle.

- Thread.
- Iron.
- Scissors.
- Multimeter.

We are going to make a force sensing resistor (FSR) using a plastic called Velostat. Velostat is a carbon impregnated polymer that is electrically conductive. When force is applied, its resistance to electrical current goes down. There are fabrics that do this as well; however, as of writing this chapter, they are unavailable to the general public. Velostat can be cut into any shape to create custom resistive sensors. The resistance of a piece of Velostat is dependent on its width and length.

When a handmade FSR is placed on an elbow or a finger, the extent of the bend changes the resistance, making it a useful bend sensor. To get a stable reading, the sensor needs to be held in place against the bend it is sensing. This can be done by sewing it to a glove, shirt, or handmade accessory. We are going to sew a small finger sleeve for our sensor.

To make a finger-sized bend sensor and a sleeve to hold it to your finger, cut the following:

- One strip of Velostat 3/8 inch × 2 1/2 inches.
- Two L-shaped pieces of conductive fabric 1/4 inch wide × 2 1/4 inches long.
- Two strips of felt 3/4 inch wide × 3 inches long.
- Small square of jersey knit.

Iron down the conductive fabric contacts and cover one with the Velostat. Sew it down with a couple of stitches at both ends to hold in place (Figure 16.39).

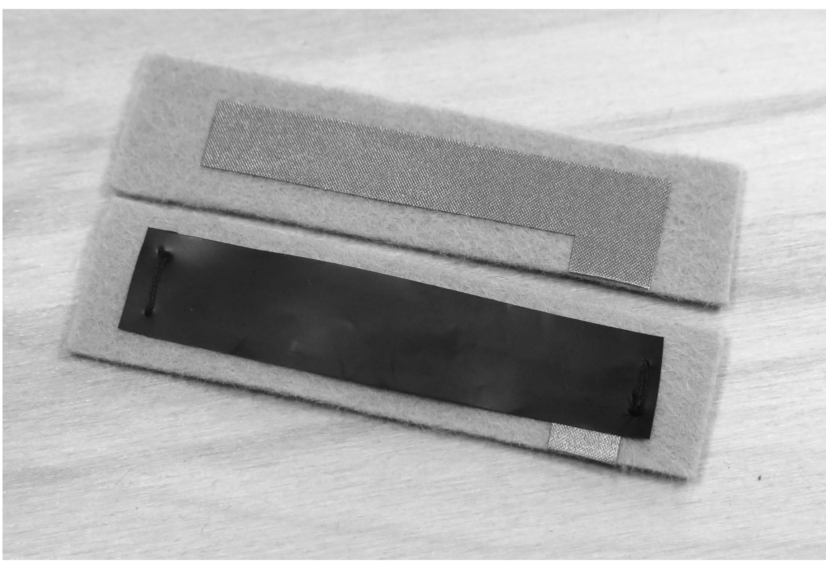
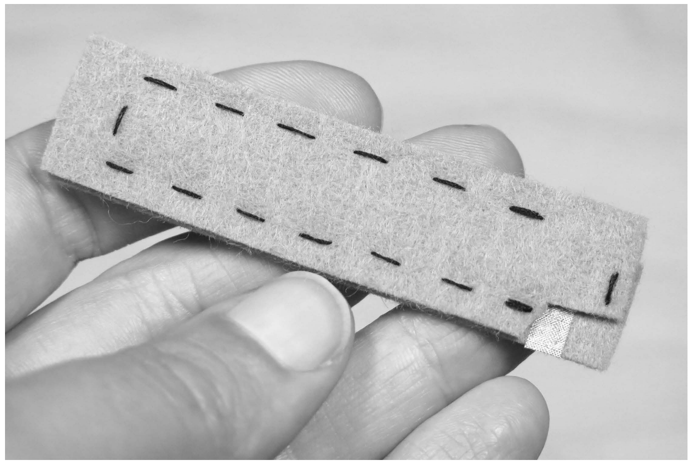


Figure 16.39 Velostat stitched over one contact.

Figure 16.40
Finished sensor.



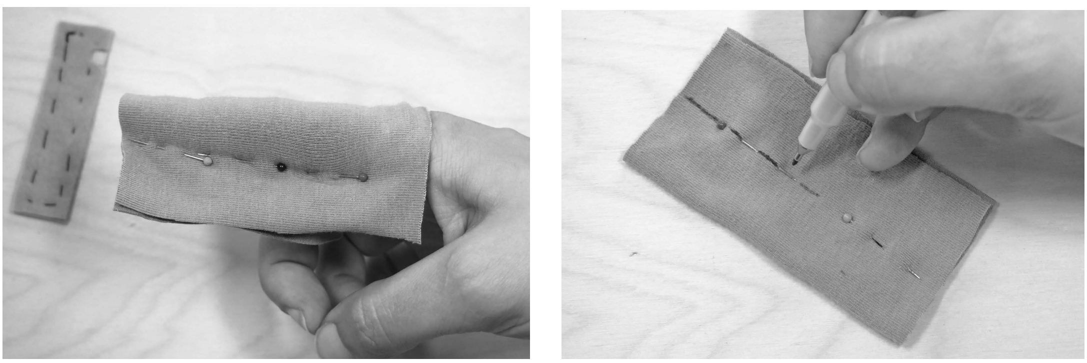


Figure 16.41 Fitting fabric to finger with pins; marking the fabric.

Pin the sensor together and connect it to a multimeter. Press it and watch the resistance go down in relation to the force applied. It's important that, when stacked, the two contacts do not touch each other to create a short. This is why the Velostat piece needs to be *larger* than the contact pads, and the leads are on opposite sides from each other when closed. If your multimeter displays a 0 or low number, the conductive fabric is shorting. Plug the sensor into the resistor ladder circuit where R1 is and listen to the timbre change when pressed or bent. To finish, sew around the edges (Figure 16.40). Cut a couple of openings to expose the leads if you like.

To sew the sleeve, pin the knit fabric around your finger to fit snug over the knuckle. Be careful not to make it too tight or the bend sensor will be squeezed against your finger, reducing the range of resistance you have to play with. Take the sleeve off and mark where the pins are with a fabric pen (Figure 16.41).

Remove the pins, center the sensor between the marked lines, and sew it down (Figure 16.42).

Sew the two sides together along the markings to close the sleeve. Trim the excess fabric (Figure 16.43).

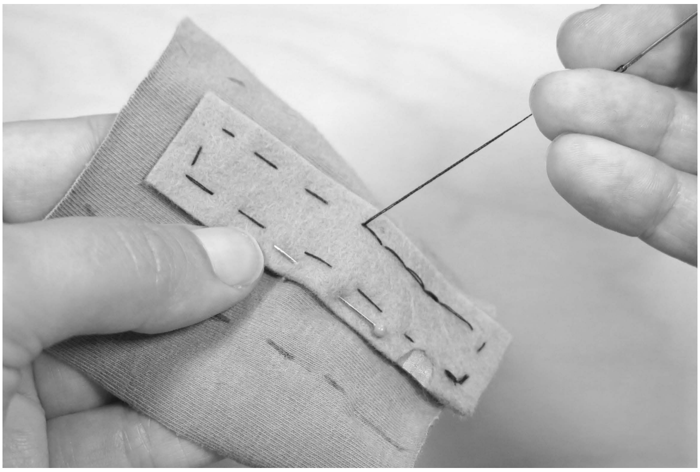


Figure 16.42
Sewing the sensor to the sleeve.

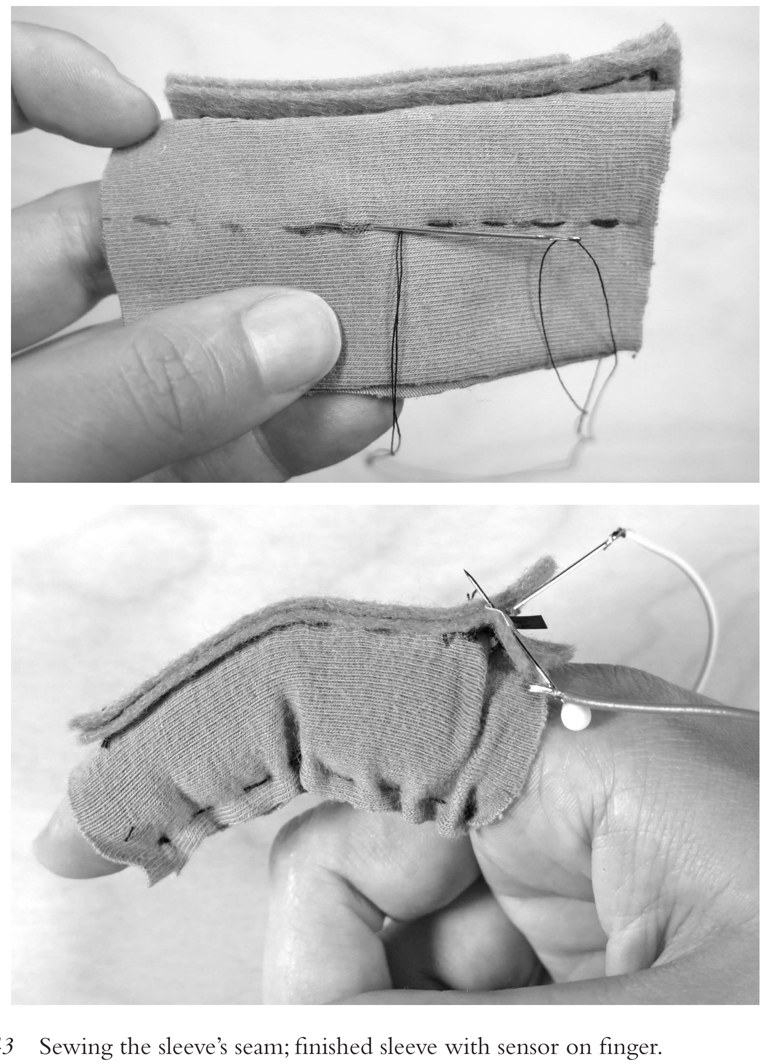


Figure 16.43 Sewing the sleeve's seam; finished sleeve with sensor on finger.

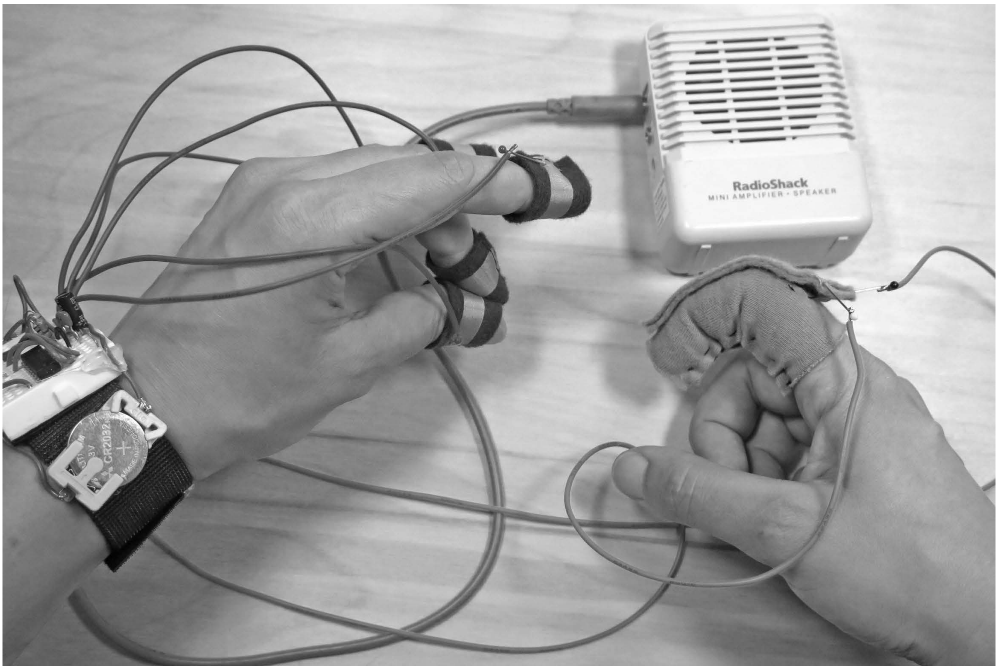


Figure 16.44 Playing with the sensor.

Solder up two more wire-to-pin connectors, pin one to each tab, and slide the sleeve over the finger. Plug it back into the circuit and bend your finger to modulate the second oscillator. Touch each fingertip with your thumb to change the pitch via the ladder (Figure 16.44). If you get tired of hearing the sensor all the time, try putting a soft switch in series to switch the sensor in and out, using any bare fingertip.

As soon as you start moving, wires may try to wiggle their way out of the breadboard. To reduce the amount of stress and keep them in the breadboard, add some strain relief: hot glue all of the hanging wires to the side of the breadboard, leaving a little slack at the breadboard end.

TAKE THE CIRCUIT OFF THE BREADBOARD

Components can be transferred and secured to fabric in a few different ways. Here is a method that emphasizes modularity, with a soft breakout board. You will solder the components to a protoboard and break out from points in the circuit to conductive fabric pads that in turn get connected to soft components. For connections, you can either pin, alligator clip, snap, sew, or solder all your soft components together. Yes, this fabric is solderable! Making it modular means you can keep experimenting rather than finalizing a single circuit design.

The protoboard should be double-sided and mimic the layout of a breadboard.[2] If you can only find a single-sided board you will need to assemble it upside-down,

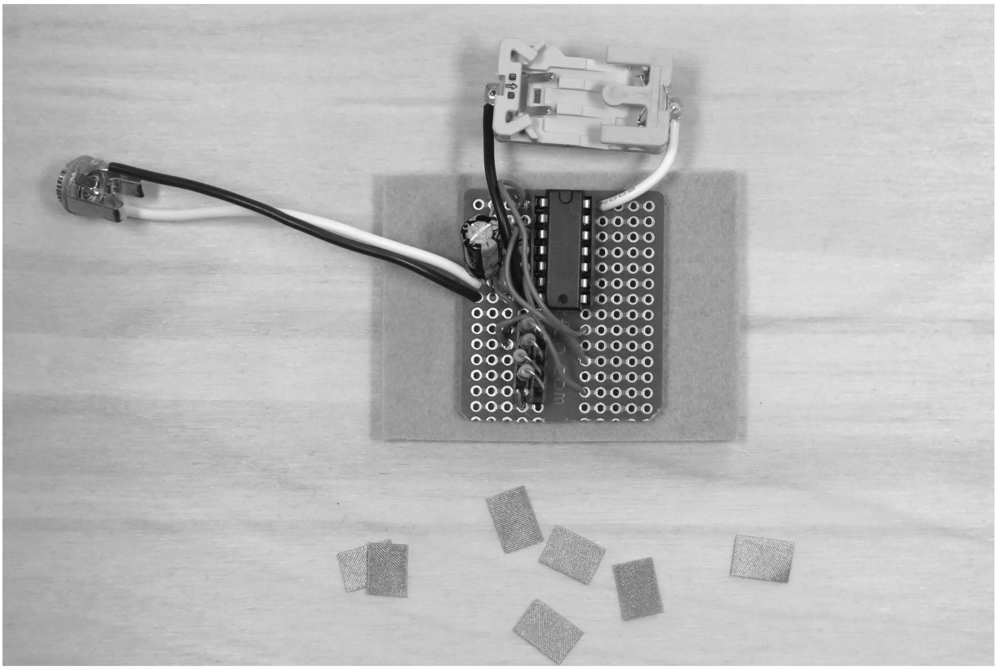


Figure 16.45 Cut felt and conductive pads with finished protoboard circuit.

with the copper traces up, and solder components from the top. It's doable but trickier. You will sew it down using conductive thread, and it's important that the thread makes solid contact with the exposed metal portion of the trace.

There are five contacts for the resistor ladder switches and two contacts for the bend sensor. This makes seven connections needing to be broken out from the circuit.

To make more room to sew, things have been slightly shifted from the original breadboard layout. Follow the diagram to solder up the circuit to the protoboard to ensure everything will fit nicely the first time. Use low-profile female headers in place of the resistors if you want to be able to swap them out. Start by soldering the IC socket, then all the other components, headers (if using them), and wires. Cut a piece of felt 2.25 inches by 1.75 inches and seven small pieces of conductive fabric for the pads (Figure 16.45). Iron down four pads on the side of the protoboard where the audio jack is and three on the opposite side as shown in Figure 16.46. Use regular thread to stitch through an unused row on the protoboard to secure it to the felt.

The conductive thread you are using is stainless steel, so it will not corrode, and it is washable. The resistance is fairly low (about 14 Ohms per foot). The resistance is proportional to length, which can add up and is important to take into account if you use it for traces in a circuit design. Conductive thread can be tricky to start working with. It's basically an exposed wire that can fray and shift, making it liable to create short circuits, as well as unreliable connections, if you don't work neatly and plan ahead.

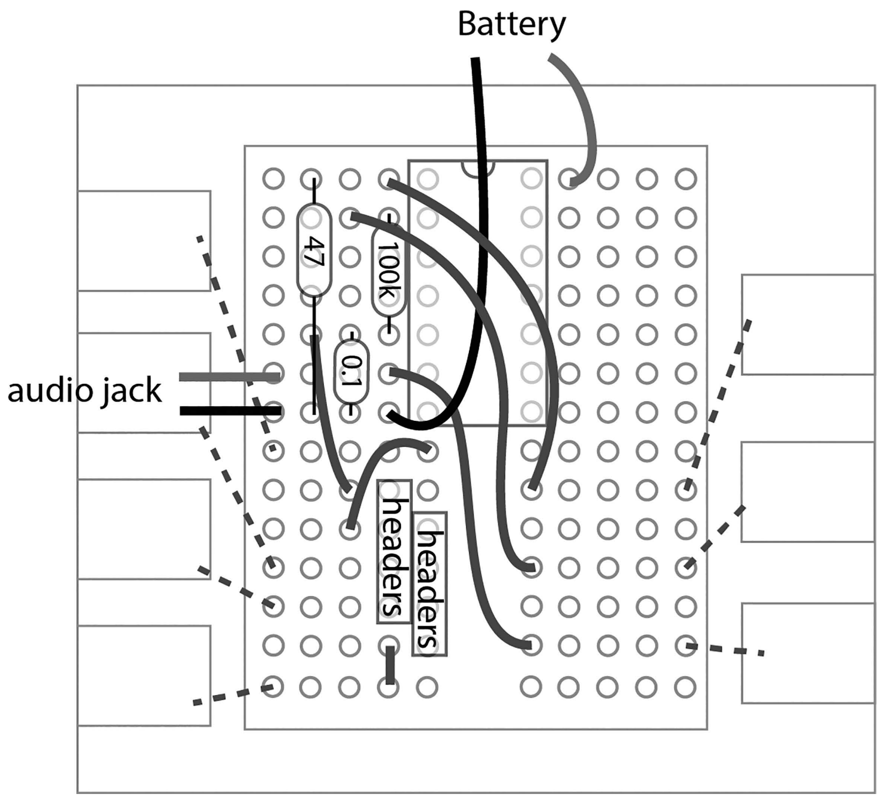


Figure 16.46 Diagram with felt and conductive fabric pads.

Here are some tips and rules to follow when working with it:

- Plan and visualize traces and connections before you start sewing to ensure you have enough room that they do not touch. If a trace needs to cross another, make an insulated bridge.
- Make small, tight stitches.
- Use the stitch-in-one-place method instead of tying a knot. If you do make a knot, secure it with hot glue or clear nail polish.
- Always cut off any remaining tails.
- Stitch at least three times when making a connection.
- Check each connection and trace with a multimeter after you make it.

To give a visual target, mark the rows you will be sewing to. Then draw the connections from each pad to its corresponding row, making sure each row-to-pad connection is as far from the adjacent ones as possible. Using conductive thread, stitch to one pad (remember the rule of making at least three stitches!). Next, slide the needle under the surface of the felt to bury and protect the thread and then bring the needle up next to the marked row (Figure 16.47). Stitch through the hole on the protoboard,

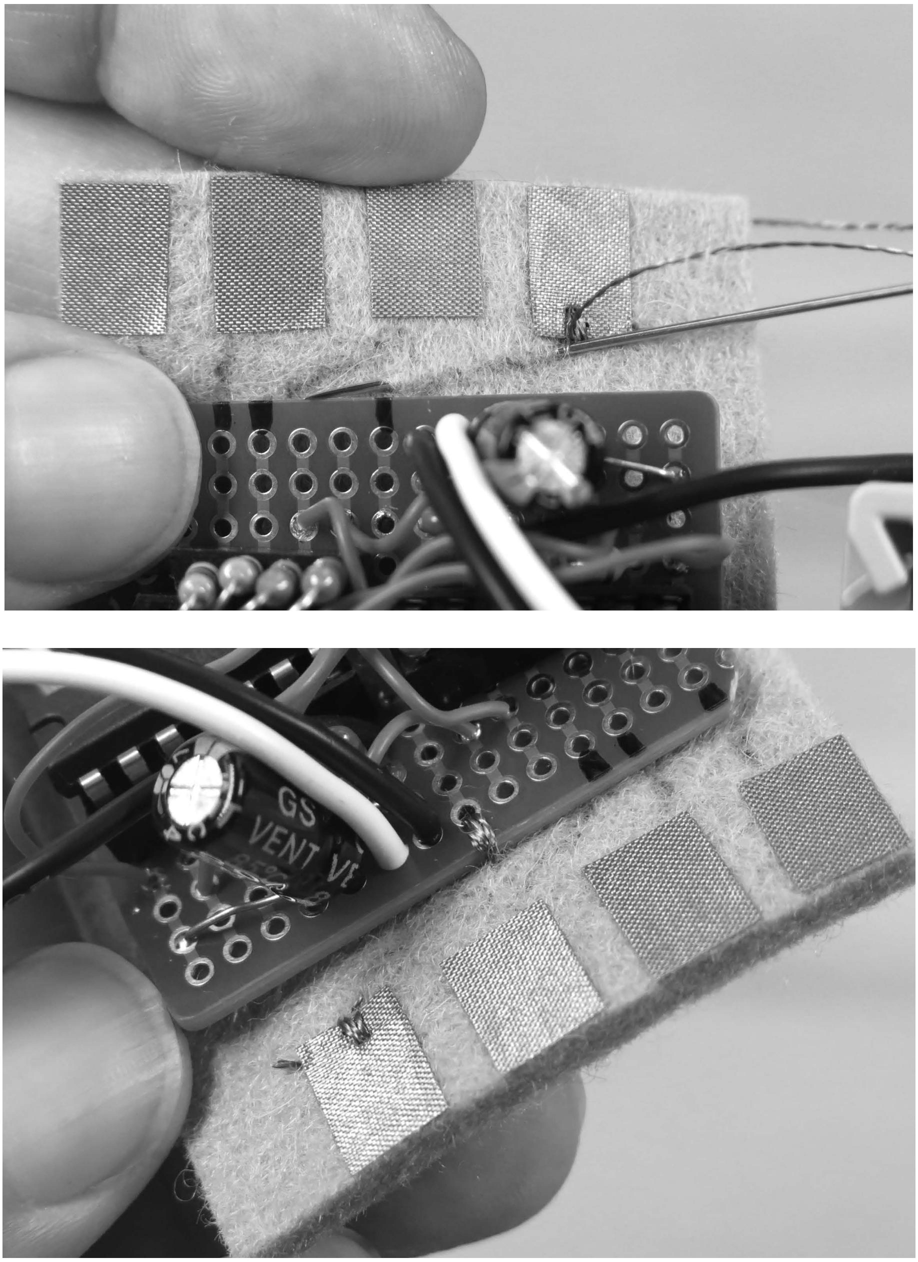


Figure 16.47 Conductive thread stitches on conductive fabric pad; conductive thread stitches going through protoboard hole.

ideally making a stitch in the above adjacent hole as well to ensure a good electrical connection.

Once finished, cut your two tails and check the connection with the multimeter before moving on. Do this for the remaining six pads (Figure 16.48).

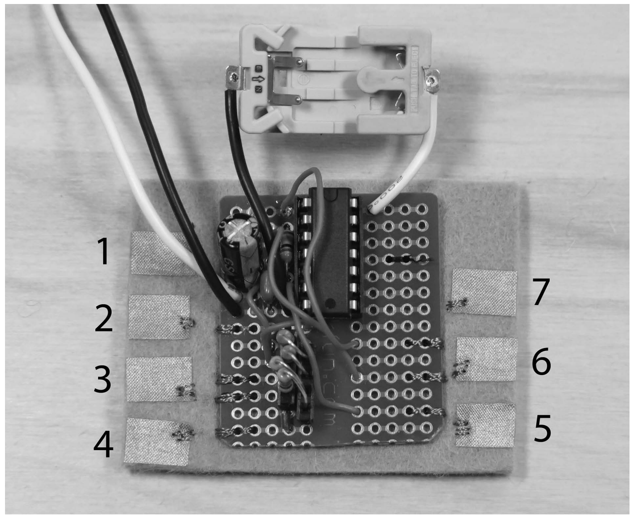


Figure 16.48
Finished soft breakout board.

To test, clip one of the homemade sensors to the pads that are connected to pins 1 and 2 of the IC and an alligator lead to any of the four pads that are connected to the resistor ladder. Power the circuit and plug it into the amp (Figure 16.49). Press the sensor and touch the lead connected to the resistor ladder to the pad that is connected to pin 6 of the IC. You should hear a pitch change and a note played once you touch the lead to the pad. If not, check your connections. Use the fabric pads as electrodes; instead of connecting the sensor, press one finger across the two pads that are connected to pins 1 and 2 to put yourself in the circuit. Since they are already exposed, soft circuits are excellent for laying your hands on, cracklebox-style (see Chapter 12).

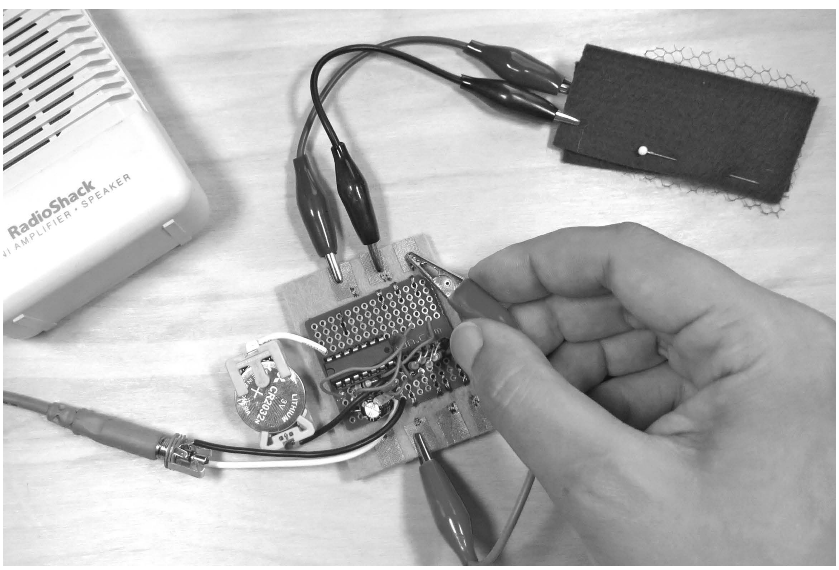


Figure 16.49 Testing resistor ladder switches.

FIBER-TO-FIBER CONNECTIONS

Now you're ready to use all the components you have made. Start connecting and playing around. Put soft contacts between two hands or in between pleated fabric. Try different configurations using alligator leads and pin connections. To keep it wearable, cut a hand-sized square piece of felt with a wrist strap on which to mount the soft board (Figure 16.50). Or lay the components onto another piece of felt and create a soft interactive surface that folds, squishes, and stretches. Once you are ready to commit to a circuit layout, you can sew, solder, or snap everything together using conductive thread, fabric, or wire.

Snaps

Snaps are a very popular connector in e-textiles. Sew them to conductive fabric using conductive thread. They make great switches, too.

Stitches

Use conductive fabric or a running stitch of conductive thread to create a trace, then stitch that to conductive fabric.

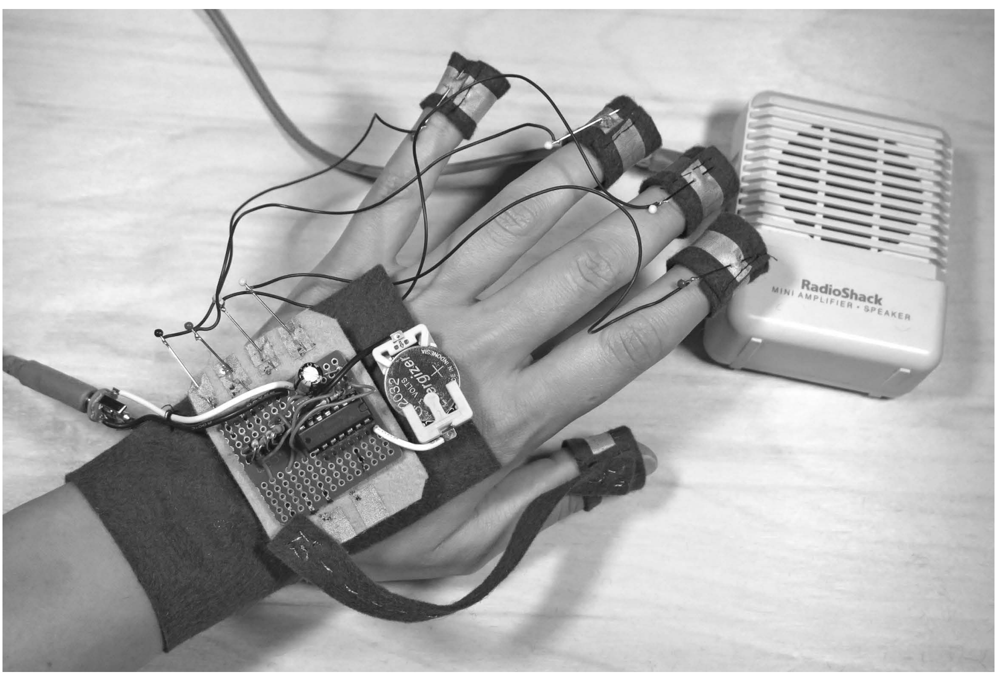


Figure 16.50 Soft breakout board made wearable with pinned and snap connections.

Solder

Soldering is possible on some conductive fabric, like the one we use in this chapter. However, the connections are delicate, and you will need strain relief. Using hot glue over a connection will help stabilize the joint so it doesn't break off.

Pins

Solder a pin to the other end of your fabric to breadboard wires so both ends can pin into conductive fabric to make a connection.

NOTES

1. lessemf.com
2. www.gikfun.com/electronic-pcb-board-c-60/mini-solderable-breadboard-gold-plated-finish-proto-board-pcb-p-766.html

CHAPTER 17

On/Off (More Fun With Photo Resistors) Gating, Tremolo, Panning, and More

You will need:

- Two sound sources to process, such as smartphone, computer, radio, electric guitar, etc.
- Two amplifiers.
- Some photoresistors.
- A flashlight or flashlight/strobelight app for your cellphone.
- A Hex Inverter (74C14/40406/4584) or Quad NAND Gate (4093).
- A breadboard.
- Some LEDs (light emitting diodes).
- Some heat-shrink tubing (optional).
- Assorted resistors and capacitors.
- Some solid hookup wire.
- Some plugs and jacks.
- Clip leads and Y-cords.
- A 9-volt battery and connector.
- Hand tools.
- Plastic electrical tape.

As we have seen in our oscillator experiments, the photoresistor changes resistance in response to changes in light level. We've harnessed this change in resistance to control the pitch, volume, and filtering of an oscillator. The photoresistor can also be used as a gate or volume control to pass, block, fade, or pan any audio source (such as a music file or an electric guitar) and can substitute for some switches in toys, keyboards, and other circuits. In this chapter we'll take a look at some of these applications.

FLASHLIGHTS

Assemble the simple circuit shown in Figure 17.1. Using clip leads, connect one leg of a photoresistor to the "hot" or tip of any audio signal, such as the output of your phone (probably not a vintage portable CD player as shown in the photo), and connect the

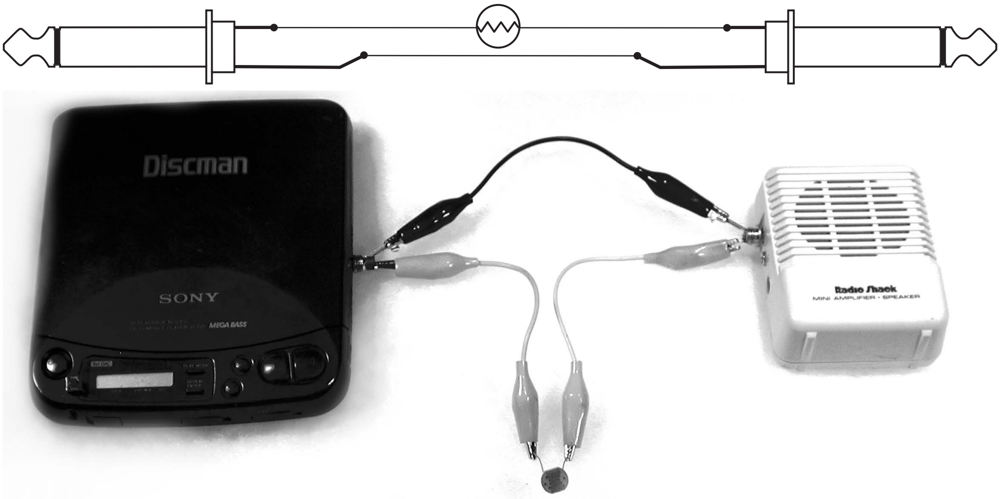


Figure 17.1 A basic photoresistor gate circuit.

other leg to the hot/tip of your amplifier input. Connect the shield/ground of your audio source to the shield/ground of the amplifier input (remember Rule #10: every audio connection needs signal and ground). Turn on the amp, play your sounds, and confirm that audio passes through.

Now take the whole rat's nest into a dark place, like a closet, or turn off your lights and draw the blinds. The sound should get quieter. Switch on your flashlight and pass the beam across the photoresistor—the sound should get louder when the cell is lit, quieter when the cell is dark. Run the strobe light app on your cell phone and you've got an old-school tremolo. This super simple circuit won't shut off the sound completely, but you should hear a significant volume drop between light and dark. The back side of a photoresistor is usually translucent, so total darkness and quieter level can only be achieved if you fully darken the cell: enclose it in your hand, for example. Covering the back with black electrical tape eliminates this problem. Make sure you don't let the legs touch against one another or the signal will pass through unattenuated regardless of light level.

You can increase the dynamic range of this circuit (the difference in loudness between on and off) by adding a resistor of about 10 kOhm between the *output* side of the photoresistor and ground, as shown in Figure 17.2. The resistor "clamps" the output to ground when the circuit is off, minimizing bleed-through of the diminished input signal and thereby increasing the depth of the muting when the circuit is in its off state. Until we add the clamping resistor, this circuit works in both directions: either jack can be used as an input or output. Once the clamp is connected between a jack's hot and shield, we set that jack as the output and the other as input.

This circuit is an absurdly simple but nonetheless very effective light-controlled audio gate. You can use it with any audio signal, but it is more effective with nominally line-level signals, such as a phone or laptop, rather than the very high signals generated by your oscillators—not because of any flaw in our circuit but because the "off" leak

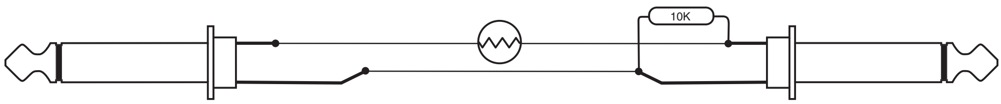


Figure 17.2 Clamped photoresistor gate.

through of such a hot signal is high enough that it can swamp the high gain input stage on an amplifier. This is a "passive" signal processor, which means that it needs no batteries (although your flashlight does). The only requirement is darkness, which makes it well suited for stage use, camping trips, or those in arrears on their electrical bill.

PING PONG, PONG PING

By adding a second photoresistor and some more connectors and audio devices, we can expand our gate into a light-controlled panner or mixer. Hook up the configuration shown in Figure 17.3.

Connect an audio signal through two clip leads to two photoresistors. Clip the free leg of one photoresistor to the hot/tip of a plug connected to one amplifier and connect the free leg of the second photoresistor to the hot/tip of a plug connected to a second amplifier. Link together the grounds on all three connectors with two more clip leads. Seek the cover of darkness once again, play your source material, turn on the amplifiers, and sweep the flashlight beam back and forth across the two photoresistors: the sound should pan between the two speakers, following the movement of the light across the two sensors.

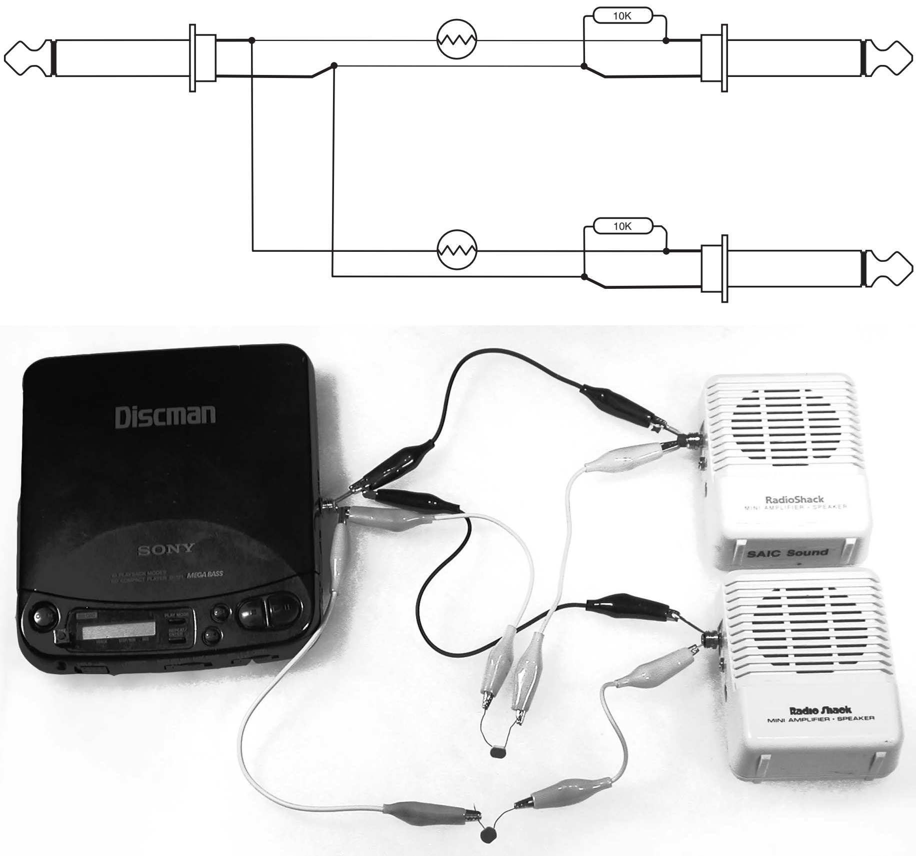


Figure 17.3 Light-controlled panner.

Now rewire the circuit slightly, as shown in Figure 17.4. This time connect one audio source through a clip lead to one leg of one photoresistor. Connect a second audio source through a second clip lead to one leg of a second photoresistor. Combine the free legs of both photoresistors through clip leads to the input of one amplifier and link all the grounds together. Turn off the lights. Now when you pass a flashlight across the two photoresistors you should be able to cross-fade and mix between the two sources, like cutting between turntables.

You can solder up this a version of this circuit with two input jacks and two output jacks, as shown in Figure 17.5, so it can be used as a panner, a mixer, or a two-channel gate. To make a panner, use an audio Y-cord to connect a single audio source to both input jacks and patch the outputs to two amplifiers. For a mixer, connect a different source to each input and use a Y-cord to mix the outputs of both photoresistors to a single amplifier input. For two independent channels of gating, hook two audio sources to two amplifiers with no interconnections (like the two CD players and amps shown in the figure).

This basic panner/mixer circuit can be expanded with more photoresistors, inputs, and/or outputs to make four-channel panners, multi-channel mixers, etc. Note that male plugs are shown in all these drawings since they make it easy to distinguish the

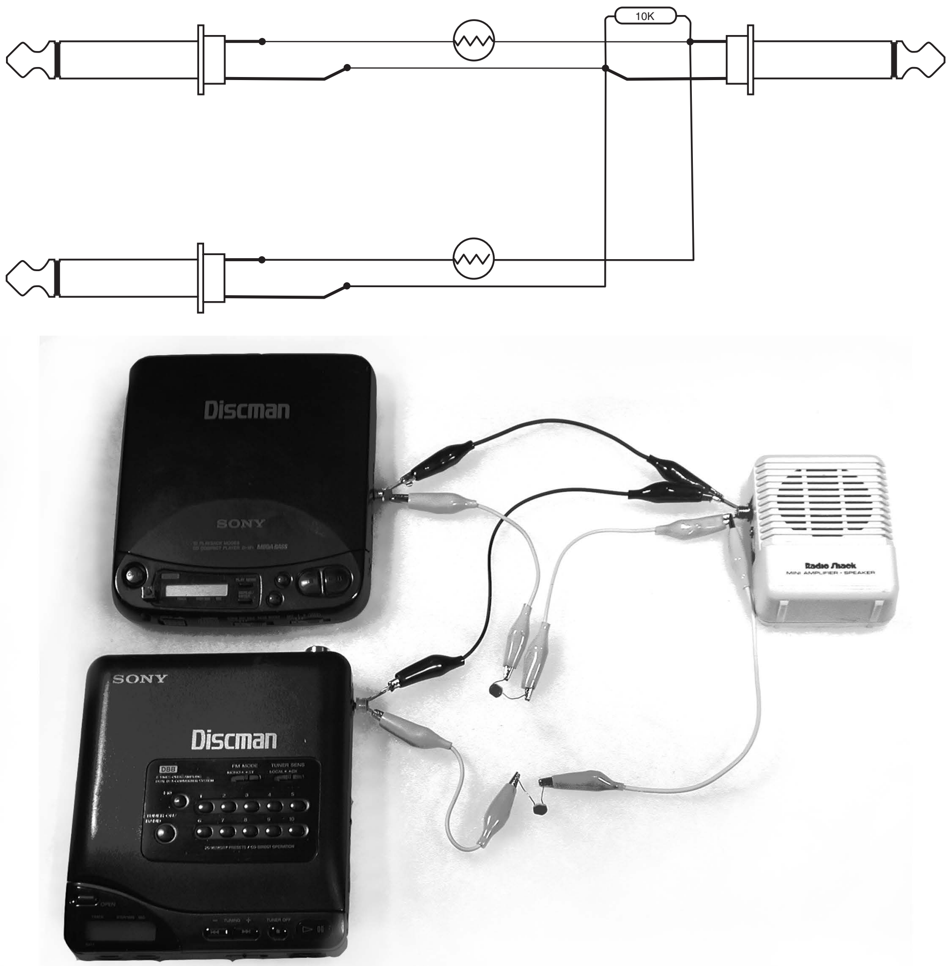


Figure 17.4 Light-controlled mixer.

signal/tip from ground/shield, but of course you can wire female jacks instead—as you probably would if you built the parts into a box of some sort.

Flashlight-controlled circuits like these (as well as our earlier photoresistor-controlled oscillators) occupy a distinguished place in the history of early live electronic music: similar circuits can be heard in the work of David Tudor, Lowell Cross, and other pioneering hacker-composers. Figure 17.6 shows a four-channel panner designed and built by composer and pianist Frederic Rzewski in 1967.

BLINKIES

An LED is a small and cheap source of light that can be controlled electronically. Get one (or, better yet, two) in the color of your choice. LEDs are polarized, like batteries, electrolytic capacitors, and the ordinary diodes we used to mix our oscillators in

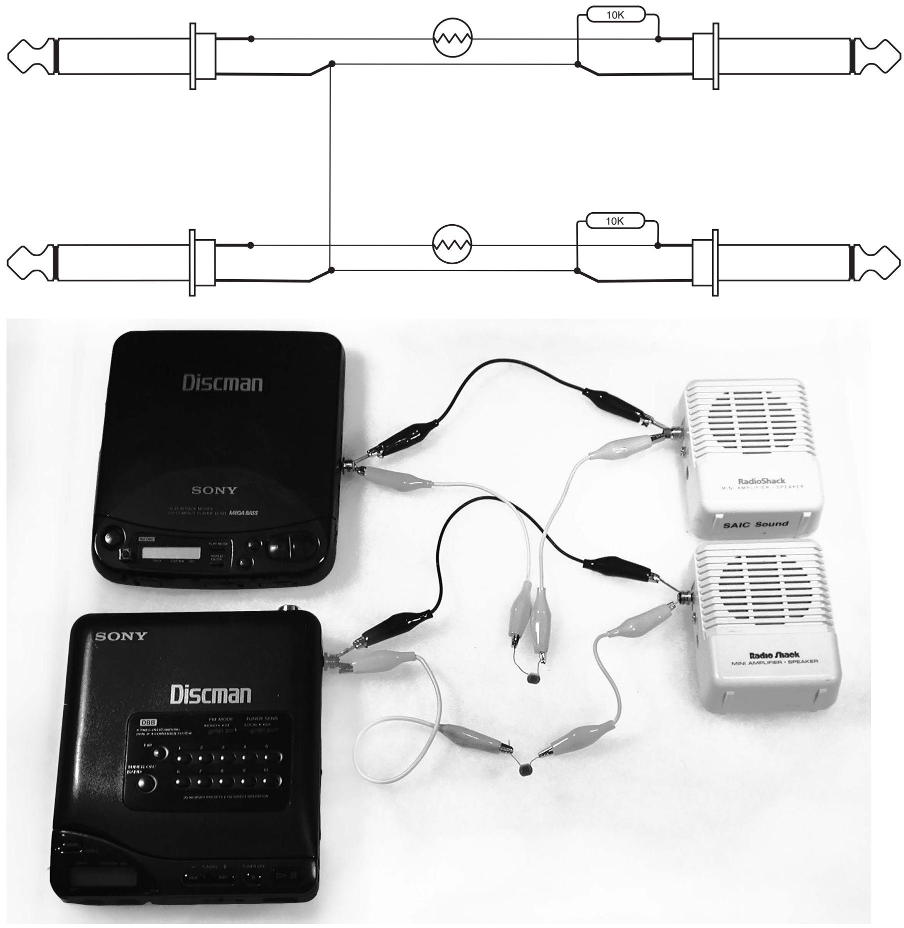


Figure 17.5 Patchable optical panner/mixer.

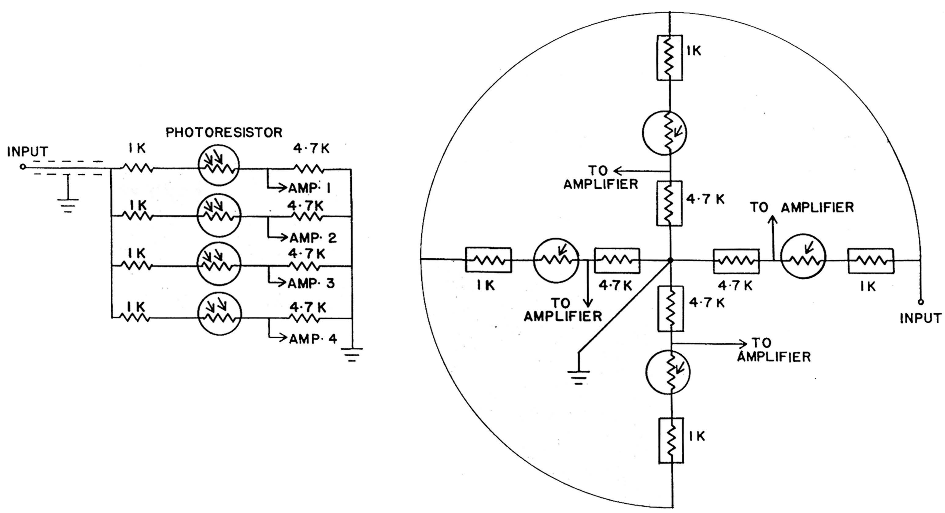


Figure 17.6 Four-channel panner by Frederic Rzewski (1967): schematic diagram of one channel (left) and physical arrangement of one channel (right).

Chapter 13; they only light when current flows through in one direction, not the other. Notice that one leg of the LED is shorter than the other, and if you look closely you will see that the lower rim of the LED is slightly flattened above the shorter leg—the short leg and flat side indicate the "−" connection of the LED, and the other leg is the "+" connection (Figure 17.7).

Breadboard the circuit shown in Figure 17.8. It should light up. Swap the polarity of the LED and observe that it only lights in one orientation. Substitute different values for the resistor and note the change in brightness: the smaller the resistor, the brighter the light, but only to a point, after which the LED might burn out or the circuit driving it (our next step) will start to misbehave. Don't use a straight wire—1 kOhm is a good value to start with if you are using a 9-volt battery.

Rule #22: Always use a resistor when powering an LED, otherwise the circuit and/or LED might blow out.

Breadboard the circuit shown in Figure 17.9. This uses the simple oscillator from Chapter 13 (you could use a 4093-based design from Chapter 15 if you prefer), but

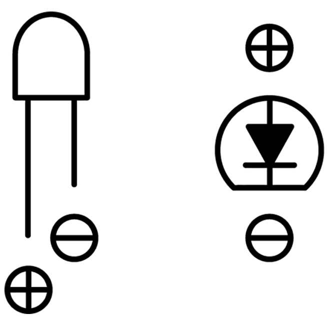


Figure 17.7 LED orientation.

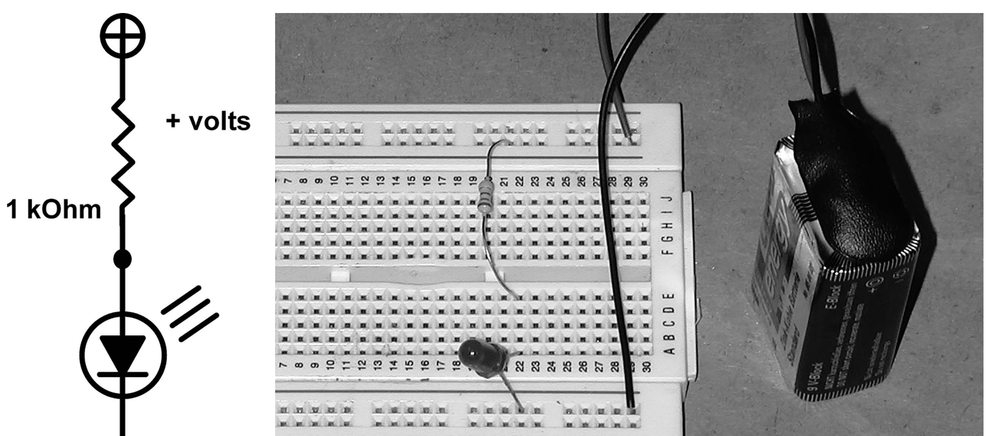


Figure 17.8 Lighting an LED.

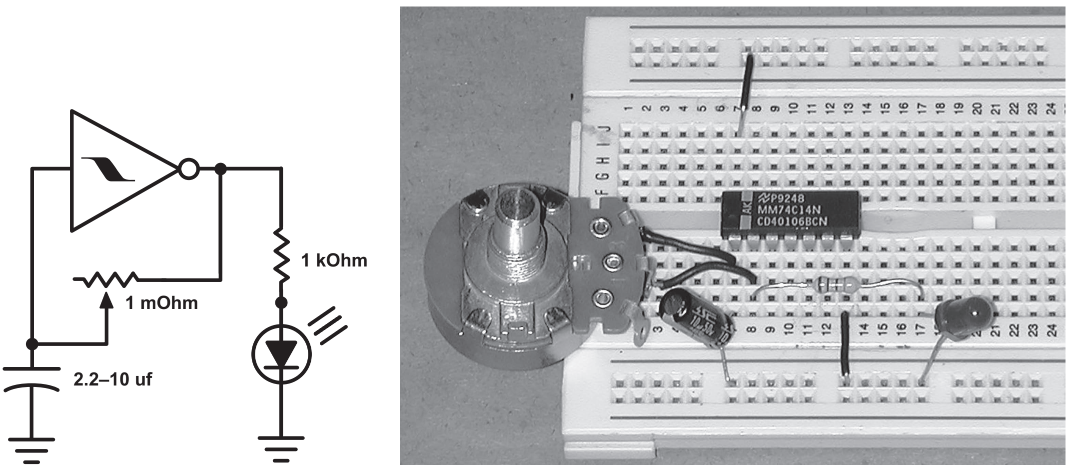


Figure 17.9 Blinking an LED with an oscillator.

now we are connecting the oscillator's output to an LED instead of to an amplifier. The LED should blink, and if you also connect pin 2 to an amplifier, you'll hear a tick-tock in sync with the blinking light. Since we want it to blink at an observable rate, we use a large capacitor (2.2–10 uf) and a big pot (1 mOhm) to keep the oscillator in the metronome range, rather than an audio frequency. If it doesn't blink, you probably used too small a pot or capacitor, put the LED in backwards, or have omitted some connection. Vary the speed and watch the effect. Fun enough just to look at, but wait—it gets better!

Now take the LED and nuzzle it up against a photoresistor as shown at the top of Figure 17.10. Spread the leads of the photoresistor and LED apart so they do not touch each other, then wrap the photoresistor and LED in electrical tape so that they are sealed from outside light (right photo). Don't let any legs short together or the circuit will not work—you can wrap some tape on each of the four wires if you're cautious. Replace the LED in the Figure 17.9 circuit with this bundle, making sure that you have the LED's polarity right—check the placement of that shorter leg since, if you've done the wrapping right, you won't be able to see the light blink (Schrödinger's cat!). The photoresistor's legs should be sticking up in the air. Connect the hot/tip of two jacks or plugs to the photoresistor legs as we did in Figure 17.2 and link the connector grounds with another clip lead or solder some wire between them, as shown in Figure 17.11. **Note that you do not connect the ground of the breadboarded**

Figure 17.10 LED and photoresistor kissing (left) and bundled in electrical tape (right).

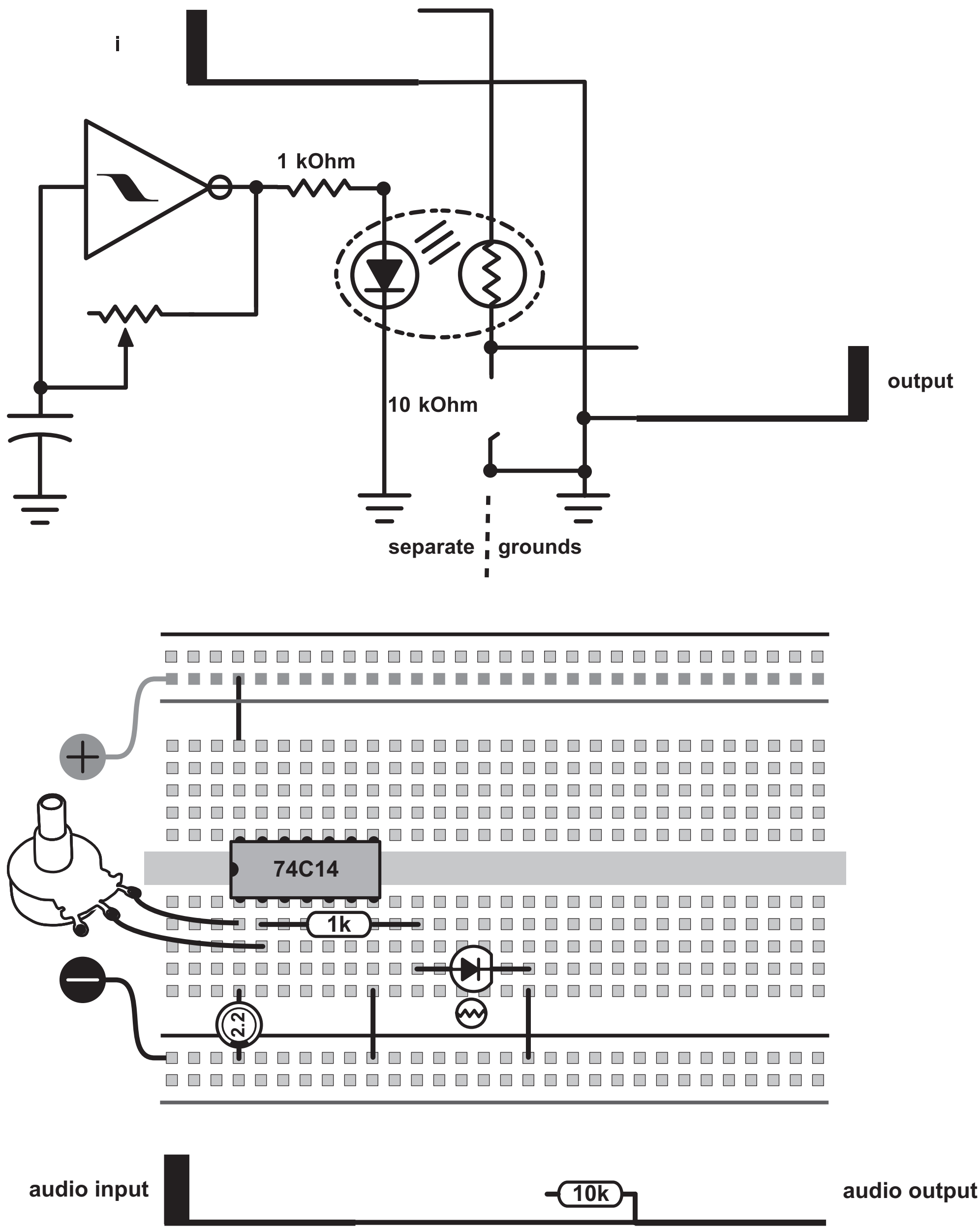


Figure 17.11 A blinking-LED-controlled gate.

circuit to the grounds on the audio jacks—this is important for optimizing the audio quality of this circuit!

Connect an audio source to input jack and connect the clamped output jack to an amplifier. Vary the speed of the oscillator. You should hear your sounds get chopped on and off as the LED blinks. As you speed up the oscillator, the distinct on/off rhythm is replaced by a kind of wobbly modulation. Experiment with different size capacitors until you find a workable range—you may want to add a resistor in series with the pot to limit the maximum speed, as demonstrated in Chapter 13 (Figure 13.14).

If you want to *see* what's happening as well as listen, just add a second LED in parallel as an indicator light, as shown in Figure 17.12.

We can extend this basic oscillator/LED/photoresistor gate design to create an automated version of our flashlight-controlled panner/mixer. The circuit shown in Figure 17.13 uses one stage of the hex inverter to make a low-frequency oscillator

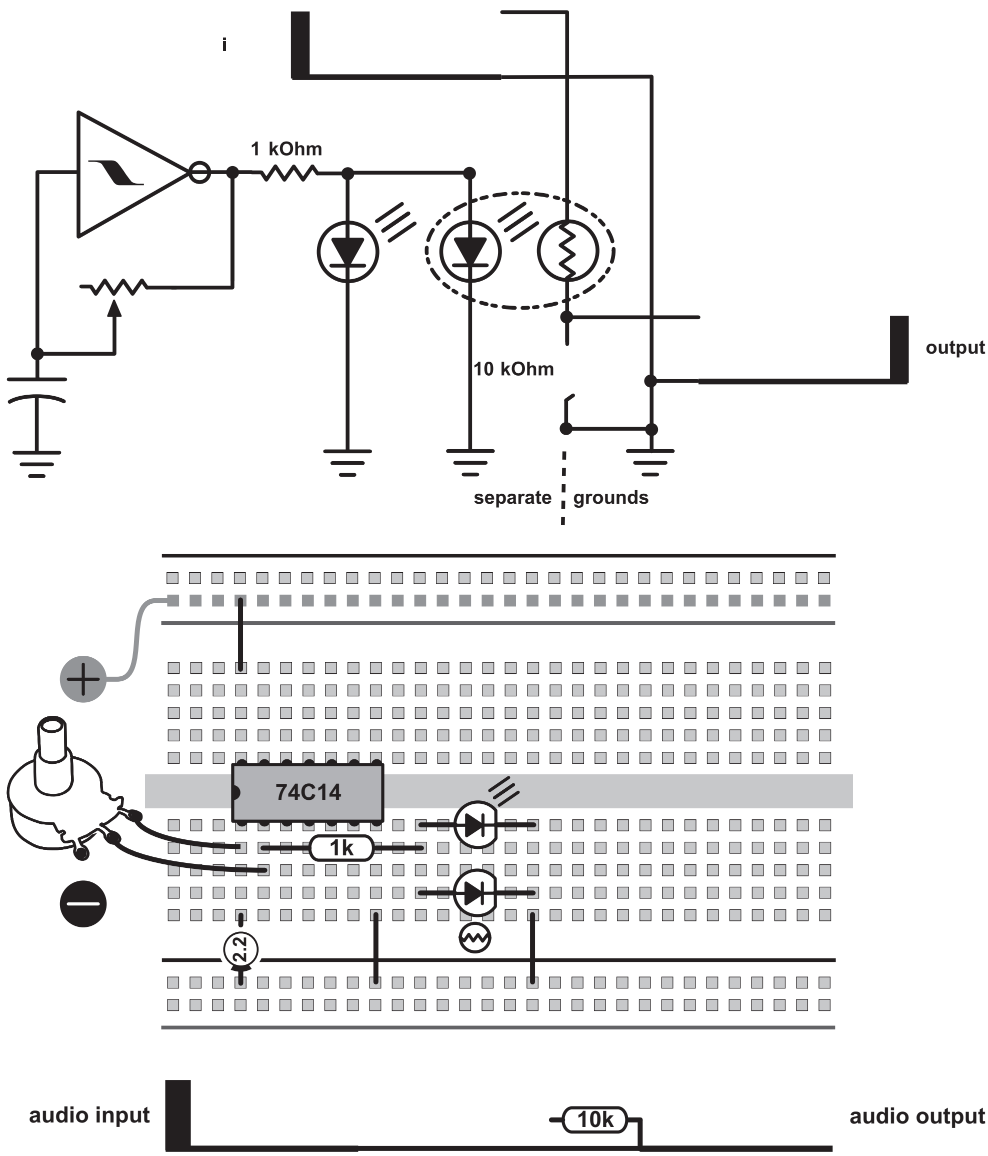


Figure 17.12 A blinking-LED-controlled gate with second indicator LED.

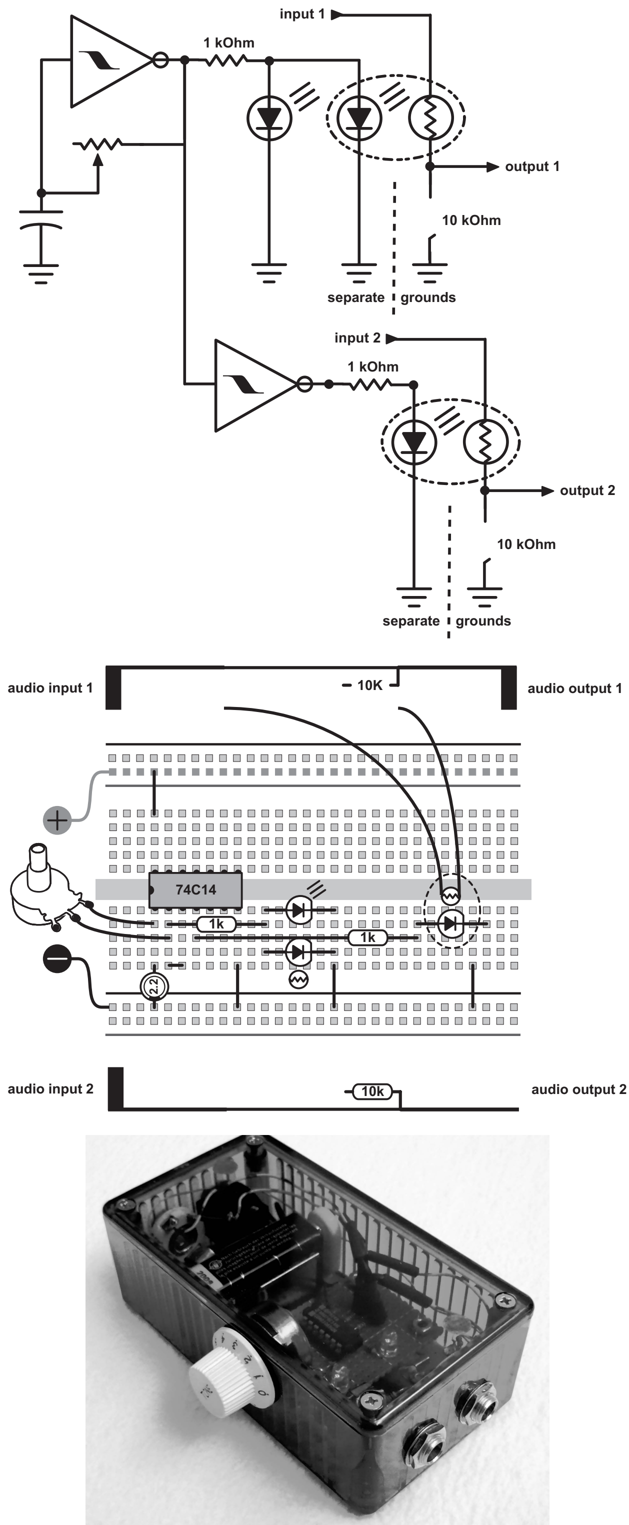


Figure 17.13 A blinking-LED-controlled panner/mixer; assembled module from the collection of Robert Poss.

while a second stage simply inverts the clock signal generated by the first. The jacks have been omitted from the schematic for clarity—just expand the hookup arrangements from the previous two circuits. By connecting a separate LED to each inverter, each via its own resistor, one LED is *off* when the other is *on* (remember how an inverter always outputs the opposite state of the input signal?). Note that when wiring a stage of the 74C14 as a simple inverter, you need no capacitor or feedback resistor, as you do for an oscillator.

Use a Y-cord or clip leads to connect one audio source to both inputs and connect each photoresistor output to a separate amplifier; adjusting the oscillator frequency changes the panning speed—Psycho-Pan-Scan! Now hook up two different audio sources to the two circuit inputs, mix both circuit outputs to one amplifier input (Y-cords or clip leads), and adjust the oscillator to cut between the two signals —Super Crab! Remember to link all the input and output jack grounds together, but (as with the other gate circuit) you do not connect the oscillator's ground bus to the ground of the audio jacks.[1]

The basic concept of the blinking LED chopping audio can be extended from simple oscillators to more complicated control circuits—driving the LED with the output of the cascaded gated oscillators we made in Chapter 15 yields weird rhythms. Just don't forget to include a resistor before the LED.

As with the flashlight-in-the-closet experiment, these circuits do not produce a total mute when off—some of your audio signal will continue to bleed through even when the LED is off, and even if you use the 10 kOhm clamp resistor. The amount of bleed depends on the specific photoresistor used, how effectively it is shielded from outside light, and the intensity of the LED (a brighter LED will give a wider dynamic range). Select a photoresistor with as large a difference as possible between on- and off-resistance—they're usually best picked by ear, by substituting different choices into the circuit, although a data sheet (if available) can help. Bear in mind that the slight leakage of audio during the "off" state will most likely be masked by other sounds in your mix.

If the masses of electrical tape offend your sensibilities, you can put the photoresistor and LED inside an opaque soda straw or the plastic sleeve of a mini-plug or guitar plug—you may want to put some Blu-Tack or opaque silicon sealant into the ends of the tubes to prevent light leakage. Once again, be careful to avoid shorting the leads against one another.

Heat-shrink tubing is another tidy solution to light isolation (see Figure 17.14). Slide narrow pieces around each leg of the LED and photoresistor to insulate them from each other. Slide a wider piece over the LED and cell, nuzzle the two components

Figure 17.14 LED and photoresistor in heat-shrink tubing, before shrinking (left) and after shrinking (right).

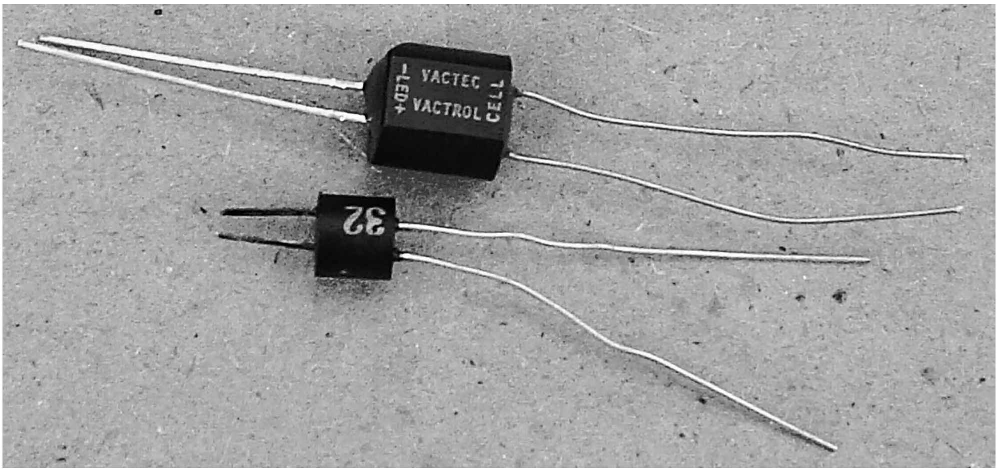


Figure 17.15 Pre-packaged audio opto-isolators.

tightly together, and apply heat from a hair dryer to shrink the tubing tight around them. Voilà! A microelectronic "Bruit Secret."

One can also buy pre-packaged LED-photoresistor modules that are well suited for insertion into the circuits we've just built (see Figure 17.15). The most widely used is the Vactrol by PerkinElmer, but they're not always easy to find, and rolling your own can be oddly satisfying anyway. Caution: the Vactrol is sometime referred to as an opto-isolator; many versions of opto-isolators are available from many sources, but most of them are intended for digital applications and do not employ the photoresistor necessary for controlling an audio signal.

Optical gating and dynamic control are much prized by audiophiles for sonic purity. The complete separation of the audio ground from the oscillator's ground contributes to the inherently high quality of the sound (which is why I stressed that you should keep the grounds separate). Only a few small (if confusing) additions stand between these simple circuits and some very expensive studio noise gates, compressors, and limiters.

OTHER USES FOR PHOTORESISTORS

As you should grasp by now, the photoresistor is a resistor like any other, but for its Nosferatu-like response to light. You can substitute a photoresistor for almost any resistor in almost any circuit and then modulate that resistance with light—either performed (flashlights, shadows, etc.) or automated, as we showed in this chapter using blinking LEDs. If you already have a circuit whose pitch is controlled by a photoresistor, wire up an oscillator-driven LED, press the blinking light against the cell, and listen to what happens.

Modulating the pitch of a photoresistor-controlled audio oscillator with an LED blinking at a genre-appropriate beats-per-minute yields a pleasingly disco-tinged

"Syn-Drum" swoop. Just take any of the audio oscillator designs from the previous chapters and replace the photoresistor in the feedback loop with one of our LED-photoresistor bundles. Drive the LED from a second, low-frequency oscillator (use the design in Figure 17.9). Add a pot in series or parallel with the photoresistor to adjust the sweep range.

Sometimes, if its on-resistance is low enough, a photoresistor can be substituted for a low-current switch. If you have a circuit-bent electronic toy with switches to trigger sounds or enable functions, try paralleling one of those switches with a photoresistor: connect the two photoresistor legs to the points on the toy's circuit board that are joined when the switch is closed. This is very effective when the toy has the kind of switches that consist of trace patterns on the circuit board shorted by a rubbery switch. Drive the LED with a slow oscillator. The function associated with the switch should be triggered when the LED is on. If it works, you've got a simple solution to automating some of the toy's functions; if not, try another switch or another toy.

Don't use the photoresistor as a substitute for the power-on/-off switch in a circuit or between the circuit and its speaker, since it can't pass enough current. But the photoresistor switch is a convenient way to extend the duration of toy playback, especially of slowed down samples, by automatically and repeatedly "pressing" the same button; they can also be used to press various switches at different rates to produce quasi-random results. (See Chapter 21 for more information on this kind of advanced circuit bending.)

NOTE

1. A tip for advanced hackers: you can substitute the normally-closed connections on switched audio jacks for the Y-cords.

CHAPTER 18

Mixers and Matrices
Very Simple, Very Cheap, Very Clean Ways of Configuring Lots of Circuits

You will need:

- An assortment of sound-making and sound-processing circuits, found or made.
- A few pots of the same value (approximately 10 kOhm–100 kOhm), preferably audio taper.
- Assorted resistors and photoresistors.
- An unwanted computer keyboard.
- Some solid and stranded hookup wire.
- Assorted jacks and plugs.
- Some clip leads.
- An amplifier or two.
- Soldering iron, solder, and hand tools.

MIXERS

With your collection of noisemakers growing, a mixer might prove useful. Here are some completely *passive* circuit designs—they use no batteries, no chips, no circuit boards, no on/off switches. Even if you already have a "real" mixer, these offer a convenient way to expand your inputs and outputs.

The simplest mixer of all is just a Y-cord tying two signals together. In Figure 18.1 a CD player and the output of a distortion pedal mix into a graphic EQ through a Y-cord. Remember earlier, in Chapter 13, when I said you shouldn't do this with the output of our basic oscillator circuit because it introduces existential angst that can stop oscillation? Professional audio designers assume users will do the stupidest thing: a Y-cord mixer is pretty high on the stupid scale (or low, depending on how you orient the chart), so they include resistors in the outputs of their circuits—everything from phones to fuzztones—so that if their boxes shorted together, they will mix safely just like our oscillators did after we added resistors. You can keep adding channels to this snaky mixer by buying more Y-cords and patching them together until they resemble the family tree of a royal family.[1]

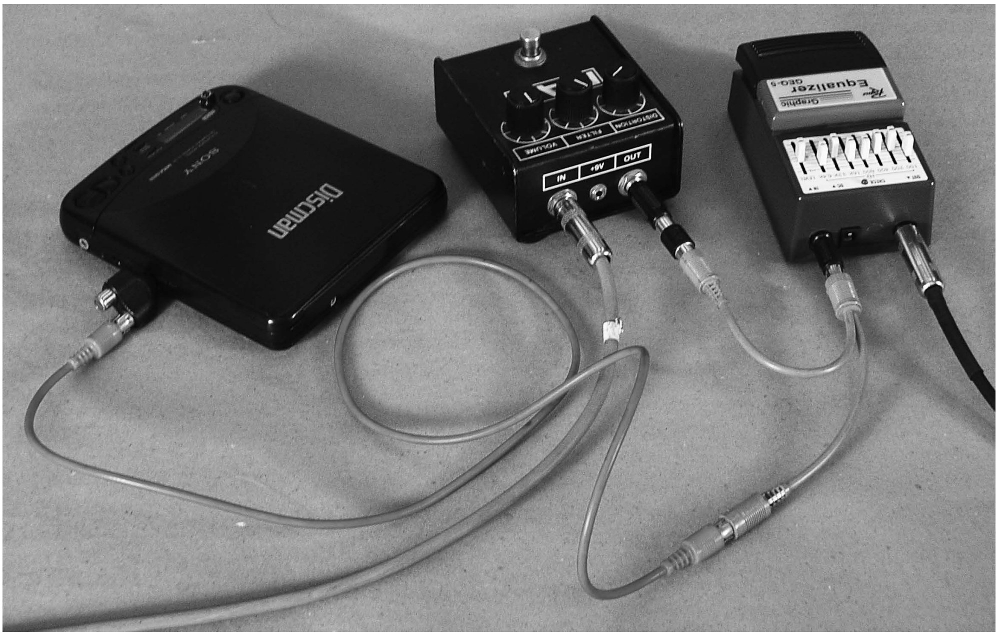


Figure 18.1 Y-cord mixing.

If, on the other hand, you want to mix and match commercial electronic devices (with built-in resistors) and weird stuff (with unknown output characteristics) built by you and your friends, you need to increase your mixer's complexity ever so slightly. We can build a nice, safe-for-all-circuits passive mixer with nothing more than a jack and a resistor for each input needed (Figure 18.2). Each input jack connects to the output jack through its own resistor; all the grounds are tied together as well. You can expand this design for any number of inputs: just add a jack and resistor for each new channel.

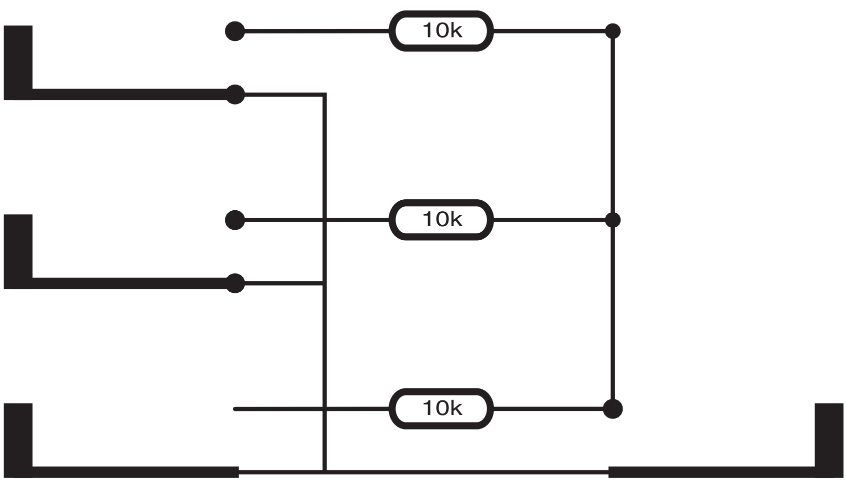


Figure 18.2 Fixed resistor mixer (inputs on left, output at right).

"Wait," you protest, "these aren't mixers! Where are my knobs? How can I adjust the levels?" You may have noticed that the majority of audio devices, from delay pedals to laptops, have controls for adjusting output volume. If you plug them into our fixed resistor mixer, each device's own volume control does the job of the corresponding channel fader you'd expect to see on a "real" mixer. Tacky, yes, but poverty is the mother of invention, and, as you'll soon find out, potentiometers can be the most expensive part of a mixer. If you can do without, leave them out.

On the other hand, if you are post-economy and looking to park your portfolio in potentiometers, let's advance to some more familiar-looking mixer designs.

Figure 18.3 shows the schematic for a basic three-input mixer. The signal (i.e., tip) of each input jack is connected directly to one ear of a pot. The center tap on the pot is soldered to a summing resistor. The free ends of all the summing resistors are tied together and connected to the signal/tip of the output jack. The grounds/shields of all the jacks are connected together with wire and soldered to a similar wire linking the other ear of all the pots. Figure 18.4 shows which ear of the pot to use for input and which for ground if you want a traditional mixer behavior (i.e., turning the knob clockwise raises the level). The pots can be of any value from 10–100 kOhm, but it's best if all the pots are the same. The summing resistors and pots should be about the same value, i.e., for 10 k pots use 10 k summing resistors, for 50 k pots use 50 k summing resistors, etc.

You can expand the circuit in Figure 18.3 to accommodate any number of inputs simply by adding another jack, pot, and resistor for each new input signal. By including some switches and a second output jack, you can add stereo outputs to your mixer, as shown in Figure 18.5. The ground interconnections have been omitted from the drawing for clarity's sake; instead of looping lines, the common ground bus is indicated by

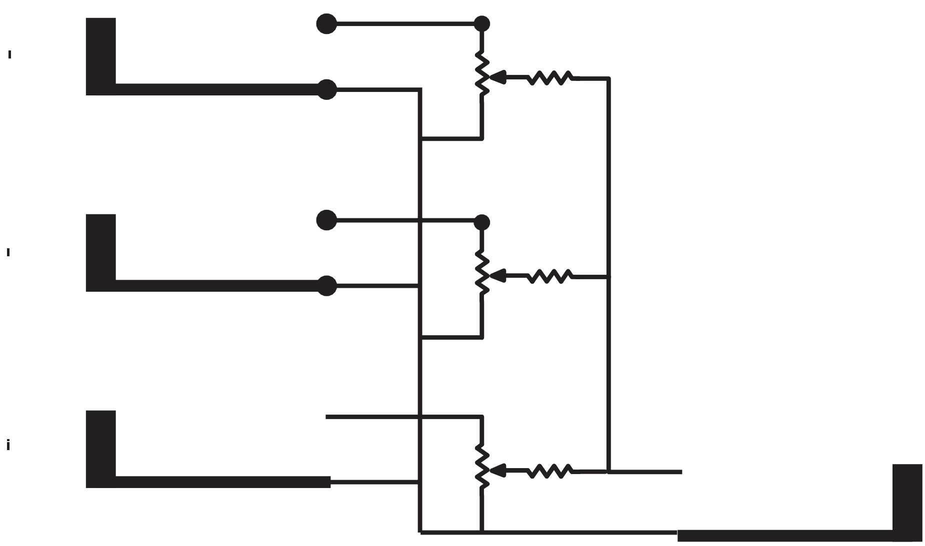
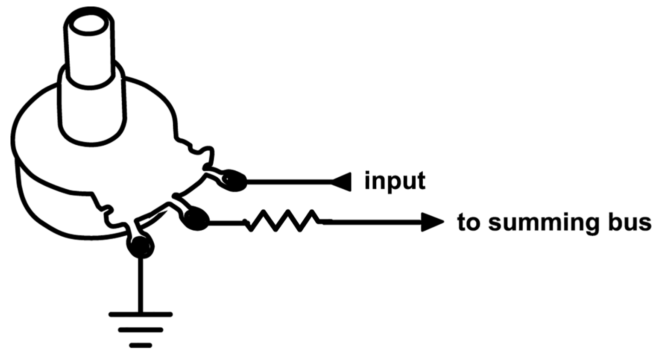


Figure 18.3 Basic three-input mono mixer.

Figure 18.4
How to wire an audio fader.



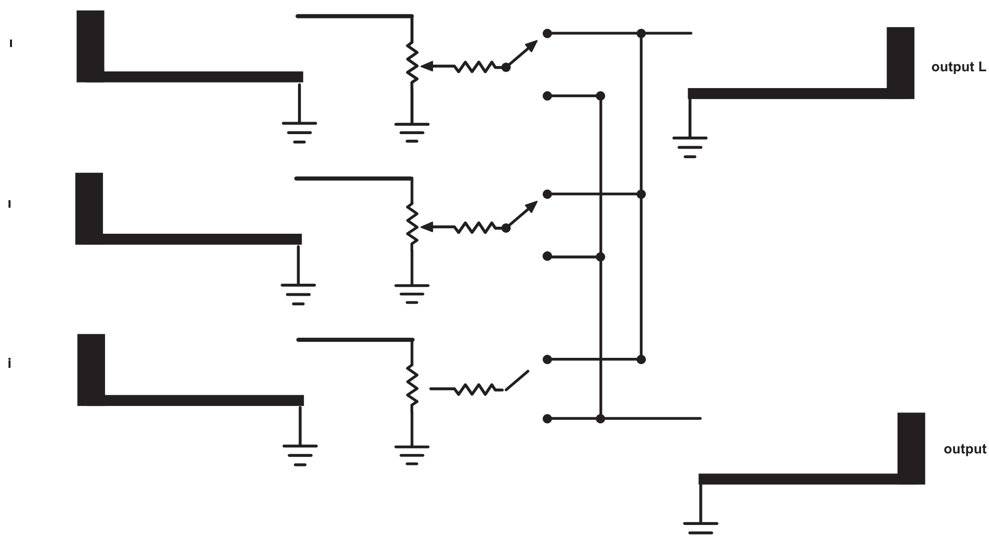


Figure 18.5 Basic three-input stereo output mixer.

the runic ground symbol. Each summing resistor is connected to the switch's common terminal (C); the normally open terminal (NO) is connected to one output jack and the normally closed (NC) to the other. Throwing the switch swaps the signal between the left and right outputs. It's a kind of binary panner, alternating between hard left and hard right, with no gradations in between—crude, but simpler (and cheaper) than adding a proper pan pot.

Pots are specified as "linear taper" (what we've been using so far in this book, often labeled "B") and "logarithmic taper" (sometimes known as "audio taper" and often designated "A", as in "Audio"). Raising and lowering the volume feels and sounds smoother with a log/audio taper pot since our ears respond to changes of volume on a logarithmic scale, and audio faders use a log curve. You can get them from the same online retailers as linear taper pots, but you may not have as wide a choice of values at budget prices. Audio taper pots are generally preferred for mixers, but linear ones will suffice if budget demands.

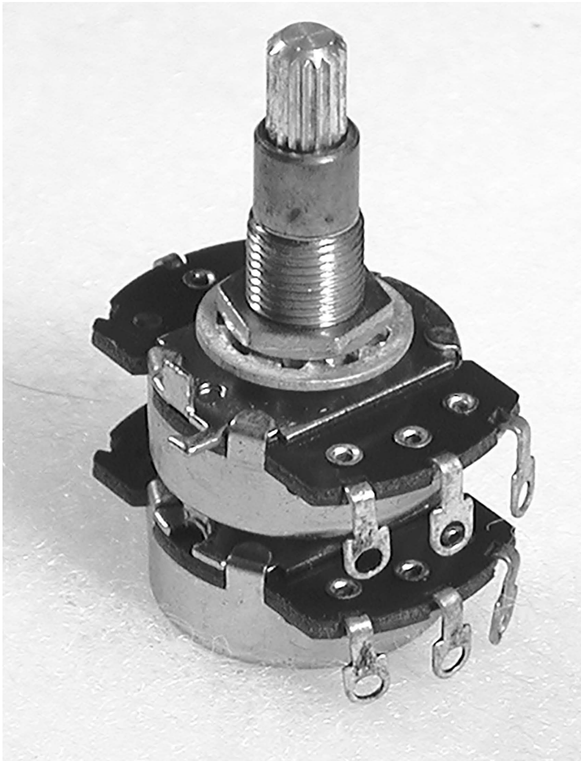


Figure 18.6
A stereo potentiometer.

If you are mixing a lot of stereo signals (like the outputs of phones or laptops), you may want to use stereo pots, which are two pots coupled to a single shaft; this mechanical design lets you adjust two signals in parallel from a single knob (Figure 18.6). One pot in each stereo pair will take the left half of a stereo input and send it to your left bus while the other will take the right input and send it to the right bus—like our stereo mixer, but without the switches. You wire each section exactly like an ordinary single pot.

You can use rotary pots or slide pots (easy to find through online sources), which make your circuit look more like a "real" mixer, but be warned:

Rule #24: It is easier to drill round holes than slots.

Because they contain no preamps or other gain circuits, these designs are best suited for mixing line-level signals of more or less similar strength (i.e., phones, laptops, audio recorders, or a gaggle of oscillators). They cannot boost a microphone signal up to line level but can only attenuate line-level signal down to that of a mike—which is awkward, unless you plan to patch the output of this mixer into a mike preamp (for a mixer with gain, jump ahead to Chapter 23). But what these designs lack in gain and fancy features they make up for in low cost and high audio quality—as with the optical gating and panning circuits in Chapter 17, passive mixers are coveted by audio purists.

MATRICES

David Tudor, one of the pioneers in the field of live electronic music, used matrices to combine relatively simple circuits into complex networks that produced sound of surprising richness and variation (see You Nakai's and Michael Johnsen's essay on the website). Instead of simply mixing a handful of sound sources down to a stereo signal, Tudor interconnected his circuits with multiple feedback paths and output channels. The rise of the "no input mixing" school of internal machine feedback a few decades later exposed a new generation of musicians and listeners to the possibilities of matrix feedback. Consider the 3 × 3 matrix mixer shown in Figure 18.7. (Input and output jacks have been omitted for clarity.)

You will notice that the design is similar to that of our basic mixer in Figure 18.3, but here each input signal is connected to three pots instead of one, and we have three output buses. You can expand this circuit with as many pots and jacks as you need and can afford. Figure 18.8 shows some hacker-built matrix mixers using this design.

Connect a few circuits, including both sound-generating circuits, such as your oscillators or toys, and some processing circuits, such as the photoresistor gate, a distortion circuit, or other guitar pedals (such as a delay, wah-wah, or graphic equalizer). Send one output of the matrix to an amplifier for listening, and the others can be sent

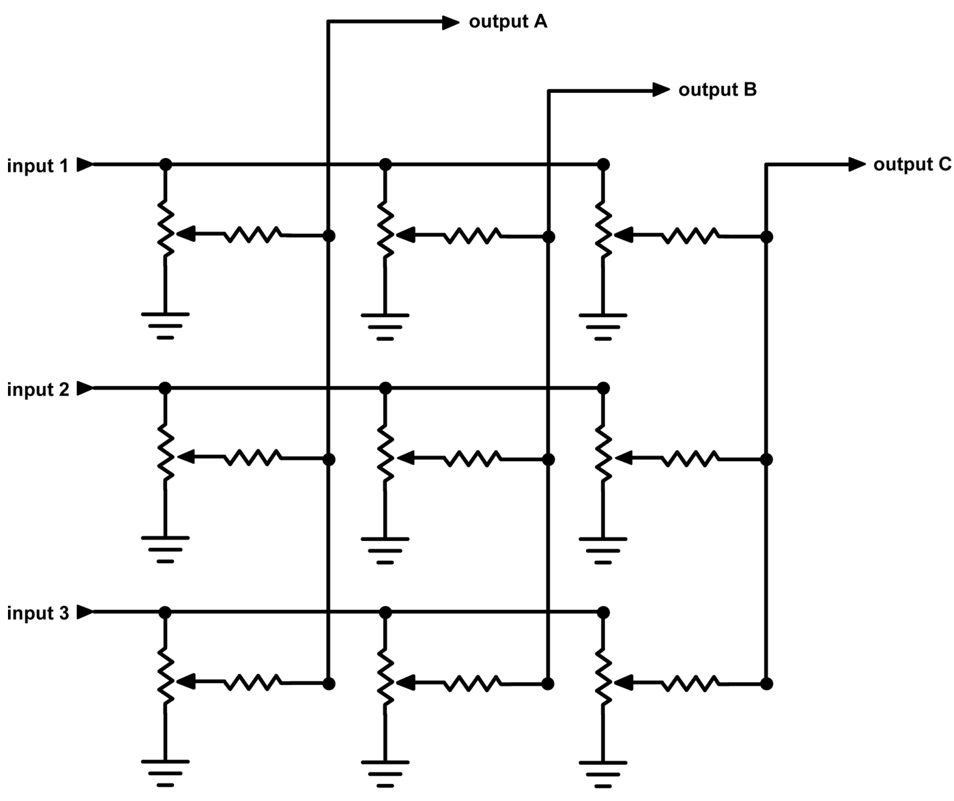


Figure 18.7 3 × 3 matrix mixer.

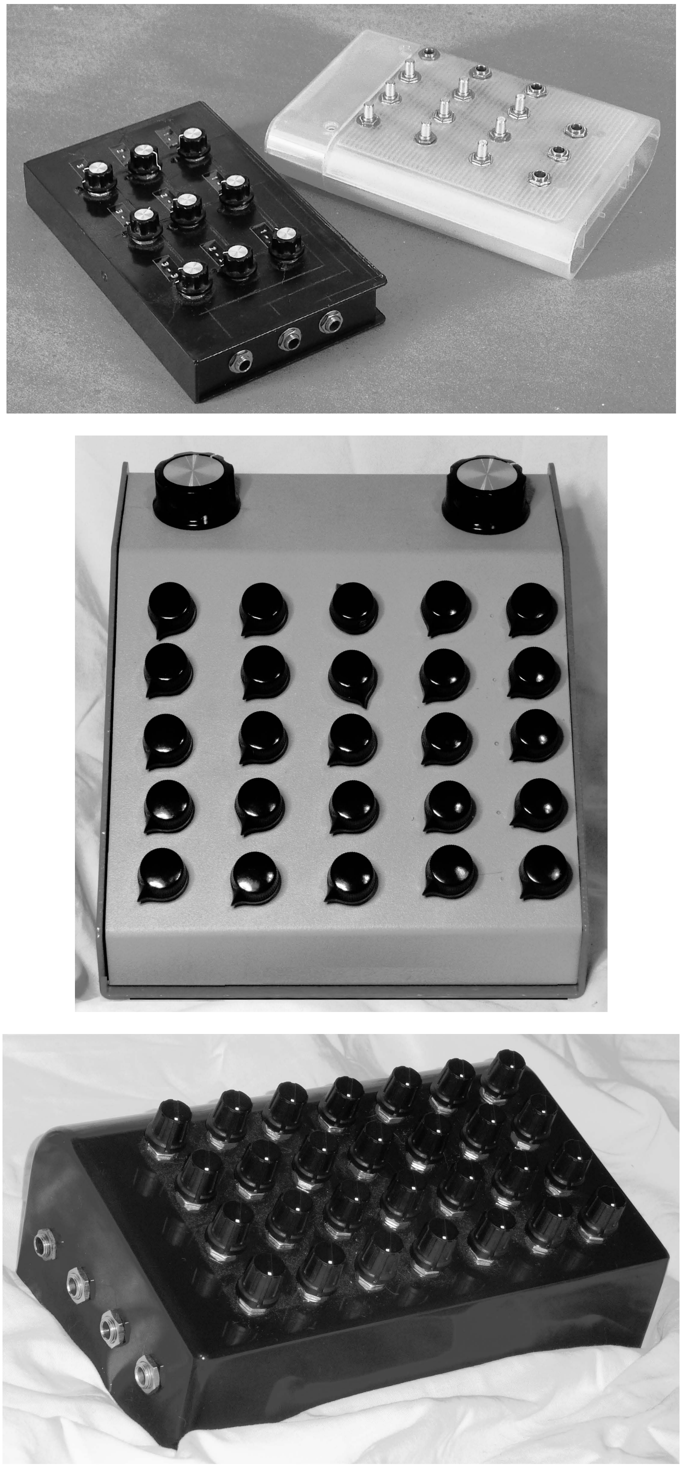


Figure 18.8 Matrix mixers by Alex Inglizian (top: upper left) and James Murray (top: upper right), Douglas Ferguson (center), and Steve Marsh (bottom).

to the inputs of your sound-processing circuits. By adjusting the levels of the various pots, you can create a straightforward signal path (toy through distortion to speaker) or a more devious one (toy through distortion to speaker, distortion also to delay, which goes both to speaker and back into its own input).

The transducer-based pseudo-reverbs discussed in Chapter 8 work beautifully in these configurations. Some of the most unassuming rock pedals reveal astonishing musicality when placed in feedback loops. Incorporated into matrices, time-based effects (such as delays and flangers) contribute instability that transforms a table of commonplace effect boxes into a richly challenging performance instrument (Figure 18.9).

If you intend to use a matrix to generate feedback, you will need some gain, which you can provide with almost any effect pedal. Feedback matrices benefit greatly from the inclusion of some kind of equalization to aid in steering pitch response and nulling out unwanted shrieks—a simple graphic EQ effect pedal provides both the requisite level boost and useful frequency shaping.

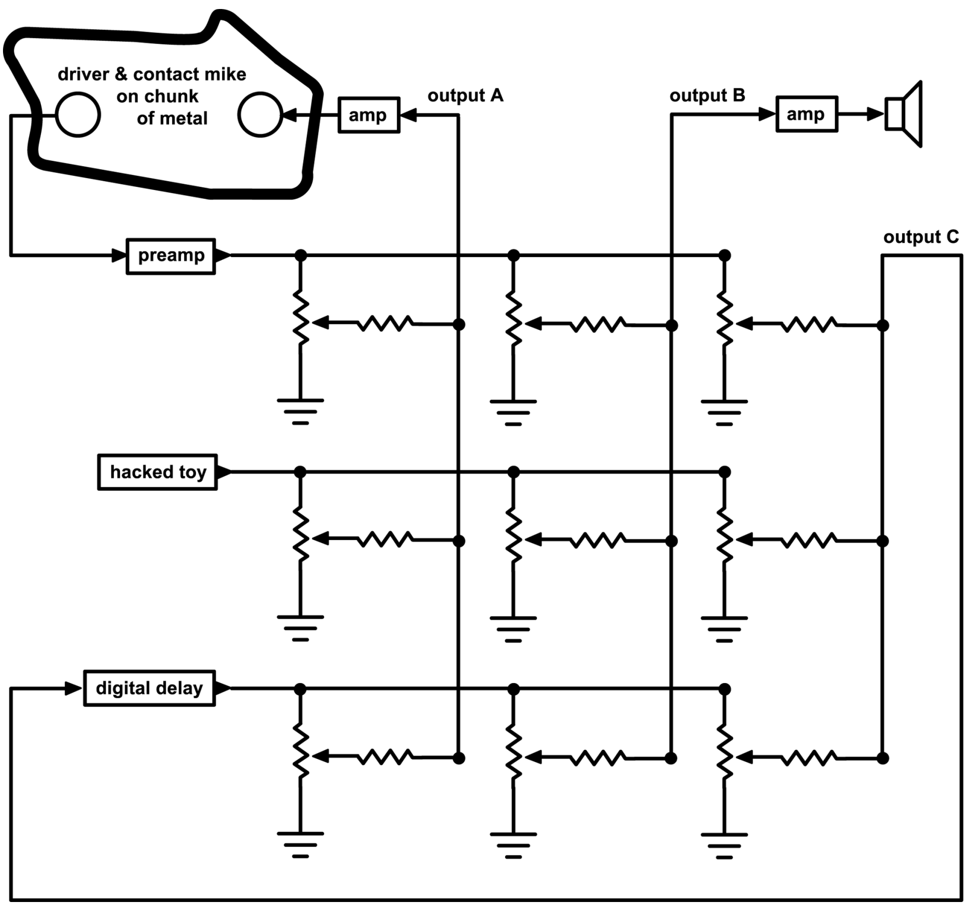


Figure 18.9 Circuits in a matrix.

A last word of advice on matrix mixers: when using passive ones such as ours, it behooves you to keep any unused inputs turned fully off. This reduces the crosstalk between channels.

COMPUTER KEYBOARDS

Discarded computer keyboards constitute a significant percentage of e-waste. Why not spare the landfill by building a useful musical tool? A keyboard consists of a switch matrix: instead of each key closing two discrete contacts (like a doorbell), it bridges specific lines in an X–Y grid, as shown in Figure 18.10.

Pressing Key 1 connects horizontal row 1 to vertical column A, Key 3 connects row 1 to column C, Key 4 connects row 2 to column A, etc. In a computer keyboard, there are enough rows and columns to handle the full alphabet, plus numbers and all those extra function keys. A 10 by 8 matrix, for example, accommodates 80 keys. The computer scans the matrix to detect which key is depressed by sending a pulse down each row and checking which column it exits (like a high-tech version of Splat the Rat). If you open up the keyboard and scrutinize the circuit, you'll notice that the traces are arranged in a vague grid (Figure 18.11).

Solder a wire between the ground terminals of two female jacks and attach a clip lead to each of the hot terminals. Connect a loud sound source, such as an oscillator, to one jack, and connect the other jack to an amplifier. Use the clip lead to connect the test signal to one of the traces, which are probably routed to connectors at the edge of the circuit board (Figure 18.12). Hold down any key and touch the amplifier lead to each of the other traces, one at a time, until you hear your sound. Then release the key: if the sound toggles on and off, you've found a cross-point in the matrix. Note this information somehow: use a sticker to mark the top of the key

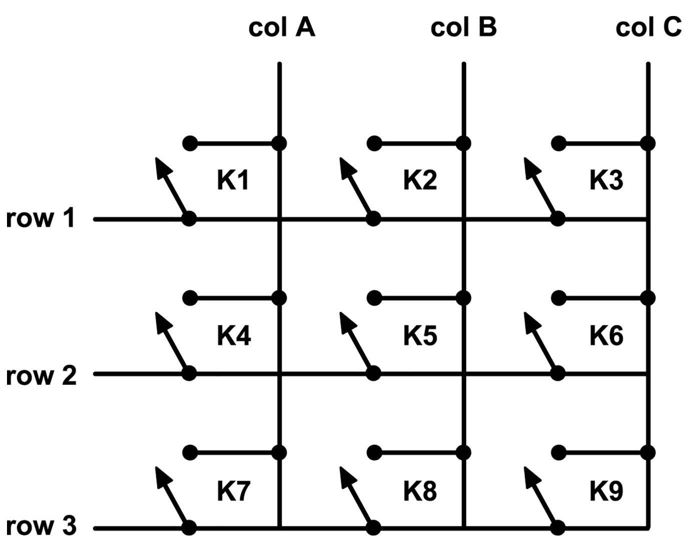


Figure 18.10 A keyboard matrix.

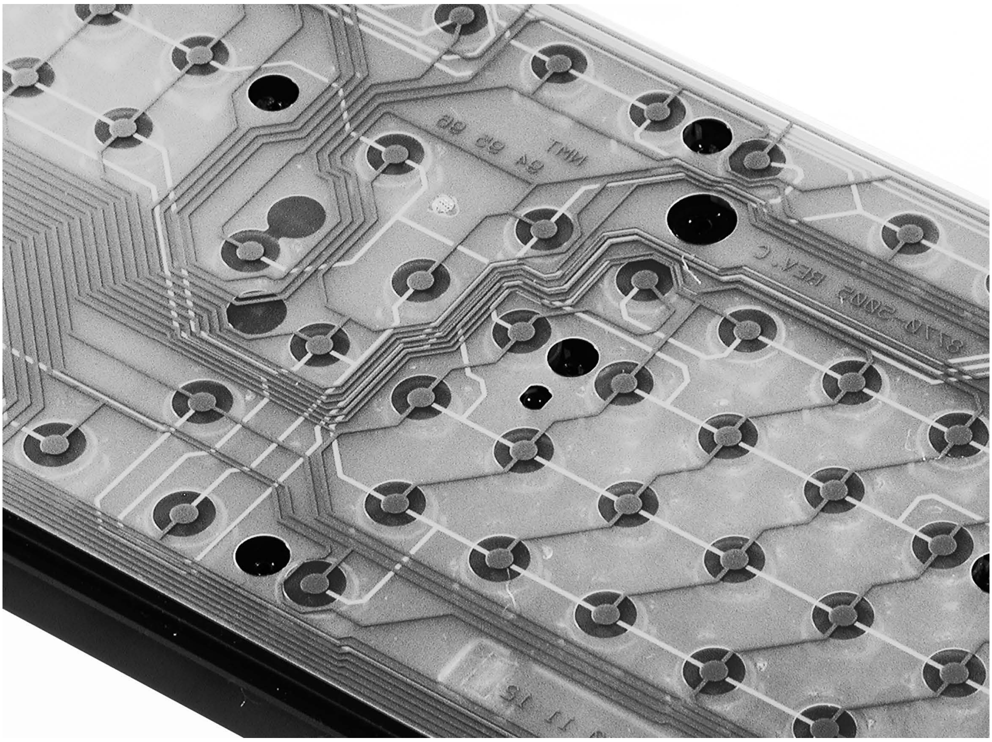


Figure 18.11 Computer keyboard circuit board showing matrix traces.

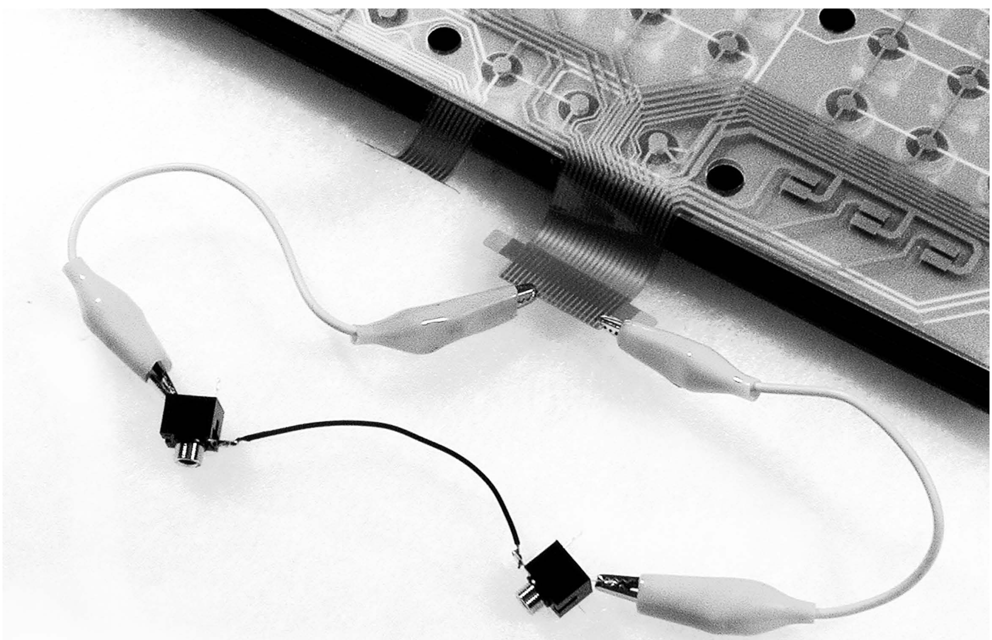


Figure 18.12 Computer keyboard wired for testing audio routing.

or start a chart in which you keep track of the keycap legend that corresponds to a particular cross-point (i.e., the letter *g* connects row 4 to column 3). If it doesn't turn off when you release the key, you've just touched another point along the trace that the sound source is connected to, so keep testing other points until you find a functional cross-point.

By repeating this admittedly arduous process, you should be able to decode the matrix into rows and columns. Solder each row and column to the hot terminal of an audio jack. Connect all the shields together. Use all the rows as inputs and all the columns as outputs or vice versa. By pressing keys you can route any input to any output. You can use this device alone—as a signal router for spatial distribution, for example—or in conjunction with the matrix mixers described prior to add switching to matrix-based signal processing.

If this decoding process sounds too daunting, you can find smaller, more easily mappable switch matrices in touch-tone telephones (3 columns × 4 rows), calculators, or various toys—all frequently discarded items. Raw matrix keypads of various sizes can also be bought from a number of online retailers. Membrane switches, commonly used inside inexpensive keypads and keyboards, have the advantage that, in addition to being able to close the switches by direct button pressure, when opened they can often be "played" by drawing across the surface with a stylus or rolling a billiard ball. For a more aggressive matrix array, Seth Cluett wired up his own matrix out of a bunch of arcade game switches (Figure 18.13).

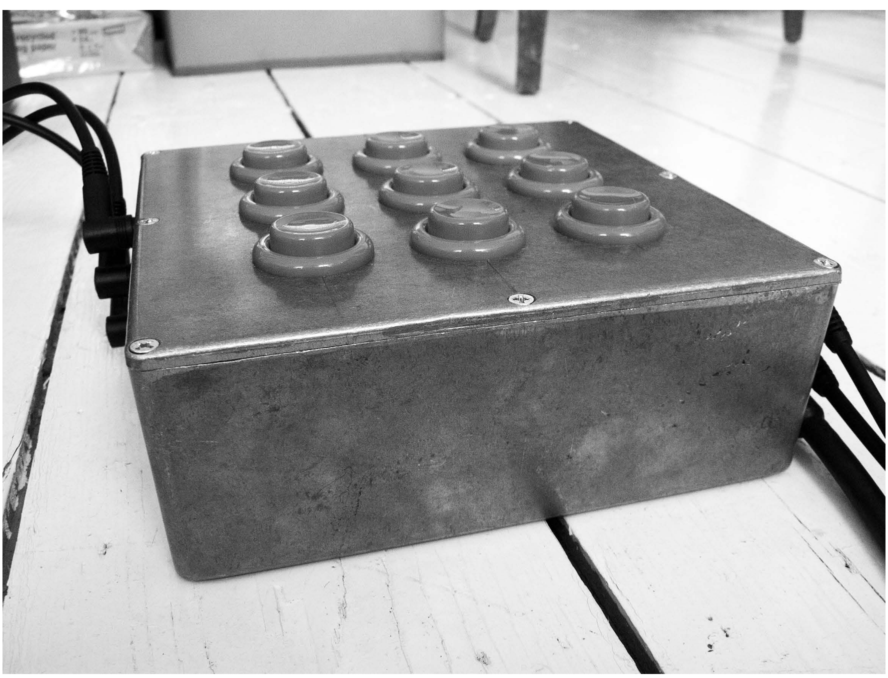


Figure 18.13 Seth Cluett, 3 × 3 audio matrix built with arcade game switches.

AUTOMATION

The basic photoresistor gating circuit described in Chapter 17 can be expanded into a matrix as well (Figure 18.14). The photoresistors can be activated by flashlights, ambient light and shadow, video or film projection, or oscillator-driven blinking LEDs, depending on the degree of control or indeterminacy you desire. For maximum dynamic range, remember to cover the backs of the photoresistors with electrical tape or place them in opaque lightpipes (like sections of soda straws or heat-shrink tubing).

COMING TOGETHER

The outputs of any of these various mixer designs can be patched directly to power amplifiers (guitar amps, stereos, etc.) or sent through DI boxes to a PA system. You can patch the outputs of your homemade mixer to any unused inputs of a commercial one—many mixers have "effect returns" that are perfect for this purpose.

These designs here work fine as stand-alone mixers (see the examples back in Figure 18.8) or for expanding an existing mixer. But they are cheap and simple enough

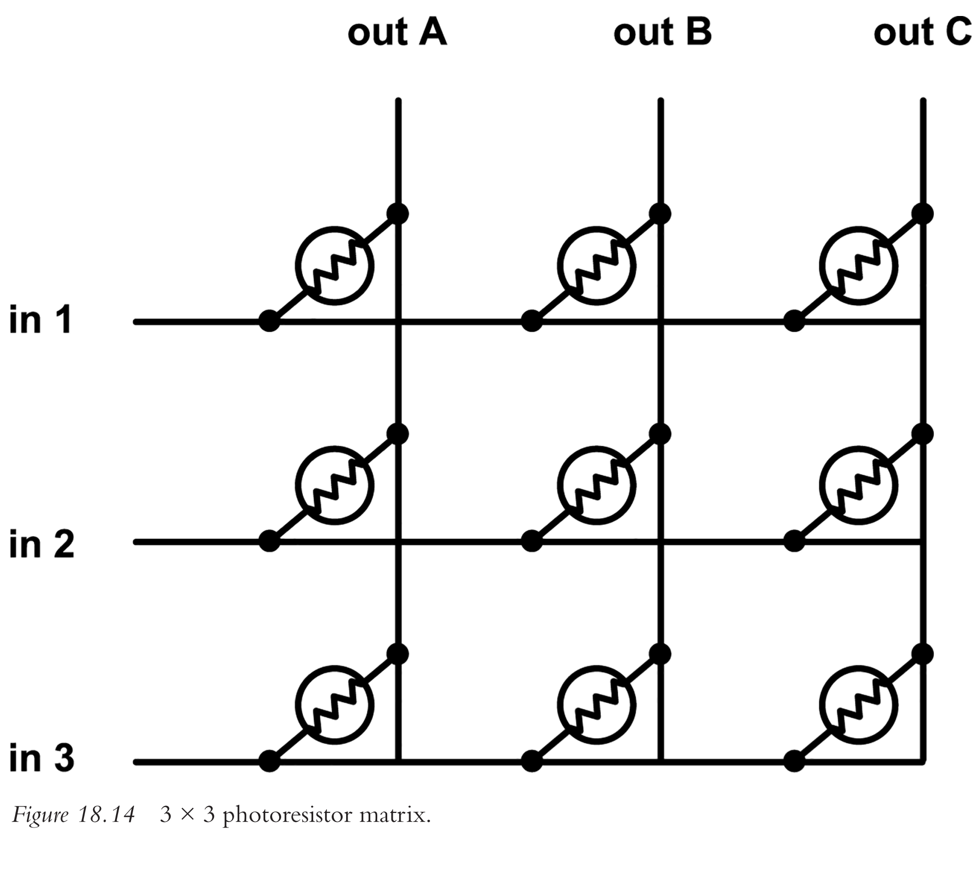


Figure 18.14 3 × 3 photoresistor matrix.

that you might also consider building them into any of your "multi-voice circuits": you can add individual level controls to each of the six channels of a hex inverter oscillator in Chapter 13, for example.

NOTE

1. Andy Keep, "Audio Y-connectors: My Secret for Instant Guerrilla Oscillators, Raw Synthesis and Dirty Cross Modulation." *Leonardo Music Journal* Vol. 17 (2007). Pp. 30–31.

CHAPTER 19

Boost and Distort

A Simple Circuit That Goes From Clean Preamp to Total Distortion

You will need:

- Something to amplify: an electric guitar works best; a contact mike, coil pickup, or file player are also useful.
- One of your oscillator circuits.
- A breadboard.
- A CD4049 (or CD4069) CMOS Hex Inverter.
- Assorted resistors, capacitors, and pots.
- Some solid hookup wire.
- An amplifier.
- Assorted jacks and plugs, to match your amplifier and sound source.
- A 9-volt battery and connector.
- Hand tools.

In addition to turning sounds on and off, as we did in Chapter 17, there are many occasions when we want to make something LOUDER (Second Law of the Avant-Garde). Loudness comes in different flavors, and a little experimenting with the CMOS CD4049 Hex Inverter will demonstrate several of them. This is yet another example of a digital logic chip being "misused" for analog purposes.

The chip's internal configuration and pinout are shown in Figure 19.1. It has two more pins than our earlier chips: 8 on a side, 16 in total, versus 14. The 4049 is a rare exception to the general rule of diagonal corner pins for power hookup in CMOS chips: although the ground connects to pin 8 as expected, + volts connects to pin 1, rather than the anticipated pin 16. The "NC" by pins 13 and 16 indicates that they make "no connection" to any internal circuitry. Why? I have no idea. Note also that the 4049 inverters face the opposite direction than those in the hex Schmitt Trigger variant of an inverter. IMPORTANT: do not substitute the hex Schmitt Trigger (74C14/40106/4584) for the 4049 in the examples in this chapter—the 74C14 *is* an inverter, but it has a different internal circuit design that won't work properly in these configurations (the 4049 omits the Schmittiness essential to making our oscillators snappy but is incompatible with the more analogish designs in this chapter).

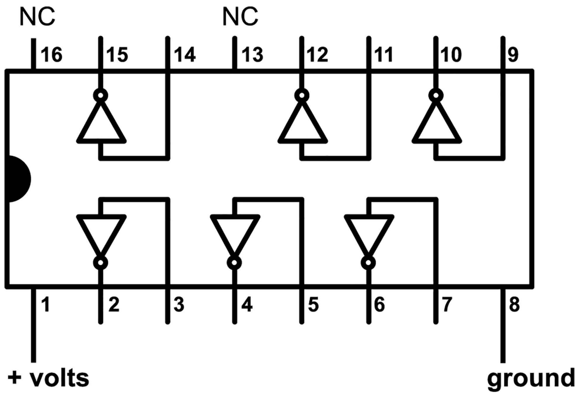


Figure 19.1
CD4049 Hex Inverter pinout.

At the head of this chapter, I've listed another chip, the 4069, as an alternative to the 4049. It has the identical pinout, voltage requirements, etc. In most cases the two parts can be substituted for each other with carefree impunity. However, because they differ slightly in their internal circuit topology, periodically some silicon sage makes the argument that one of them is preferred over the other for particular audio designs. I've had similar results with both, so while I specify the following circuits as employing the 4049, you should feel free to experiment with either chip.

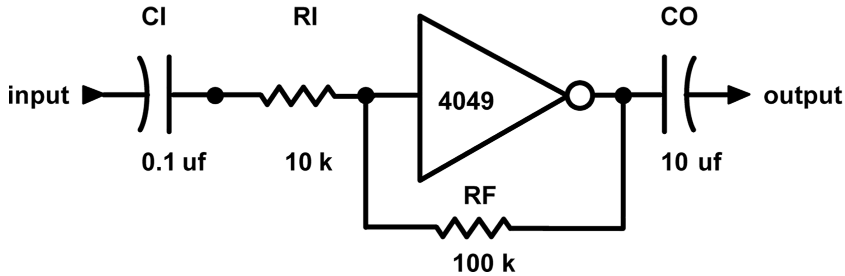
PREAMPLIFIER

Hook up the circuit shown in Figure 19.2. This is a general-purpose preamplifier circuit, useful for boosting low-level sound sources, such as microphones, contact mikes, guitar pickups, and coils, to the line-level signal strength typical of computer audio, flash recorders, etc. After preamplification these signals can be sent to powered speakers, such as those used for computers or phones, or mixed with line-level sources using simple, passive mixers (see Chapter 18). A *preamplifier* is not the same as a *power amplifier*—this circuit will not drive a loudspeaker effectively—for that you need another kind of design, discussed in Chapter 24.

Our preamp circuit has five components:

1. The CMOS inverter stage. As with our oscillator circuits, the six sections of the 4049 chip are interchangeable.
2. The input resistor, RI, generally around 10 kOhms.
3. The feedback resistor, RF, generally larger than RI (see the following).
4. The input capacitor, CI, generally around 0.1 uf, but the value is not critical.
5. The output capacitor, CO, generally around 10 uf, but the value is not critical.

Your guitar (or other sound source) connects to the jack at the *right-hand* side of the breadboard as shown while the output emerges from the *left* jack—a much needed Semitic twist on the left-to-right orthodoxy we've been observing in our designs so far. But pay attention or you'll fall back on your left-to-right habits



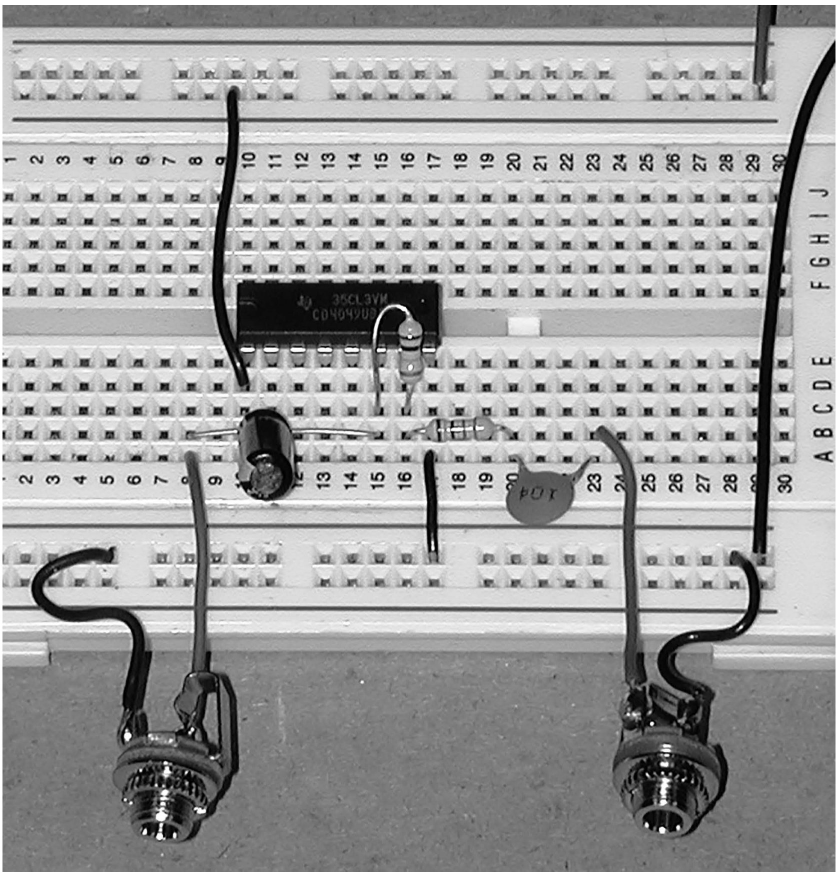


Figure 19.2 A basic preamplifier, input at right jack, output at left.

The values of CI and CO are not critical, but when using electrolytic capacitors, please observe the polarity shown in the schematics. RI can usually be set at 10 kOhm. The only real thought goes into selecting RF. The gain—how much the circuit amplifies the incoming signal—is determined by the ratio of RF/RI. So, if RI is 10 kOhms and you use a 100 kOhm resistor, the gain will be 10, which means that any signal you plug into the circuit comes out 10 times larger. If RI = 10 kOhms and RF = 10 mOhms, the gain is 1,000, which makes your input much, MUCH louder.

An aside about loudness: a gain of 10 means the voltage amplitude of the signal is 10 times larger than before, but that doesn't mean that it *sounds* 10 times louder. As mentioned previously, our ears have a logarithmic, rather than linear, response to loudness. A gain of 10 translates into a change of 20 decibels (dB), the unit of measurement used for loudness. A change of 6 dB is perceived as a doubling of loudness, roughly. So a gain of 10 equals an increase of 20 dB, which would sound about three times louder. A gain of 100 equals a 40 dB boost, or about seven times louder. I suggest that you don't obsess too much about the math but evaluate gain by listening.

By substituting a large pot (i.e., 1 mOhm) for the fixed feedback resistor RF, we can vary the boost of the circuit, as shown in Figure 19.3. For a typical preamp (as you might use for a contact mike or coil), you may wish to wire up a 10 k resistor in series

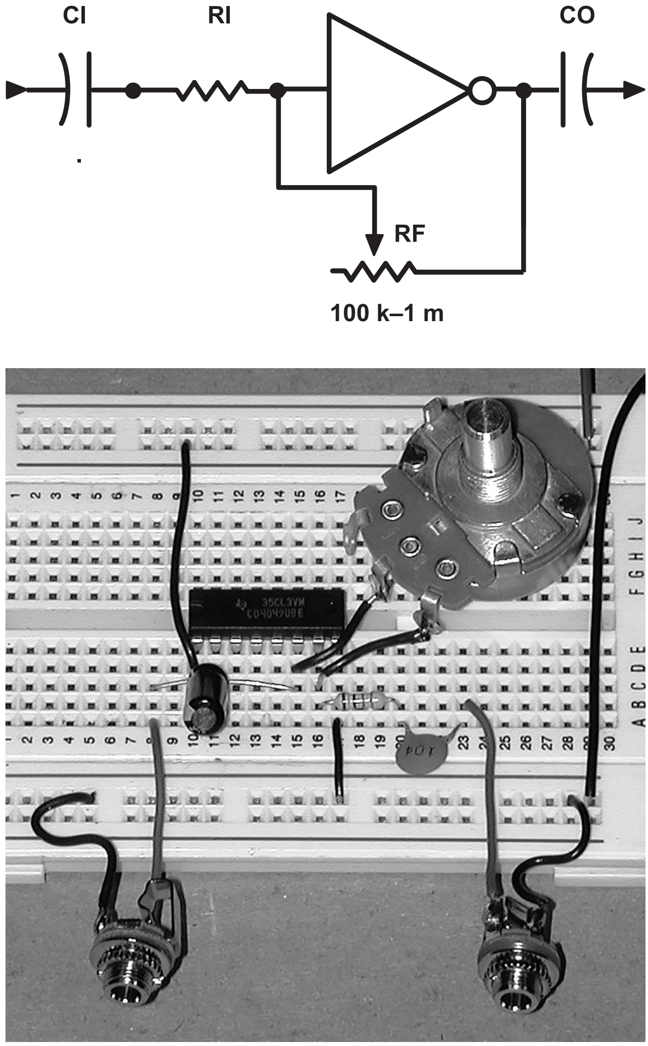


Figure 19.3 Preamplifier with variable gain.

with a 1 mOhm pot: this lets you adjust the amplification smoothly from unity gain (10 k/10 k = 1/1) when the pot is at one extreme of rotation to a gain of just over 100 (output = (1 m + 10 k)/10 k = 101/1). Reduce the size of the pot to 100 kOhm (or increase the size of RI) if you are getting too much gain.

Besides the different power supply connections, the unused pins and the "backwards" orientation of the inverter stages, this circuit uses a few more parts than anything we've built so far. It might seem confusing at first, and with each additional component the chance of making a hookup mistake increases. It may take a few tries before you get the circuit working properly. Remember that the most important detail of this design is the choice of the feedback resistor, RF. The input and output capacitors (CI and CO) block the DC voltage present in the circuit from the reaching whatever you're plugging into—as with the electret mike circuits in Chapter 11 and our triangle wave output in Chapter 15. Don't think about them too much now, just put them in. They are necessary for the stability of the circuit but usually don't affect the sound much.

TONE CONTROL

You may notice some noise or high-frequency oscillation at high gain. You can minimize these artifacts by putting a *very* small feedback capacitor (CF) in parallel to the feedback resistor as shown in Figure 19.4. Try values in the range of 10–100 pf (picofarad)—the "small" 0.1 uf capacitors we have been using in our oscillators are way too big.

Beyond getting rid of noise or unwanted oscillation, this feedback capacitor sets the upper limit of the frequency response of the circuit. By substituting slightly larger capacitors, you can transform the preamp into a simple low-pass filter tone

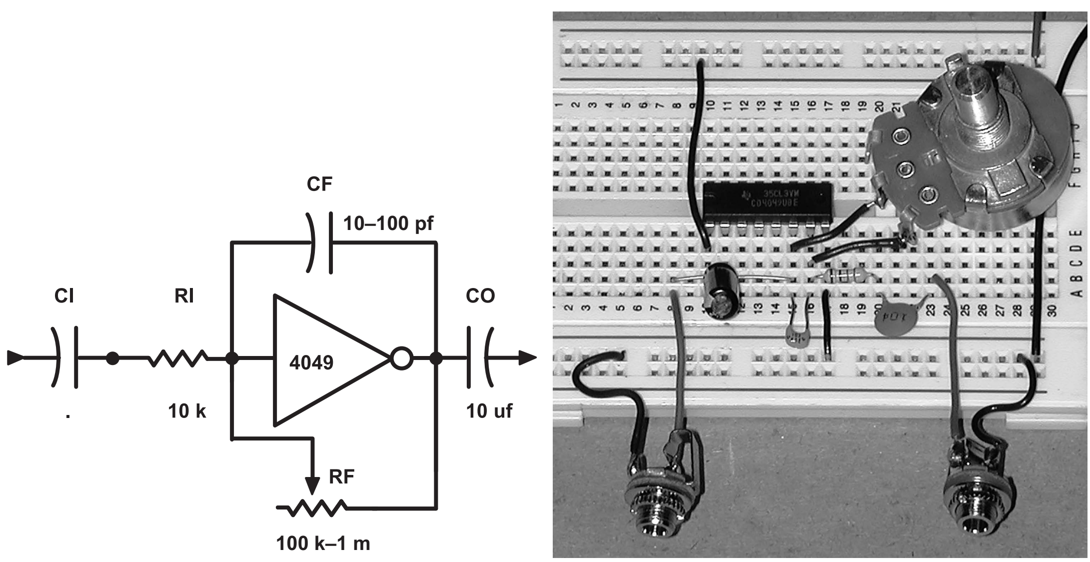


Figure 19.4 Adding a feedback capacitor.

control—one that blocks all frequencies above a certain pitch while letting those that are lower pass through.

Connect a sound source to the circuit and adjust it for relatively low gain so as not to distort the listening amplifier. Try different values for the feedback capacitors: 10 pf, 100 pf, 0.001 uf, 0.01 uf, and finally 0.1 uf. As you increase the size of the capacitor, you should notice more treble rolling off and the sound getting bassier. By the time you reach 0.1 uf, the circuit will probably be cutting off *all* audible frequencies; you'll hear almost nothing and might think the circuit is mis-wired. You will probably need to patch the preamp into a better amp and bigger speaker than the mini-amp we've been using if you want to hear the detail of these changes and the increase in bass.

Given the extremely bright high end of our square wave oscillators, some high-frequency roll-off can be welcome now and then, as we demonstrated with our filter add-ons in Chapter 15 (Figures 15.11 and 15.12). Patch any of your oscillators through a preamp set for unity gain (RI = 10 k, RF = 10 k) and try the aforementioned capacitor substitutions. The waveform should mellow out into something more like a triangle wave or sine wave, bringing relief to canine and human listeners alike.

Unlike the low-pass filter on a synthesizer, the high-frequency equalizer on a mixer, or even the simple treble control on a home stereo, the amount of roll-off cannot be continuously adjusted in this design. This circuit is best suited for fixed settings, such as softening the oscillators into pseudo-triangle waves or mellowing a distortion circuit (see the next section of this chapter on distortion). You could add a switch to select between two or three different capacitors if you want some variation (rotary switches can select among several different values). Before you sneer at switched EQ, remember that vintage Neve, API, and Pultec equalizers command stratospheric prices despite (or because of) such switching—and a few other factors, admittedly, such as brilliant design, beefy transformers, luscious knobs, rarity, etc.

But if you insist on continuously variable EQ, try the circuit in Figure 19.5, which is a variation on the low-pass filter we introduced in Chapter 15 (Figure 15.12). Instead of being tacked directly onto the output of an oscillator, a similar arrangement of a capacitor and two photoresistors is added to a stage of a 4049. The two photoresistors should be close together so they receive the same amount of light and "track" each other with the same resistance value, otherwise you will hear a change of loudness as the filter is swept. Experiment with different values for C filter—0.01 uf is a good place to start. CI and CO are needed, as in our basic preamp circuit, to block DC.

For more accurate control of this filter, you can substitute a dual potentiometer for the two photoresistors. A dual pot consists of two potentiometers controlled by a single shaft (Figure 18.6). Thanks to this mechanical coupling, they track each other very closely and eliminate the problem of trying to keep the same light level on the two photoresistors.

Remember the input and output capacitors I told you not to worry about? If you enjoy worrying, try capacitor substitutions on CI. As you *decrease* its value from the default 0.1 uf down to 0.01 uf, 0.001 uf, 100 pf, and 10 pf, you should notice that the bass frequencies diminish—we've made a simple "high-pass filter." With careful

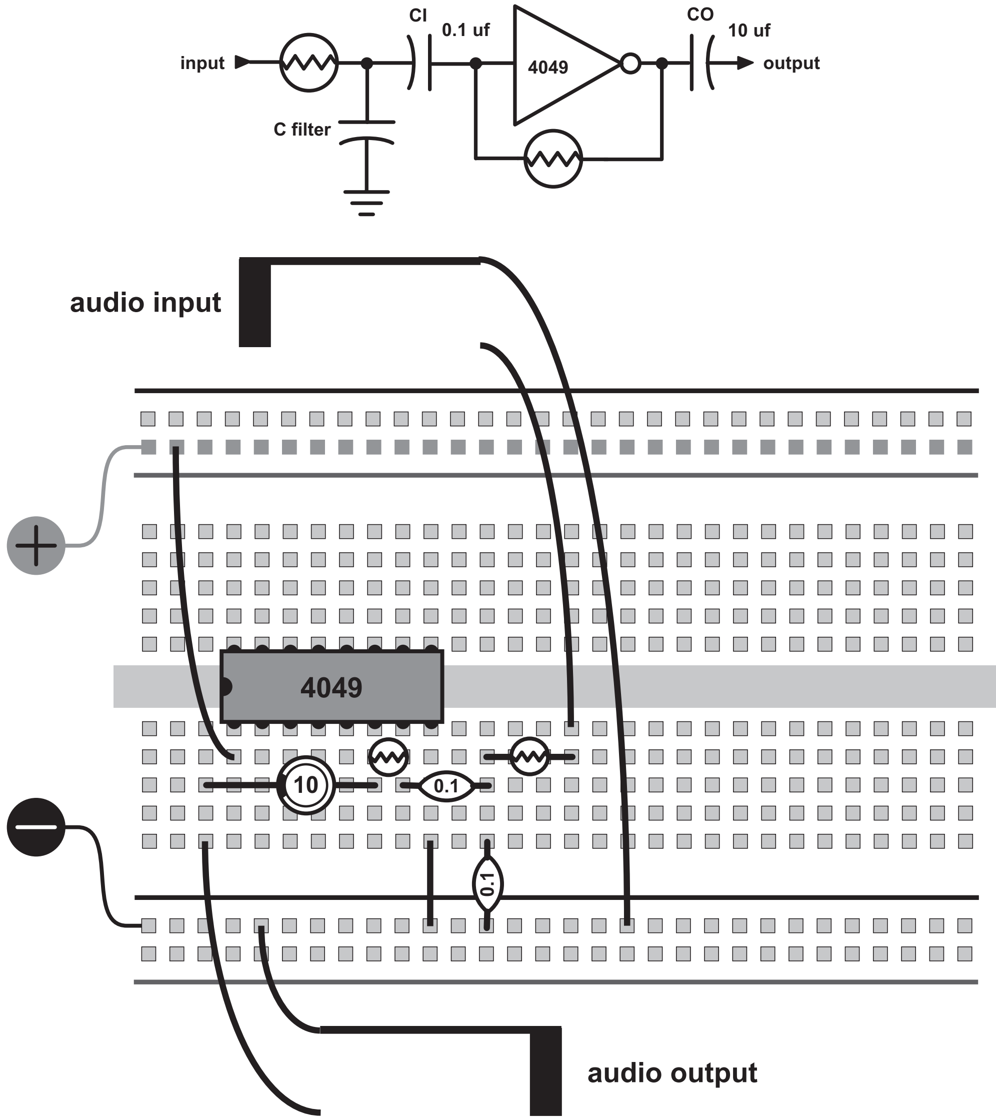


Figure 19.5 Low-pass filter.

selection of the optimum capacitor size, you can make a useful filter for rolling off low-frequency wind noise and handling noise and rumble from microphones and contact mikes.

The six inverter sections in the 4049 can be used interchangeably. You can make preamps for six contact mikes with one chip, for example. Each preamp can be wired to its own output jack for connection to an external mixer, or you can sum them together with 10 k resistors as we did with the multiple oscillators in Chapter 13. You can build them into our mixer designs from Chapter 18.

DISTORTION

Amplifier sections can be cascaded to produce greater gain: by putting two stages with 10x gain in series, you get a net gain of 100. But this cumulative amplification is not perfect, and by adding gain in series, we introduce distortion, the guitarist's friend. The circuit in Figure 19.6, based on a venerable design by Craig Anderton (the godfather of musical hacking), is simple and versatile and sounds great.

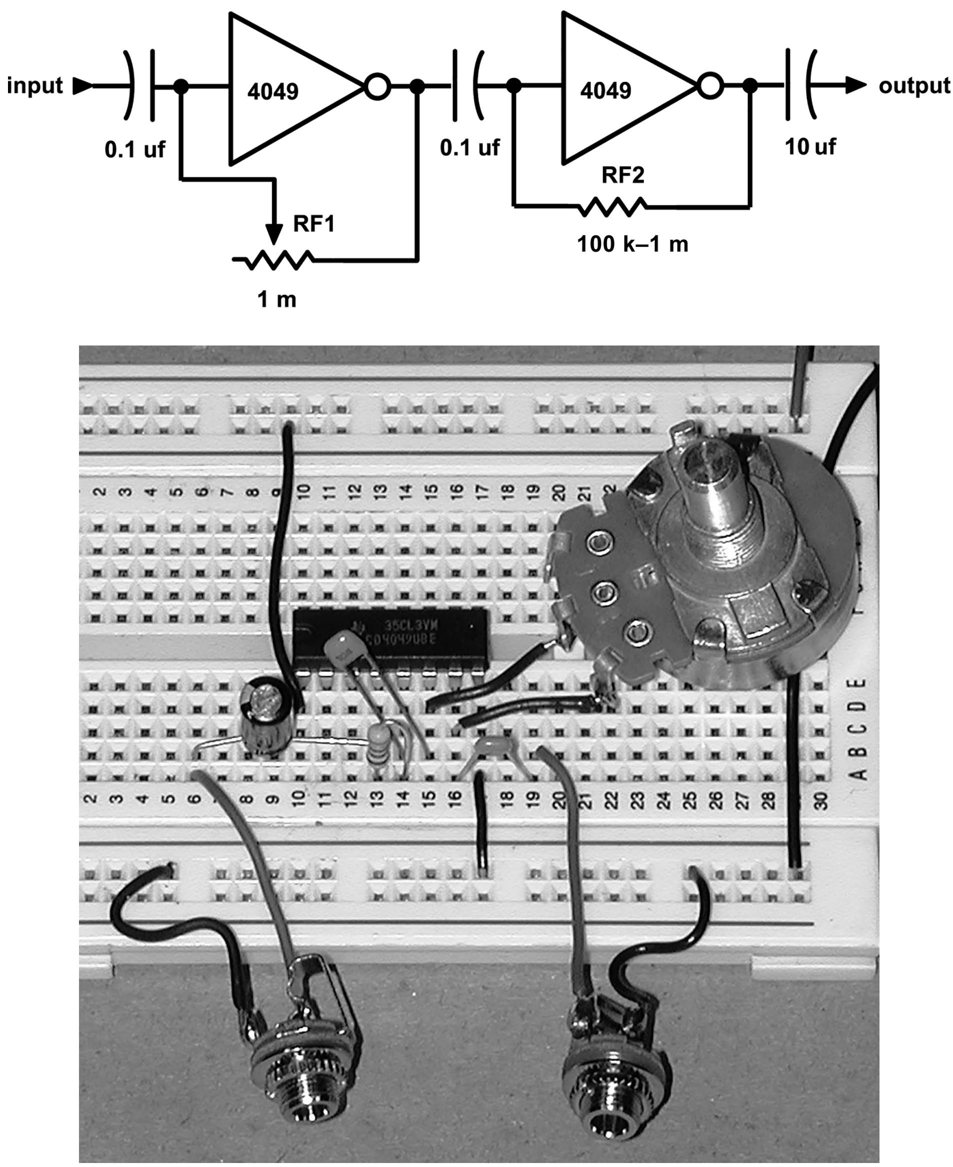


Figure 19.6 Craig Anderton's basic distortion Circuit.

Plug in a guitar (always the best instrument for evaluating distortion) and turn the pot. As the gain increases, the sound should shift from clean amplification through tube-like "overdrive" into distortion and, eventually, uncontrollable noise and oscillation. Ahh, bliss! As Robert Poss says:

Rule #23: Distortion is Truth (Poss's law).[1]

But truth and distortion come in many flavors. If you are interested in distortion, you should spend some time substituting different components until you hear your personal perfection. In particular, try:

- Various resistors, from 100 k–10 mOhm, for RF2.
- Substituting a pot for RF2 for separate control of gain at each stage.
- Adding a "clamp resistor," around 10 k–100 kOhm, from the input to the second stage to ground—this sometimes reduces noise and oscillation at high distortion/gain settings.
- Inserting 10 k input resistor between CI and the first inverter (as we had in our preamp circuits).
- Various feedback capacitors, from 10 pf to 100 pf, for CF1 and CF2, in parallel to RF1 and RF2. These values can be set with fixed values or made switch selectable, as suggested prior for our simple EQ circuit (Figure 19.7).
- Adding a stage of swept low-pass filtering as shown in Figure 19.5.
- Adding an additional gain stage, which makes the distortion more extreme (Figure 19.8).
- Following the circuit with a Schmitt Trigger Inverter (from a 74C14/40106/4584), which clips the signal into a buzzing pulse wave in the style of a classic 1960s fuzz-tone (Figure 19.9). You will probably need a resistor of about 100 kOhms between the input of the Schmitt Trigger and ground to keep the circuit from oscillating in the absence of an actual input signal. Sometimes removing the capacitor marked with the asterisk and replacing it with a straight wire also improves stability.

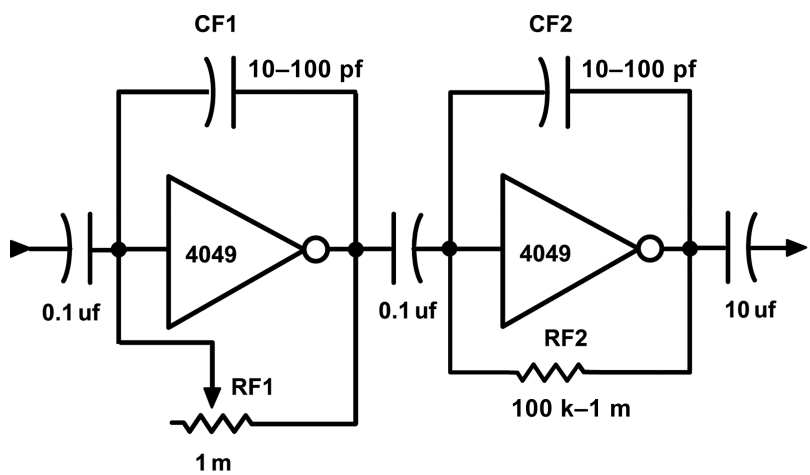


Figure 19.7 Distortion with feedback capacitors added.

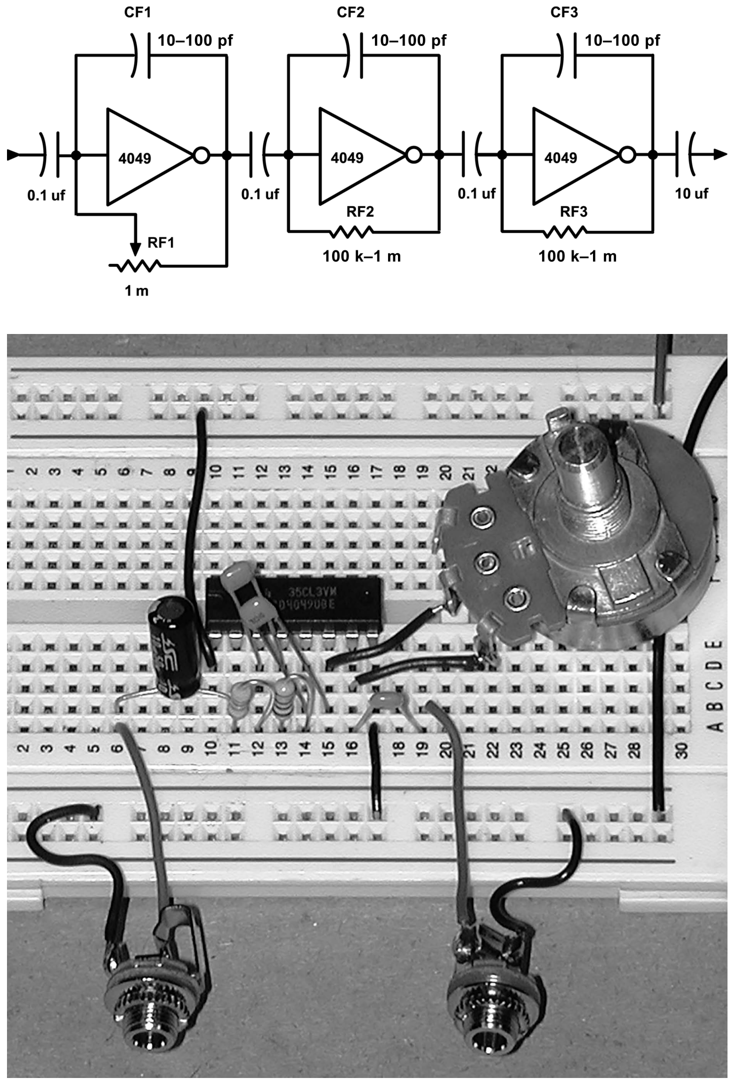


Figure 19.8 Distortion + Distortion (feedback capacitors omitted from breadboard for clarity).

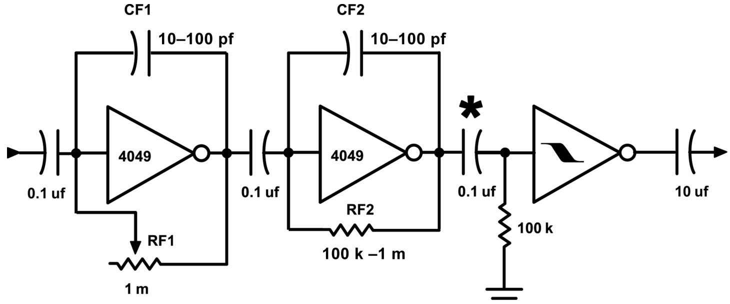


Figure 19.9 Distortion + Fuzz (feedback capacitors omitted from breadboard for clarity).

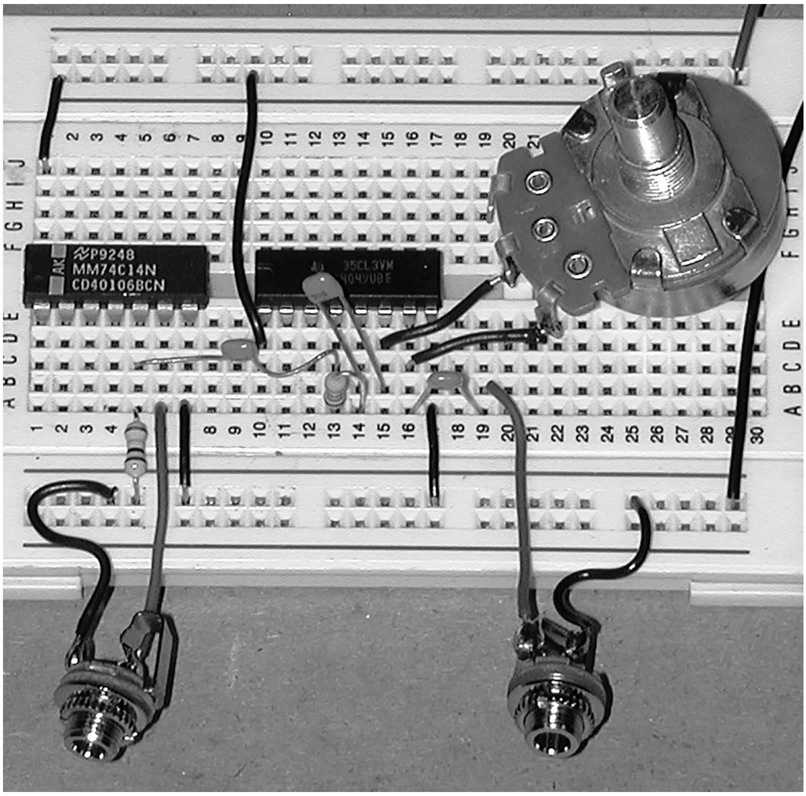


Figure 19.9 (Continued)

When running at high distortion, the output signal of these circuits reaches 9 volts peak-to-peak, like our oscillators. You may want to add an output volume pot to lower the output level independently of the amount of distortion before you plug it into a guitar amp (most commercial distortion circuits have separate controls for distortion and level). Another common feature in stomp boxes is a bypass switch that allows you to select between the processed output of the circuit and the unprocessed input. Adding this function simply requires a SPDT switch (usually a heavy-duty push-on/push-off switch that alternates closed/open on each pressing) with the NC (normally closed) terminal connected to the input to your circuit, the NO (normally open) to the circuit output, and the C (common) to the output jack's tip/hot. A toggle switch might be more useful than a pushbutton if you're operating the circuit by hand on a tabletop, rather than with your foot on the floor. Figure 19.10 shows both these features added to the basic distortion circuit, but you can include them in any of the variations discussed prior.

If you haven't started breadboarding these designs yet, let me forewarn you (and if you're already deep in it, let me reassure you): as I mentioned earlier in this chapter, these circuits—with the odd pinout and increased parts count—are more confusing

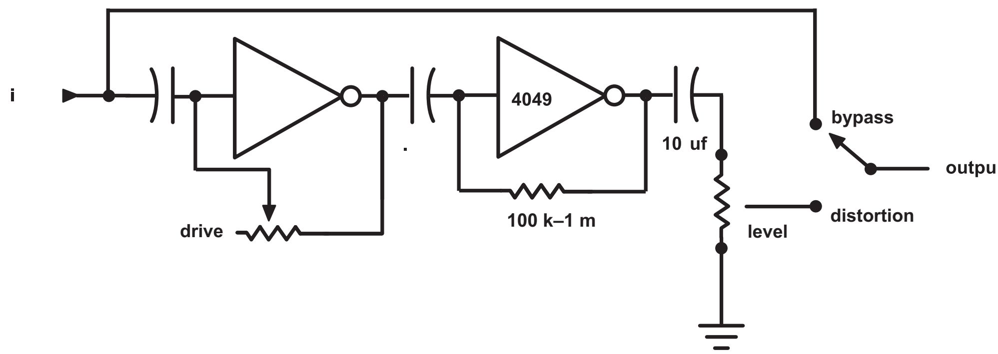


Figure 19.10 Output level and bypass switch added to basic distortion circuit.

than anything you've encountered so far in this book. Moreover, as you raise the overall gain, especially in the distortion designs, your circuit starts to behave irrationally: it's supposed to be amplifying your guitar, but it seems to be oscillating all by itself. Why? At high gains this circuit isn't only amplifying the signal you think you're sending it (the guitar), but it also boosts all manner of spurious noise and hum to the point that there is no more silence. In the battle of signal to noise, noise has the upper hand. To improve the signal's odds:

- Keep all jumper wires as short as possible—on the breadboard and between the pots and jacks and the breadboard.
- Use shielded cable to connect to the circuit from your instrument.
- Add tiny capacitors in the various feedback stages and a 0.1 uf cap between + power and ground.
- Add clamping resistors to ground at the input to each stage.
- Lower the gain of each stage.
- Build the circuit one stage at a time, testing after each addition.

It's worth persevering. As frustrating as it can feel sometimes, designing and debugging by ear are essential to building good-sounding audio circuitry. Preamplifier and distortion circuits are very useful in their own right. And if we can successfully boost ordinary analog audio to the near-square wave signals put out by our more extreme distortion circuits, we open the door to some unusual signal processing, as you will see in the next chapter.

NOTE

1. Robert Poss, "Distortion Is Truth." *Leonardo Music Journal* Vol. 8 (1998). Pp. 45–48.

CHAPTER 20

Analog to Digital Conversion, Sort Of

Modulating Other Audio With Your Circuits, Pitch Tracking, and a Sequencer

You will need:

- Something to amplify: an electric guitar is best.
- A breadboard.
- Distortion circuit from the previous chapter.
- CD4093 Quad NAND Gate Schmitt Trigger.
- CD4040 Binary Counter/Divider.
- CD4046 Phase Locked Loop.
- Hex Schmitt Trigger (74C14, CD4584, or CD40106).
- CD4017 Divide-by-Ten Counter.
- Assorted resistors, capacitors, and pots.
- Some solid hookup wire.
- Assorted jacks and plugs.
- A 9-volt battery and connector.
- An amplifier.
- Hand tools.

The preamp and distortion circuits in the previous chapter are useful on their own to boost a low-level signal or make an electric guitar sound legitimate but also in conjunction with other circuits. As I've already mentioned, an ordinary line-level audio signal (such as the output of an audio interface) measures a bit less than 1 volt peak-to-peak and fluctuates in a curvy, baroque way. The circuits we've been making with CMOS digital chips put out 9 volts peak-to-peak (if powered by a 9-volt battery), and their outputs snap between 0 and 9 volts with modernist decisiveness. They won't react to any puny 1-volt wigglings at their inputs. Until a signal pokes its head up over one-half the supply voltage (4.5 volts), it always feels like 0, no matter how much it jumps around. But with enough gain and distortion, any analog audio starts to look like one of our digital square waves. The more it resembles a digital signal, the easier it is to fool our CMOS chips into accepting it as kith and kin. This deception lets us interface analog sounds from the real world to our digital circuits for some unusual signal processing.

THE FUZZY DICER

The Distortion + Fuzz circuit in the previous chapter (Figure 19.9) was the first step in this conversion of snaky analog waveforms into crisp digital signals: we boosted our guitar high enough that it could trigger a 74C14 to produce pulse waves. The next step is the circuit shown in Figure 20.1, which is a variation on the gated oscillator in Chapter 15. Instead of using one oscillator to modulate another, however, here the quasi-square wave output of the distortion circuit is connected to the control input of the NAND oscillator. When the oscillator runs at low speeds (1–10 uf timing

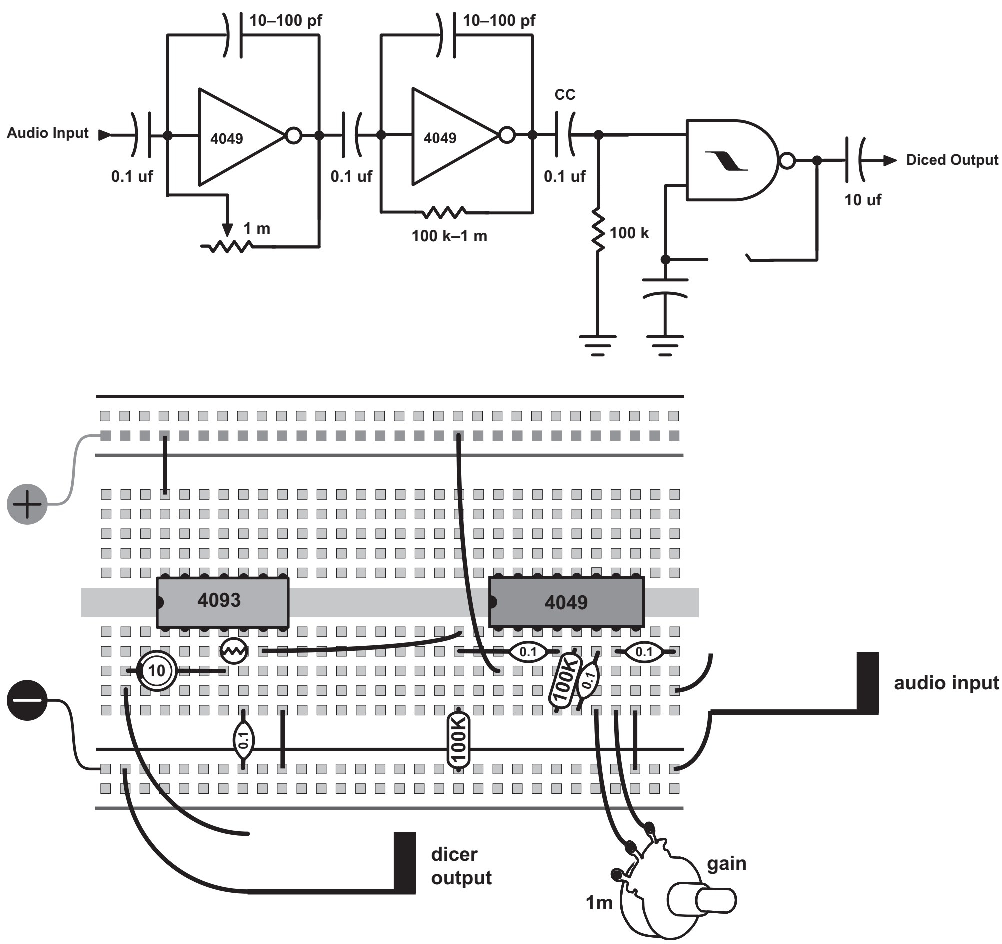


Figure 20.1 The Fuzzy Dicer.

capacitor), it effectively "chops" the output of the distortion circuit on and off—like the photocell gate in Chapter 17 but with a much sharper edge and a complete muting in the off state. At higher speeds (0.01–0.1 uf capacitor), the oscillator interacts with the distorted signal to create sounds reminiscent of a ring modulator or (somewhat inexplicably) a wah-wah pedal.

Try your favorite version of the distortion circuit in Chapter 19. Don't include any output volume control or bypass switch between the distortion section and the NAND gate; if you want, you can put them after the oscillator instead. The circuit may be more stable without the coupling capacitor CC—try both ways. A pull-down clamping resistor between the 4093 input and ground is usually needed to mute the circuit when you are not sending it an input signal (i.e., not playing the guitar). If it seems insensitive, only triggering on very loud signals, you may need to increase the gain of your distortion circuit—try the triple-inverter version shown in Figure 19.8 or the fuzz circuit in Figure 19.9.

If it's *too* sensitive, and prone to fits of high-frequency squealing, try putting larger capacitors in the feedback loops of any or all of the 4049 stages of the distortion circuit. This will roll off the higher frequencies, remove excess noise, and send the subsequent circuits a signal that emphasizes the fundamental pitch—the output of your 4049 might sound unacceptably muffled for use as stand-alone distortion but will do a better job of feeding the NAND gate with a stable signal to track. This circuit may take some tweaking, but the effort is worth it—and you can't buy one anywhere (yet).

You can modulate a second NAND gate with the output of the first, as we did in Chapter 15 (Figure 15.4) with the introduction of cascaded oscillators. If you use the same value capacitor for both stages, the cool flangey effect interacts with whatever you've plugged into the circuit. You'll need to add the coupling capacitor and clamping resistor shown in Figure 20.2. You can hear this circuit on Robert Poss's audio track on the website.

You can also use the NAND gate as a kind of digital mixer. Breadboard two distortions circuits (there are enough inverters in a single 4049) and send one distortion output into each input of a 4093 gate, as shown in Figure 20.3. Now instead of an oscillator modulating the external signal, two external signals cross-modulate each other in that vaguely ring modulator way. Try a guitar in one channel and a recording or synth in the other. Plug the same signal into both inputs. Experiment.

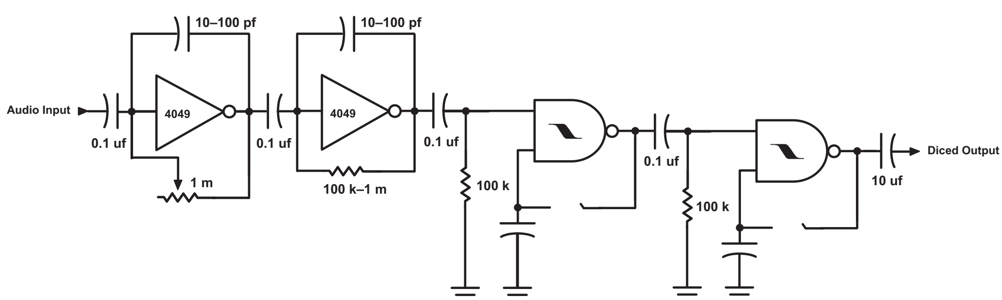


Figure 20.2 Super Fuzzy Dicer.

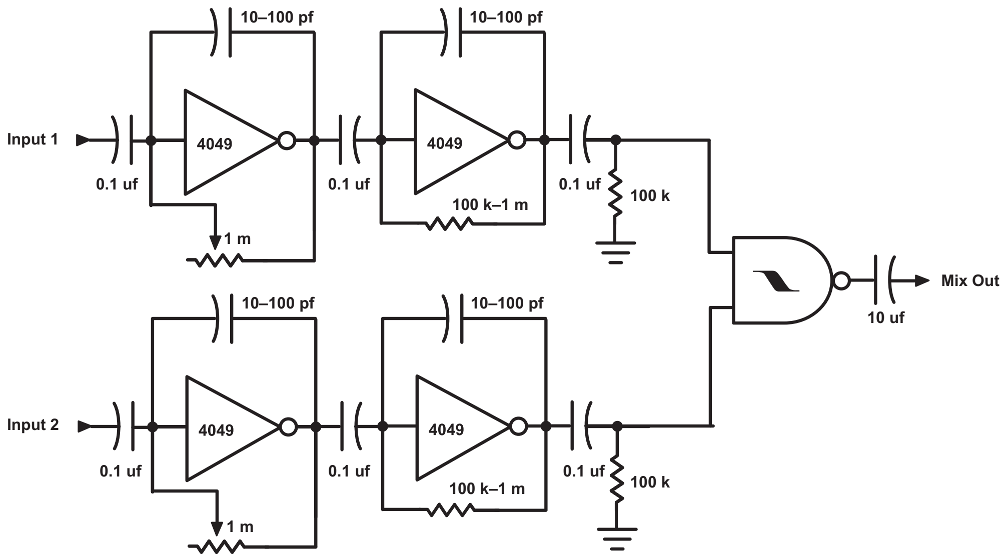


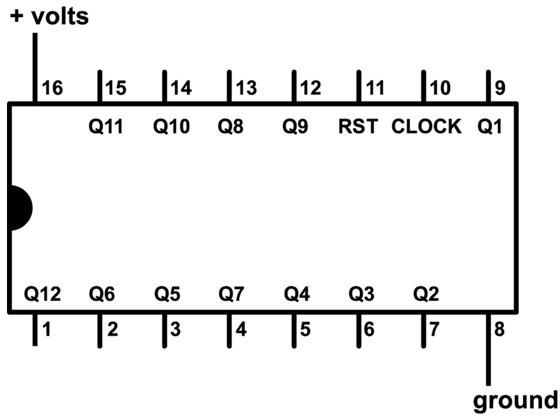
Figure 20.3 Using a NAND gate as a "digital mixer."

DIVIDERS—AN INTRODUCTION

We've been squeezing the maximum number of applications out of the minimum number of chips to save you money, to avoid clutter, and as a classic minimalist exercise in maximizing permutational strategies within a finite set of materials. As loath as I am to add more ICs to your collection, some are just too cool to pass up. The musical applications of digital logic circuits go well beyond simple oscillators and extreme distortion. Once you accept that any audio square wave can be regarded as a simple alternation between two binary numbers (0 and 1), the logical and arithmetic functions on which all digital calculations are based can be seen as potential sound transformations. Welcome to CMOS Boolean signal processing! For example, a chip that performs division, such as the CD4040 12-Stage Binary Divider, can be used to generate several harmonically related pitches from a single master oscillator. Figure 20.4 shows its pinout and internal configuration.

Breadboard the circuit shown in Figure 20.5. Note that the 4040 has 16 pins like the 4049 in the previous chapter, but the power connections have a similar diagonal arrangement to the 14-pin chips we've used so far: ground connects to pin 8, +9 volts to pin 16. You also need to connect ground to pin 11, designated "rst" (for Reset), or the chip will not run. A master oscillator, built from one section of a 74C14 or 4093, is connected to pin 10, the clock input of the divider chip (4040). Don't forget to hook up +9 volts and ground to the 74C14/4093 as well. The 4040 has 12 cascaded stages, each of which divides the frequency by 2; outputs are provided for the clock signal divided by 2, 4, 8, 16, etc., all the way to 4096.

Figure 20.4 CD4040 12-Stage Binary Divider pinout.



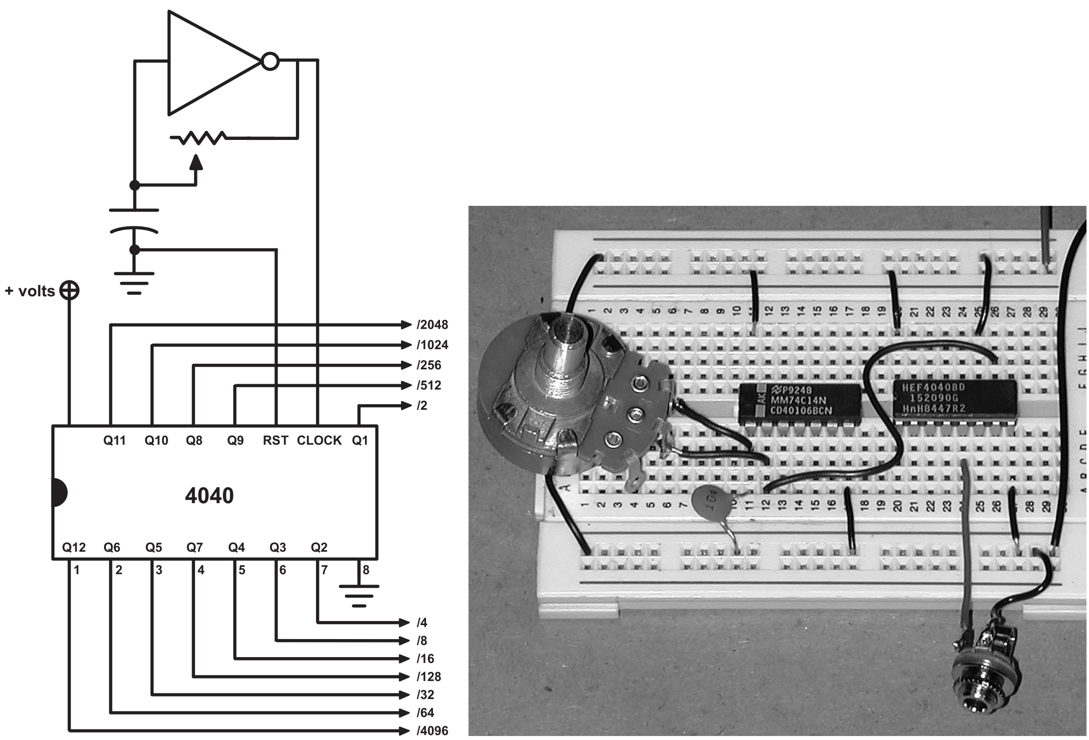


Figure 20.5 Divided oscillator.

Connect the shield of the audio jack to the ground bus and attach a few inches of solid hookup wire to the hot connection. Connect this jumper wire first to the output of the clock oscillator (master output) by inserting it into the vertical bus at pin 10 on the 4040 and tune the clock to a high-frequency audio pitch. Now listen to Output Q1 (clock frequency/2) by pushing the wire into the row at pin 9 and note that it sounds an octave lower than the clock. Output Q2 sounds an octave below that (clock/4), Output Q3 an octave below that (clock/8), etc.

You can use switches to select among these different subharmonics. The switches can be anything: momentary pushbuttons, toggle switches, rotary switches, home-made tilt switches, or just some wire jumpers on the breadboard. Because of the thick texture and low frequencies of this circuit, you may want to listen to it over a larger loudspeaker than that of a tiny test amplifier. If you want to mix multiple outputs to build up a rich waveform, make sure you send each output through a resistor before tying together (as we did with the multiple oscillators in Chapter 13). Because of the simple harmonic relationship of the octaves, you will note that the differences in mixes are subtle: a slight shift in overtone balance, rather than an impression of distinct voices being added.

If you slow the master oscillator down to the rate of a tempo, rather than a pitch, the various outputs become subdivisions of the beat—good for setting up nested rhythmic patterns. You can connect the various outputs of this divider circuit to multiple LED/photoresistor gates (like those in Chapter 17) to chop multiple sound sources in rhythmic patterns—Hacking Dub!

Disconnect the oscillator circuit from the clock input (pin 10) and connect a short piece of solid wire sticking up into the air from the breadboard. Sometimes your body carries enough of an electrical charge that if you touch the end of the wire, the noise of your flesh will trigger the divider. Listen to the different divisor outputs as you experiment with brushing and squeezing the wire. Sometimes it helps to connect large resistors (100 kOhm–1 mOhm) between the clock input pin (pin 10) and ground and/or +9 volts to stabilize the circuit when you are not touching it. An excellent ghost detector.

THE LOW RIDER

Substitute the output of a distortion circuit for the oscillator driving the clock input of the 4040 (pin 10) and the divisor outputs become subharmonics of whatever you play into it—a "Rocktave Box," to use the tacky industry parlance, or "Low Rider," if you prefer (Figure 20.6).

You can use switches to select different subharmonics (a tilt switch on the headstock of your Fender Mustang?) or set a fixed mix. Divided down far enough, an E chord becomes a rhythmic pattern, which can be used to blink an LED to control a photoresistor to gate on and off . . . whatever.

As with the Fuzzy Dicer, you may need to experiment with the coupling capacitor CC and the pull-down resistor Rd; adding the Schmitt Trigger buffer as we suggested for the Fuzzy Dicer will likely make the pitch tracking more stable. Likewise, larger feedback capacitors in the distortion stages will strengthen the fundamental pitch of the incoming signal and minimize the circuit's tendency to fly off into the unknown.

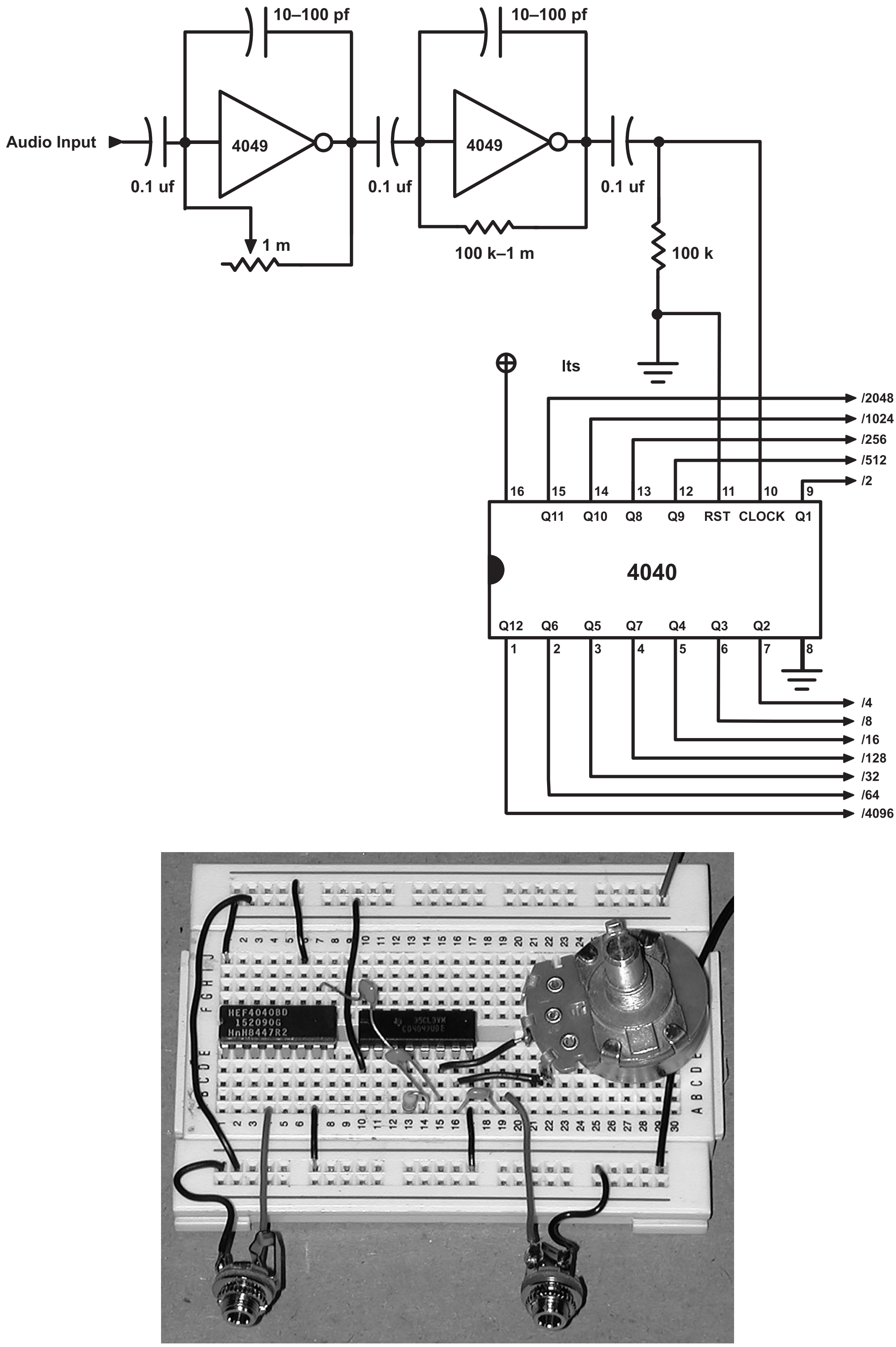


Figure 20.6 The Low Rider.

POOLS OF PHASE LOCKED LOOPS

Phase locked loops are versatile pitch tracking devices. They can be used to detect specific frequencies, convert a complex signal into a simple square wave, multiply or divide pitches by a factor, and more. The first widespread consumer application of PLLs (as they are commonly abbreviated) was in touch-tone telephone networks, converting the merry melodies generated by punching phone buttons back into numbers to route your call.

Figure 20.7 shows the pinout and internal configuration of the 4046, with all its inscrutable acronyms and abbreviations. The two key components of a PLL are:

- A voltage-controlled oscillator (VCO). Similar to our earlier designs, the VCO is a square wave oscillator whose overall range is set by a capacitor, connected between pins 6 and 7 (rather than going from one pin to ground). But instead of controlling the frequency with a resistor, a *voltage* is applied to pin 9 (VCO in). Voltage control greatly increases the versatility of an oscillator, as we shall see—it is the functional underpinning of traditional analog synthesizers. The output signal of the VCO appears at pin 4.
- A phase comparator. This module compares the phase of an external signal (the pitch we want to track) with that of the internal VCO and generates an error voltage proportional to the difference between the two. When this error voltage is connected to the VCO, the resulting control feedback loop forces the VCO to match pitch of the external signal. The 4046 contains two different styles of phase detector, each suited to a different kind of input signal (we'll elaborate later): both track the signal applied to pin 14 (signal in); the error voltage from Phase Comparator 1 appears at pin 2 and the voltage from Phase Comparator 2 is at pin 13.

Power hookups are "normal": ground to pin 8, + voltage to pin 16. "Inhibit" (pin 5) must be connected to ground for the circuit to run (connecting to + voltage shuts off the output).

PITCH TRACKING

Tracking the pitch of a real-world sound—such as a guitar, music file, or ambient recording—with a square wave oscillator may sound like a stupid idea. But the very fallibility of the PLL circuit produces wonderful artifacts: by adjusting a few parameters,

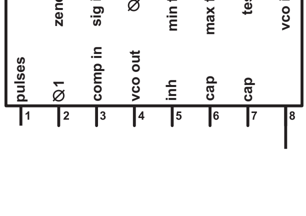


Figure 20.7
CD4046 pinout.

we can go from extreme distortion, to watery burbling, to swooping glissandi chasing our original sound. The first step is to condition our input, as we did to feed the circuits earlier in this chapter. We preamplify and distort the incoming signal until it resembles a square wave: connect the output of a basic distortion circuit to pin 14 of the PLL as shown in Figure 20.8; a 100 kOhm clamping resistor to ground keeps

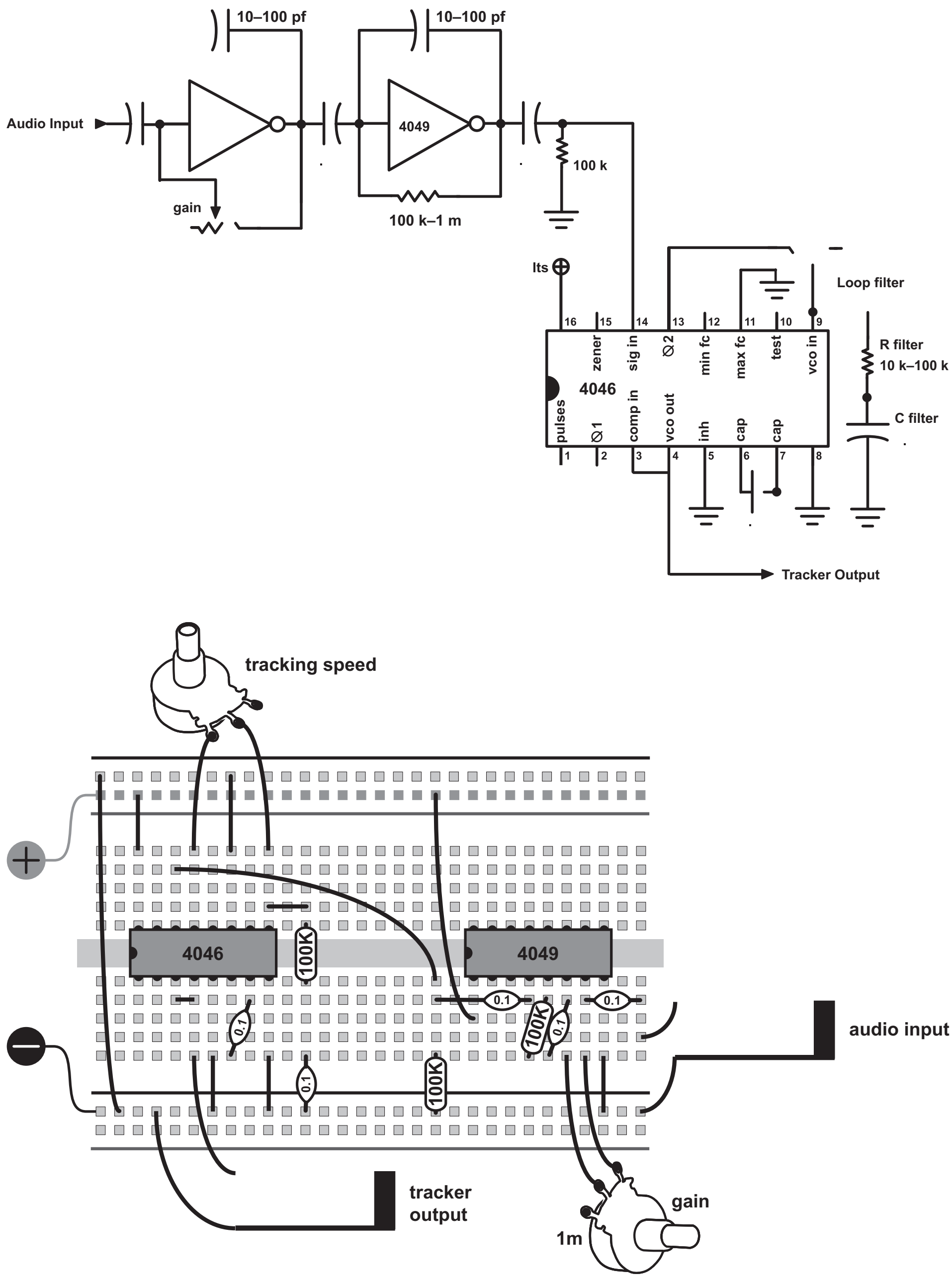


Figure 20.8 4046 pitch follower.

the circuit from flying off when no signal is present. For the VCO to track pitch, we connect VCO out (pin 4) to comp in (pin 3). VCO out is also the pin we listen to by connecting pin 4 to our amplifier.

The critical variables are:

- VCO cap (pins 6 and 7): the capacitor we insert between pins 6 and 7 determines the pitch range of the VCO. Start with 0.1 uf.
- Max freq (pin 11): a resistor between pin 11 and ground sets the highest frequency the VCO will reach. Start by linking it to ground with a wire jumper, which sets the highest pitch to the maximum given the VCO capacitor you've chosen; later you can substitute progressively larger resistors to lower the ceiling frequency.
- Min freq (pin 12): similarly, a resistor between pin 12 and ground sets the *lowest* frequency. Start by leaving it unconnected (air is a very large resistor indeed), which sets the lowest pitch to the minimum possible with the VCO capacitor. Later you can substitute progressively smaller resistors, starting with a value around 1 mOhm, to raise the minimum frequency.
- Choose which phase comparator you will use: for general audio tracking, Phase Comparator 1 (pin 13) is usually more effective, but experiment with Phase Comparator 2 (pin 2) as well.
- Loop filter: the phase comparator output you've chosen connects to the VCO through a loop filter, which determines how quickly the VCO tracks the input pitch. As shown in the figure, connect one ear of a 1 mOhm pot to your phase comparator output (here pin 13, for phase comparator 2) and connect the nose to VCO in (pin 9). Now connect the other ear through a resistor (R filter) and capacitor (C filter) to ground—good starting values are 47 kOhm and 0.1 uf.

Connect a signal to the distortion stage input (a guitar or sound file). Raise the gain of the distortion circuit until you begin to get a response at the VCO output. You should notice that adjusting the loop filter pot affects the speed of the tracking from so fast at one extreme as to sound like noise and gets progressively smoother as you rotate it. Substituting other values for R filter and C filter will also change the circuit's behavior. While keeping the same capacitor, increasing R filter to 100 kOhm will make the tracking more accurate but slower to move to a new frequency, with swooping glissandi; reducing R filter to 10 kOhm makes everything faster and noisier (a great way to turn any music file into delirious noise). You can replace R filter with a second potentiometer of an appropriate size (best with a limiting resistor in series) if you want maximum flexibility. Increasing C filter similarly slows things down and smooths them out while decreasing it dirties stuff up.

Once you've got the basic circuit running, you can:

- Try the other phase comparator.
- Try a larger or smaller VCO capacitor to shift the pitch range.
- Substitute progressively larger resistors between pin 11 (max freq) and ground to limit the highest frequency.
- Substitute progressively smaller resistors between pin 12 (min freq) and ground to limit the lowest frequency.

- Try listening to pin 1 (phase pulses) and the unused phase comparator output—these signals reflect the weird interaction between the input signal and the VCO as the VCO tries to lock on the input. You might like it.
- Breadboard a simple 74C14 oscillator running at a low frequency (2.2 uf capacitor and 1 mOhm pot). Connect its output to pin 5 of the 4046 (Inhibit) in lieu of the jumper to ground. Adjust the rate of the 74C14 to gate the 4046 on and off; at higher frequencies you get nice modulation effects.
- A simple low-pass filter (such as those in Chapter 19) can be added to the output if the square wave becomes too harsh.
- Instead of using a distortion circuit to drive the PLL directly, connect one of the divisor outputs of the Low Rider. Now the tracking will be transposed down by the interval you select.
- If you insert a 4040 between the VCO out and comparator in, you can transpose the tracking VCO *up*, instead of down. Connect pin 4 on the 4046 (VCO out) to pin 10 of the 4040 (clock input). Then connect any divisor output of the 4040 to pin 3 on the 4046 (comparator in). Linking 4040 pin 9 (divide by 2) to the 4046, for example, will cause the PLL to track an octave above the actual input signal. Don't forget to hook up power to the 4040 (pins 8 and 16) and tie Reset (pin 11) to ground.

For any these signal processors, you might want to add the bypass switch and output volume control discussed at the end of the previous chapter. If you use a single distortion circuit as the front end for all three of them, you may want to build in a simple mixer that lets you blend any combination of the clean audio input, the output of the distortion stage, the Fuzzy Dicer output, any divisors of the Low Rider, and the PLL—refer to Chapter 18 for mixer designs.

OTHER APPLICATIONS OF EXTREME AMPLIFICATION

These audio processors get us into a very wooly area of circuit conglomeration never anticipated by the original designers of these chips—trial-and-error is the best working method. This distortion-based pseudo-analog-to-digital conversion can be coupled with many other chips in the CMOS family (time to start downloading those PDFs or order a data book), some of which will yield exquisite signal transformations.

LOVE PARADE

The 4046 PLL's VCO has applications beyond its bumbling attempts at whistling along with your work. For the next project, we'll introduce one more chip. The CD4017 is a counter/divider chip, like the 4040. But where the 4040 performs *binary* division (each output is half the frequency of the previous one), the 4017 is a *decade* counter: it counts to 10, over and over, like a counselor ticking off campers on her fingers (Figure 20.9).

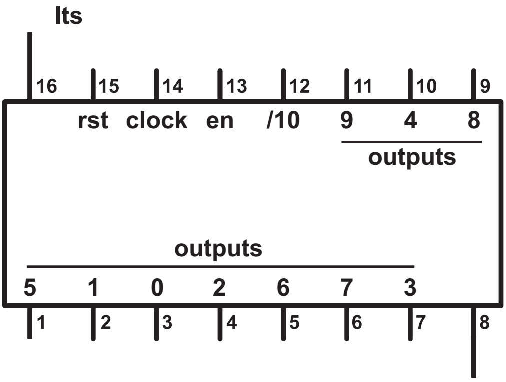


Figure 20.9
CD4017 pinout.

To familiarize yourself with the 4017, breadboard the circuit shown in Figure 20.10. The power hooks up to pin 8 (ground) and 16 (+9 volts). A 74C14 is configured as a simple clock, connected to pin 14 (clock) of the 4017. Pin 15 (Reset) and 13 (Enable) are both tied to ground through 100 kOhm resistors. Each of the outputs, 0–9, passes through a 2.2 kOhm resistor to the + side of an LED; the − side of each LED goes to ground. When you connect the battery, the LEDs should go on, one at a time, cycling from 0 to 9 before starting over (digital engineers love to start counting at 0, instead of 1—get used to it). This chip is the heart of a lot of pointless, if captivating, "LED Chaser" kits—remember this next time you're stuck for a gift.

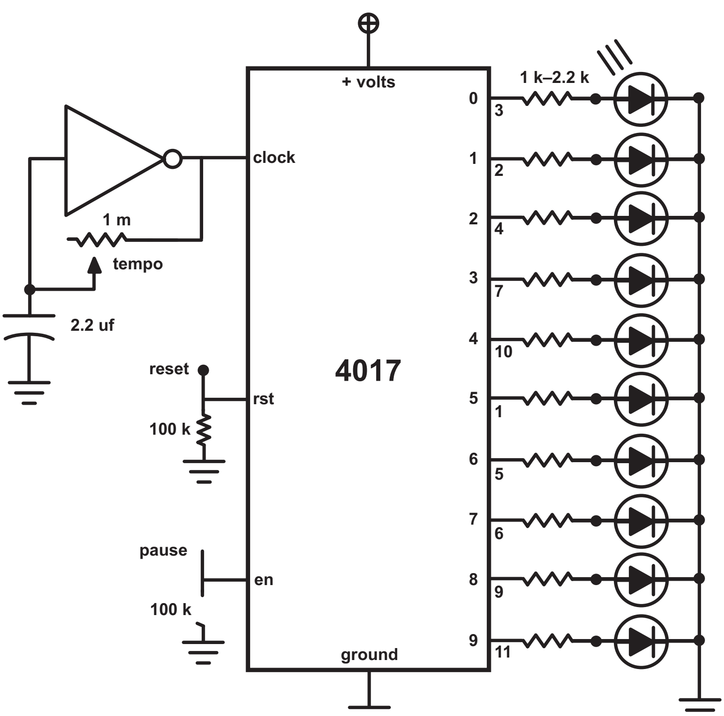


Figure 20.10
Simple 4017 LED sequencer.

I've not included a breadboard layout for this circuit because the snarl of all those resistors and LEDs is almost impossible to reduce to a meaningful image. But by now you should be pretty used to translating from schematics to the breadboard, so this is as good a time as any to take off the training wheels.

Connecting pin 15 (Reset) to +9 volts through a jumper (leaving the 100 kOhm resistor in place) immediately resets the count to 0; when the jumper is removed, the 100 kOhm resistor pulls the pin back down to ground and the count starts up again. Jumping pin 13 (Enable) to +9 volts pauses the count, which resumes when the jumper is removed and the pin is pulled back to ground by the resistor.

By coupling a photoresistor to each LED, we can turn on and off multiple audio signals sequentially or pan a signal among 10 speaker channels—just follow the guidelines for gates and panners from Chapter 17.

By adding a potentiometer and diode to each output of the 4017, we can construct a basic analog sequencer that generates a control voltage for each of its 10 steps.[1] This connects neatly to the VCO in the 4046 (Figure 20.11). The pitch of each step of the sequence is adjusted with a 10 kOhm pot: rotating the shaft sets the voltage anywhere from 0 to 9 volts, which should cover the full frequency range. We've kept the LEDs from the previous circuit so you have some visual feedback on where you are in the sequence. The resistors on pins 11 and 12 of the 4046 set the maximum and minimum frequency of the VCO. I've shown good starting values; you can experiment with other values to limit the VCO's range to useful audio frequencies and to maximize the resolution of the pitch-control pots. You can use other size pots, but it's best if they all have the same value. The Reset and Enable pins can be momentarily connected to + voltage to reset and pause the sequence as we did prior with the LEDs.

If you want a sequence shorter than 10 steps, simply take the output one step greater than the highest step you want and connect it to Reset: i.e., for a four-step pattern, connect pin 10 to pin 15. You can wire up a 10-position rotary switch that

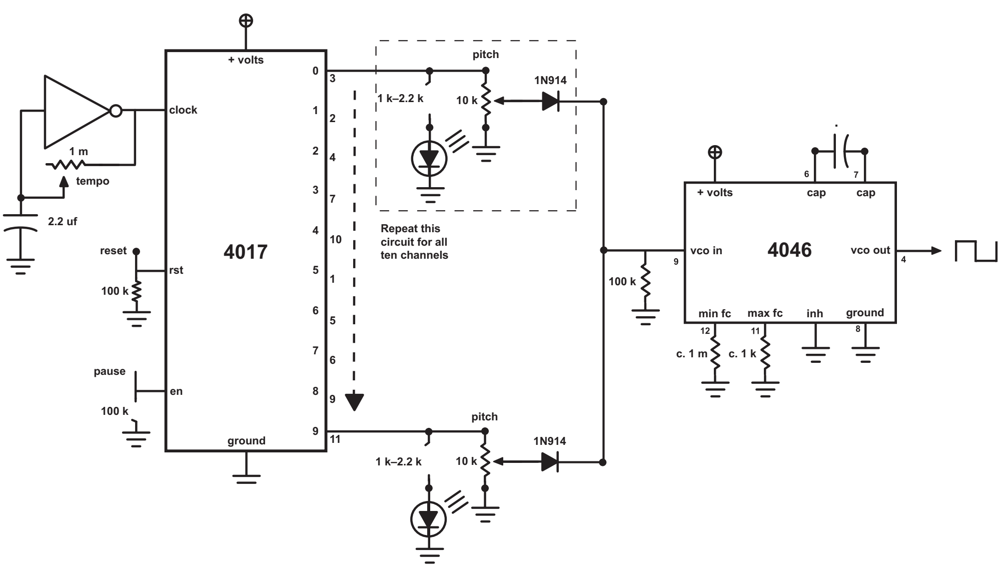


Figure 20.11 10-step analog sequencer driving a voltage-controlled oscillator.

selects any output to reset the count and make the sequence length easily adjustable. Add some pushbutton switches for Reset and Pause and you'll be in techno heaven.

As before, the profusion of pots makes a breadboard drawing impractical. In the interest of clarity, the sub-circuit that generates the control voltage and lights the LED for each stage has been shown only for the first and last steps of the sequence

It's possible to cascade multiple 4017s to make longer sequences—download a PDF data sheet on the 4017 and you'll have a schematic that will do the trick. You can also replace the simple 74C14 clock with any other source of square waves: chained 4093s replace the steady Kraftwerk-beat with off-kilter rhythms; track something with another 4046, divide down its pitch with a 4040, and the sequencer speed follows the pitch of your source (more or less).

For an experiment in wavetable synthesis, remove the 4046, the LEDs, and their resistors from the circuit. Replace the diodes with 10 kOhm resistors. Tie the free ends of these resistors together and send them to an amp. Replace the 2.2 uf capacitor in the clock with a smaller one (0.1 or 0.01 uf) so that it runs in the audio frequency range (Figure 20.12). Now the pots adjust the levels of individual segments of a 10-stage waveform generator, rather than control voltages going to the VCO in the 4046. Varying the step levels changes the timbre of the waveform. Follow this circuit with a simple low-pass filter (like that in Figure 19.5 in the previous chapter) or a basic graphic EQ pedal and you've got a very flexible, multi-timbre oscillator. The pot in the clock circuit can be replaced by a photoresistor, electrodes, etc. to make it more performable, and cascaded 4093s can be substituted for the simple 74C14 clock (as described prior) for more variation.

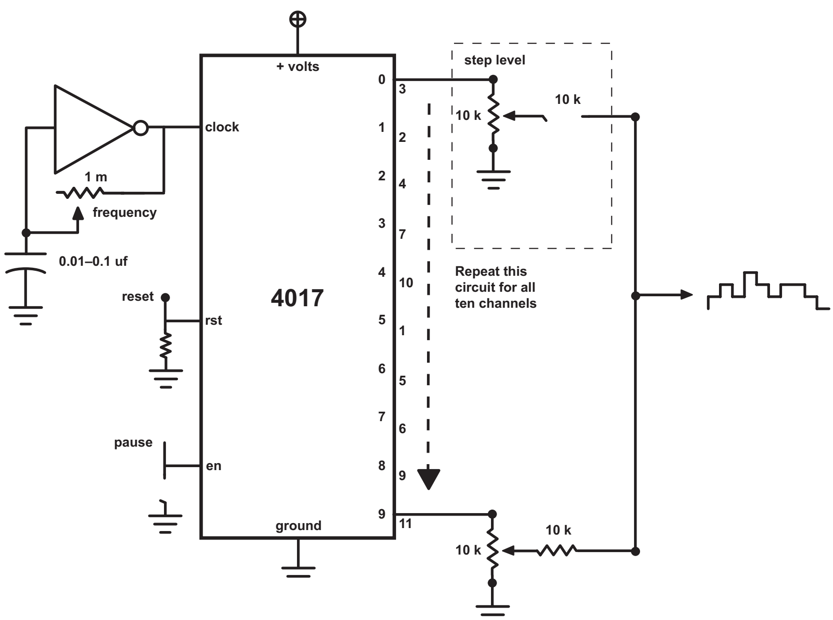


Figure 20.12 Wavetable synthesizer using 10-step sequencer.

NOTE

1. The 4017 can be found at the heart of many sequencers manufactured from the 1970s to the present day. In fact, you can connect the output of our design to the CV input of any synth module, instead of the VCO on the 4046.

CHAPTER 21

Beyond Bending
Triggering, Sequencing, and Modulating Circuit-Bent Toys

ALEX INGLIZIAN

You will need:

- Electronic sound-making toy.
- Synthesizer and/or drum machine with trigger/CV inputs and outputs.
- Soldering iron, solder, clip leads, and hand tools.
- 1/8-inch (3.5 mm) mono patch cables and panel-mount jacks.
- Breadboard.
- CMOS Hex Schmitt Trigger Inverter (74C14, CD4584, or CD40106).
- CD4017 Counter.
- CD4040 Binary Counter/Divider.
- CD4049 CMOS Hex Inverter.
- Assorted resistors, capacitors, and pots.
- Photoresistors and LEDs.
- Solid and stranded wire.
- 9-volt battery and connector.
- Multi-channel computer audio interface (optional).

CIRCUIT BENDING

Traditionally, making functional electronic objects has necessitated a fair grasp of theory and a pretty clear idea of what you wanted to make *before* you pick up your soldering iron. David Tudor, Gordon Mumma, Composers Inside Electronics, and other musical designers began chipping away at these assumptions in the 1960s and 1970s. Being self-taught, they had only piecemeal knowledge of electronic theory and were less concerned about doing things "properly" than about making something that sounded cool. Immersed in a musical ethos that valued chance, they were highly receptive to accidental discoveries—in the pursuit of the "score within the circuit," they relished wandering down side paths, rather than race walking toward a predetermined goal.

In the mid-1990s, Reed Ghazala pushed this serendipity back to the fore of electronic practice with his fervent advocacy of what he dubbed "circuit bending." Like Michel Waisvisz (see Art & Music 7, "The Cracklebox," Chapter 12), as an adolescent in the late 1960s, Ghazala encountered the sounds of accidental circuit interaction: an open amplifier left in his desk drawer shorted against some metal and began whistling. After some experimentation, Ghazala added switches so he could control the shorting, and circuit bending was born. He developed a series of techniques for modifying found circuitry—especially electronic toys, whose sonic sophistication grew in direct response to the boom of semiconductor technology in the 1980s—without the benefit of the manufacturer's schematics or any engineering knowledge whatsoever. In 1992 he began publishing instructive articles in *Experimental Musical Instruments* (an influential journal for instrument builders) and gradually acquired a cult following. In 1997 he launched his website, and today a cursory web search will reveal news groups, festivals, and workshops for circuit bending all over the world.

Circuit bending is freestyle sound design with a postmodern twang—the perfect escape for artists bored by the powerful, but often stultifyingly rational, software tools that increasingly dominate music production, yet still hooked on the digitally inspired cut-and-paste aesthetic of scavenging, sampling, and reworking found materials. With its defiantly antitheoretical stance and emphasis on modifying cheap consumer technology, bending has a natural egalitarian appeal (as well as some odd orthodoxies).

Earlier editions of this book included chapters on basic circuit-bending techniques, but these have been omitted in this revision to make room for newer material (although they can be still be found on the website). Bending flourished at a time when sound-making toys from the 1980s and 1990s filled thrift shops and flea markets. The circuit boards inside those toys were crowded with chips, resistors, and other components, rich with options for new connections and part substitutions. But open up a modern toy today and you typically find a single malevolent epoxy blob encasing everything—if toys are cars waiting for hot-rodding, lately they all come with the hood welded shut. The arc of electronic design is toward smaller components and greater density of those components on a single integrated circuit. Now circuit benders have to forage on eBay for vintage toys, and demand has driven up prices.

That said, the sound world of hacked toys can be deliciously rewarding. On the assumption that you'll occasionally encounter the perfect toy at an affordable price, we offer a chapter on combining bent instruments with the circuit designs elsewhere in this book and integrating them into tempo-based music production systems. Incorporating clocking and sequencing circuits with hacked toys opens up a world of possibilities and pushes the edges of electronic music production.

HOW TO CHOOSE A TOY

As with the radio in Chapter 12, select a toy that is expendable, not too tiny, and has a built-in speaker. A toy that makes sound is preferable to a mute one, and sampled sounds (like voices, animal sounds, or instruments) are more rewarding than simple beeps. The more buttons and switches the better, generally speaking. Keyboards are a gamble: some cheap Yamahas hack magnificently (the PSS-140 is especially satisfying) while others have curiously limited potential for interesting modification. A toy manufactured before 2000 will be more hackable than recent ones. One very important feature for the projects in this chapter: it should be *retriggerable*. Many modern toys do not have the ability to play a sound before the previous sound has completed its cycle. This is not ideal because, if our goal is tempo synchronization, we would be at the mercy of the length of the sound before any new triggers can be received. With a toy that can be retriggered, the sound can be played repeatedly at any tempo regardless of its length (sometimes all the way up to audio rate). This allows external triggers and sequencers to control sound playback from your toy, enabling tempo synchronization and pseudo-wavetable synthesis at audio rates. In addition to retriggering, some toys *loop* their sound when a button is held, opening up more sonic possibilities.

So when you're in the shop:

- Each time you press the button on your toy, check that the sound starts playing again from the head, no matter how fast you punch.
- If possible, choose a toy that continues to loop its sound as long as you keep your finger down on the button.

FIND THE TRIGGER

The first step after bringing home your chosen toy is to reverse engineer how the sounds are triggered. Open it and expose the circuit board underneath the buttons. Lift up the board and locate the spots that make contact with the buttons. You will likely find a little rubber nipple sandwiched between the plastic button in the case and the circuit board. You may notice a dark gray dot or ring on the underside of the nipple, made out of conductive rubber, that presses down against gold interlocking traces on the circuit board (Figure 21.1). When bridged by the conductive rubber, the traces are shorted together, causing the sound to trigger.

With the switch traces exposed, try triggering the sound by shorting the connection with a single clip lead to confirm your device is still functional. If your circuit board includes multiple buttons, take a close look and you'll notice that one of the two traces for each button is probably common throughout all buttons. In most cases, this common signal is the negative voltage from the batteries (ground). To confirm this, try connecting one end of a clip lead to the negative battery terminal and touch the other end to one of the interlocking circuit traces *not* connected to the common signal (Figure 21.2). If the sound is triggered, you are on the right track.

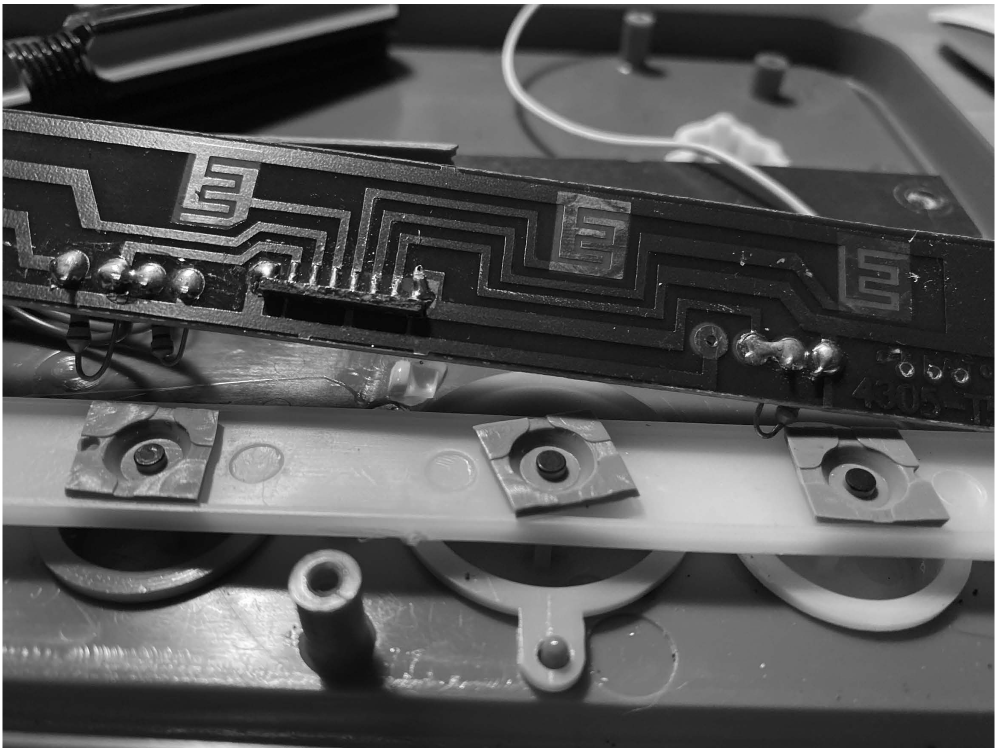


Figure 21.1 Toy button assembly—interlocking circuit traces—common PCB trace.

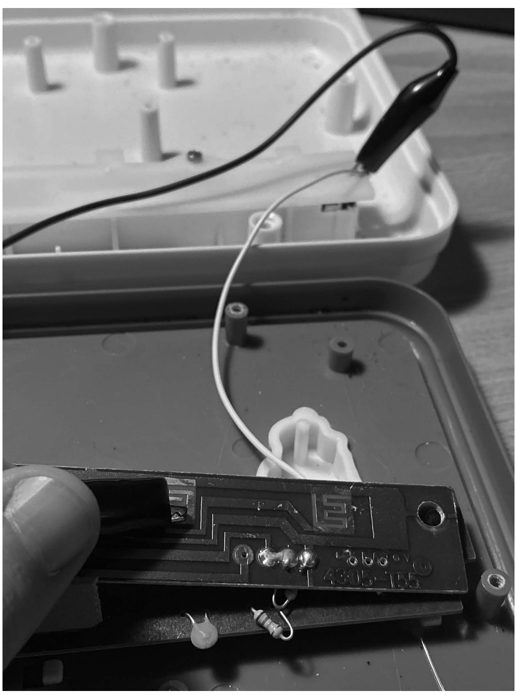


Figure 21.2
Clip lead connection from negative to PCB trace.

EXTERNAL TRIGGERING

Now that we have located the point in the circuit that triggers the sound when shorted to ground, we can begin to incorporate our external equipment. In this example, I will be using a Roland TR-606 drum machine from 1981, but any modern drum machine or sequencer with a trigger output will work (Figure 21.3).

Connect a mono patch cable to your drum machine's trigger output. Attach one end of a clip lead to the *sleeve* of the plug at the other end of the cable and the other end to your toy's negative battery terminal. Solder a short solid wire to the circuit trace that corresponds to the sound you want to trigger. Clip one end of a second lead to the *tip* of the patch cable and the other end to this wire jumper (Figure 21.4). Now dial up a funky pattern on the sequencer and with luck the drum pattern will trigger the toy's sound.

WHAT'S HAPPENING?

The trigger signal from the drum machine is a quick spike of positive voltage that falls back down to ground almost immediately. This signal activates the toy's sound the moment the voltage drops from positive to negative. This transition is an automated equivalent of connecting the clip lead with battery negative to the toy's trigger input. We can use *any* cycling or oscillating voltage from *any* source as long as the negative voltages are common between the two devices. Additionally, if your toy has the ability to loop when the button is held, this means that an external trigger signal will loop the sound as long as the waveform's pulse width is extended, instead of simply triggering the start of the sound at the high to low transition. Using a trigger source with pulse

Figure 21.3
TR-606 trigger outputs (at top of image).

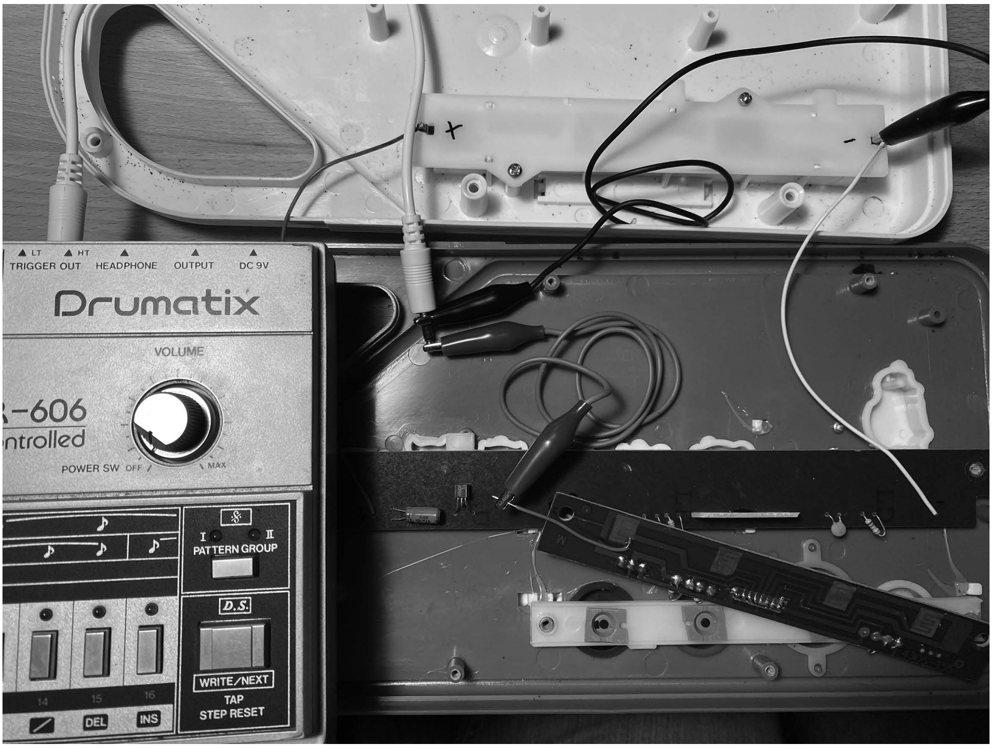


Figure 21.4 TS plug clipped to negative voltage and the circuit trace.

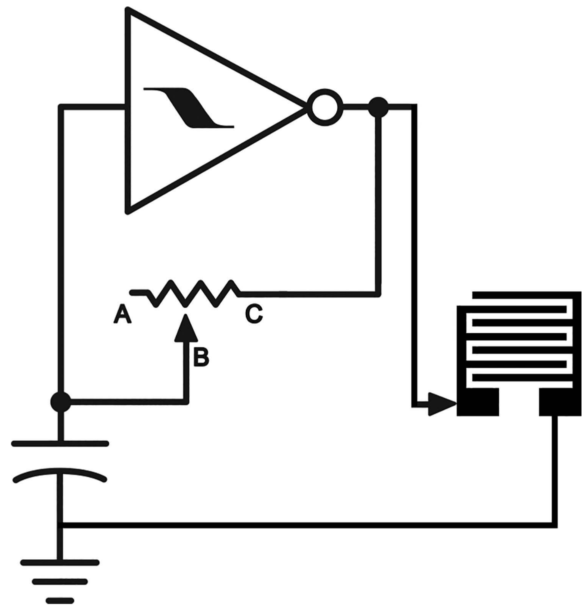
width modulation, such as a pulse wave from a modular VCO or keyboard synthesizer, will yield good results.

SEQUENCING AND SYNCHRONIZATION

Expanding on the technique of external triggering, we can now begin to experiment with other methods of control. In Chapter 13 we built a bank of six square wave oscillators from a single CMOS chip. The output of each of these oscillators can be used to trigger an individual sound on your toy using the same method: link the grounds between the toy and the oscillator circuit and connect the output of each oscillator to a button trace (Figure 21.5). You can even power your oscillator circuit from the toy's internal batteries.

Using multiple square wave oscillators to trigger sounds on your toy can produce complex rhythms since the rate of each oscillator is independent from the others. This creates cool phasing patterns as well as complex wave shaping when the driving oscillators are tuned to audio range. However, this may not be ideal if your intention is to make people dance. For dance music we will revisit the CD4017 counter chip from Chapter 20 and build our own step sequencer (Figure 21.6). With a common ground voltage to your toy, each step output from the CD4017 circuit can trigger a different sound.

Figure 21.5
Hex inverter connected to toy trigger.



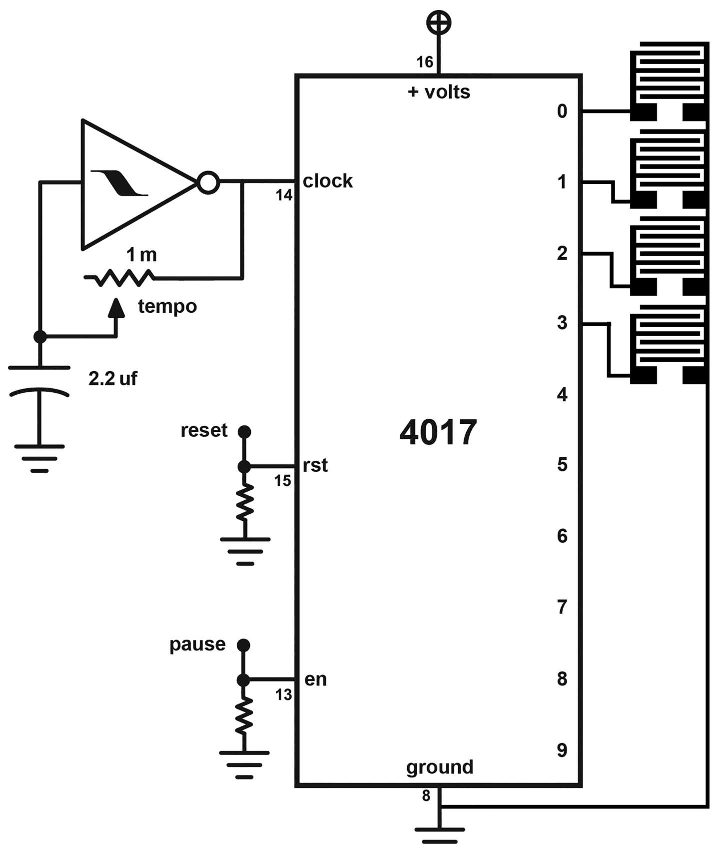


Figure 21.6
CD4017 sequencer
connected to toy triggers.

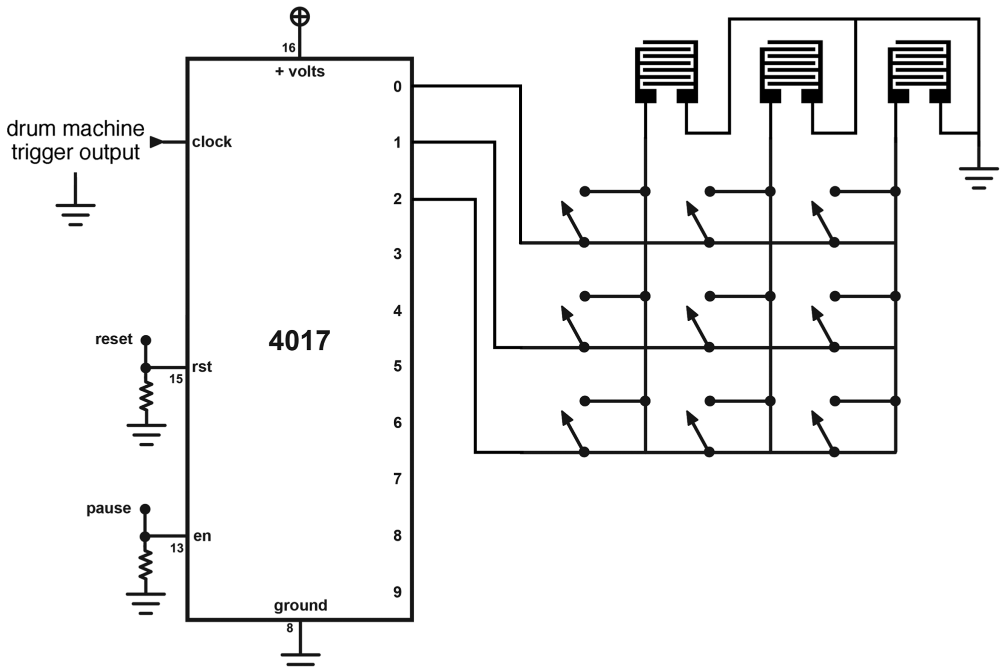


Figure 21.7 Schematic for switched multi-channel sequencer.

We can also build a multi-channel programmable trigger sequencer by utilizing a simple switched routing matrix at the output of each step of the CD4017. Returning to the discussion in Chapter 18 about computer keyboard matrices, we can wire together an X–Y grid of toggle switches that connect an input row to an output column. With a step pulse at the input, and each output connected to the toy's sound trigger, we can now have a programmable sequencer similar to the "TR-style" old-school drum machines of the 1980s (Figure 21.7). Furthermore, this can be clocked/synchronized from any of your external studio hardware with analog clock outputs (drum machine, sequencer, modular synthesizer, etc.).

Many popular modern drum machines and sequencers have an analog sync output, but the rate of the clock may be too fast to be useful in this scenario (e.g., Korg Monotribe, Korg SQ-1, and Korg Volca series). In this case, we can use the CD4040 12-Stage Binary Counter from Chapter 20 to divide the signal down to a more useful tempo.

Without opening up the can of worms that is digital microcontrollers (such as the Arduino—see Chapter 33), there's an easy way to accomplish external triggering and synchronization of circuit-bent toys with a computer-based music production setup using only audio signals. All you need is an audio interface with a spare analog output. In the Digital Audio Workstation of your choice (i.e., ProTools, Logic, Ableton), set up a synchronized click track routed to a dedicated audio output on your interface (Figure 21.8). Crank the output volume of this track as high as you can

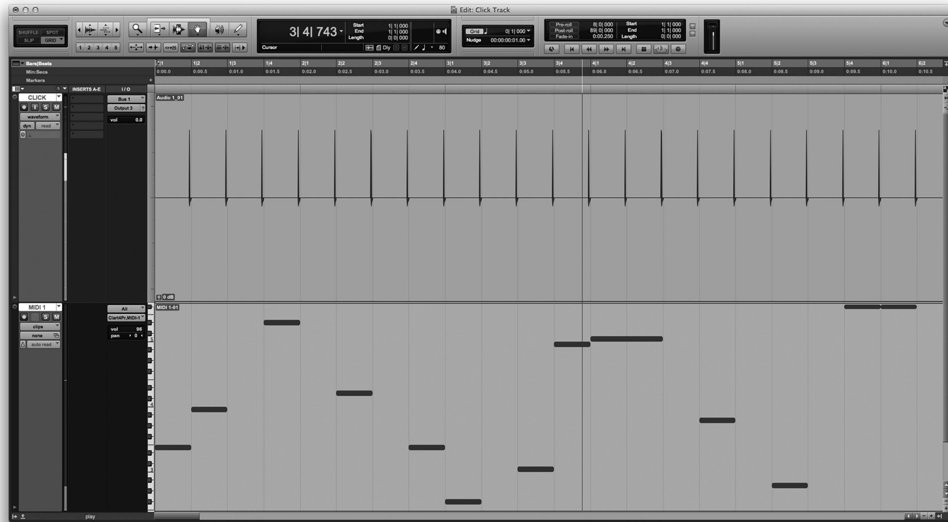


Figure 21.8 Screenshot of DAW click track.

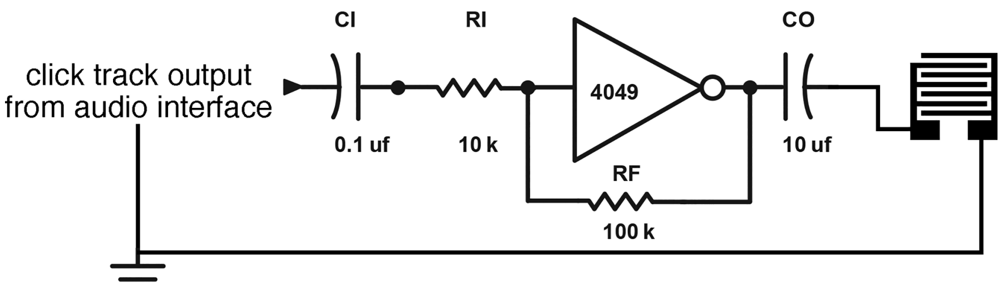


Figure 21.9 Schematic of CD4049 amplifying click track.

in your software and the interface controls. Hook up a cable from the interface to a trigger pad on your toy in the same way we linked from your drum machine. In most cases, the audio from the click track has a strong enough voltage to trigger the toy, but if it's not working, try amplifying the signal using the simple CD4049 pre-amplifier circuit from Chapter 19 or the op amp preamp we'll get to in Chapter 23 (Figure 21.9).

In addition to triggering a sound on your toy, this method can also be used to clock a CD4017 sequencer circuit from software running on your computer. If you have a multi-channel audio interface, you can program a custom multi-channel trigger sequencer in software such as MaxMSP, Pd, or any MIDI programmer—each output of your interface sends a different click track to a different switch pad on your toy (Figure 21.10).

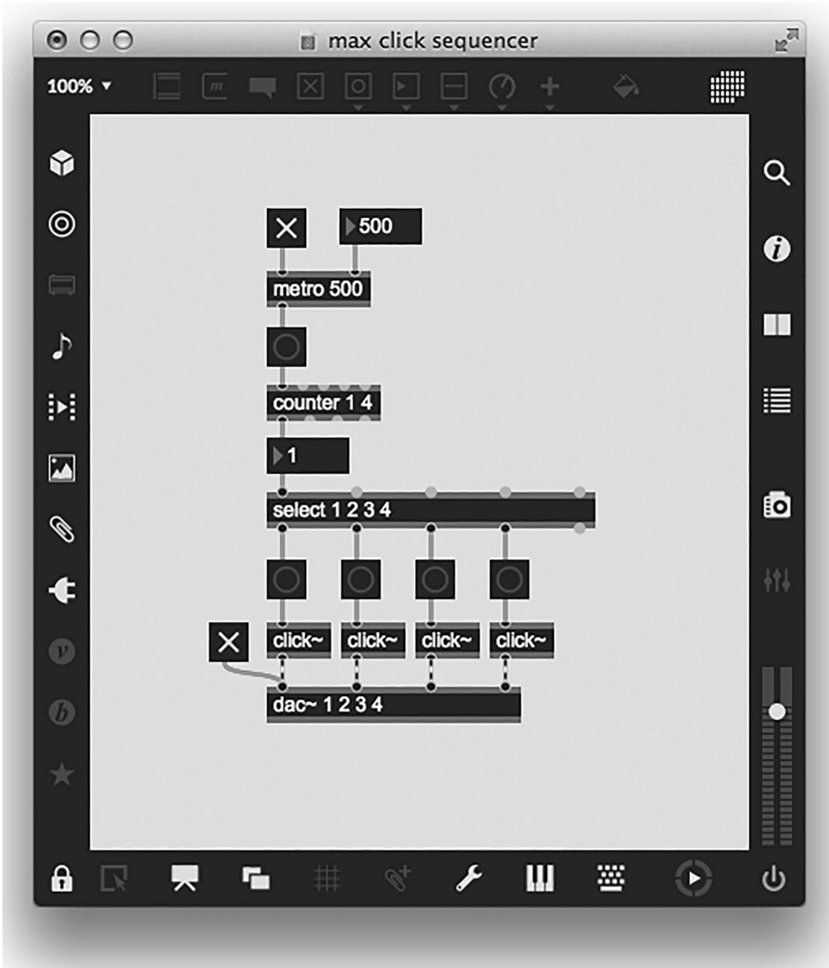


Figure 21.10 MaxMSP sequencer for triggering multiple toy sounds.

MODULATION

Beyond triggering sounds, synchronized modulation is an important feature that can greatly expand your sonic palette. Common targets for modulation are amplitude and pitch. A simple technique uses the core circuit comprised of an LED and photoresistor bundle that we built in Chapter 17 (or available commercially under the name of a Vactrol or analog opto-isolator). You can take the output from your drum machine, sequencer, or modular synth clock to drive the LED instead of the oscillator used in that chapter (Figure 21.11). In most cases the trigger or clock voltage is enough to activate the LED, but you may need the same 4049 booster we used with the sync pulse. Route the audio from your toy through the photoresistor in the LED-controlled gate circuit and you have tempo-synched gating and tremolo.

Figure 21.11 LED-controlled photoresistor with drum machine trigger input.

Photoresistors vary in their latency between changes in illumination and changes in resistance, which smooths the wave shape of the modulation source—experiment with different ones to change your modulation waveform. Gating and ducking are a common technique in electronic dance music production (and Dub) and offer excellent ways to inject some groove into a static glitchy sound. You can also use triggered photoresistors to control pitch control, enabling tempo-synched vibrato and frequency modulation.

PACKAGING

The techniques covered in this chapter require very few components; the circuits have tiny footprints and can often be mounted inside the toy's case. Modularity is important in an electronic music studio, and maintaining this openness with your hacked toy is a simple task. For each trigger input to your toy's sounds, we can wire a 1/8-inch (3.5 mm) panel-mount jack and attach it through the plastic case for patchable programing between your sequencers/clocks and your toy (Figure 21.12).

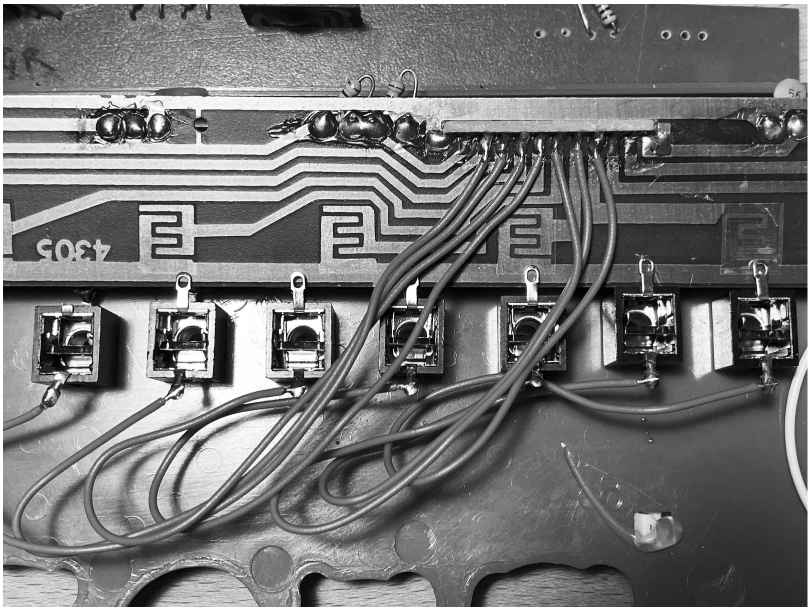


Figure 21.12 3.5 mm panel-mount jack wiring.

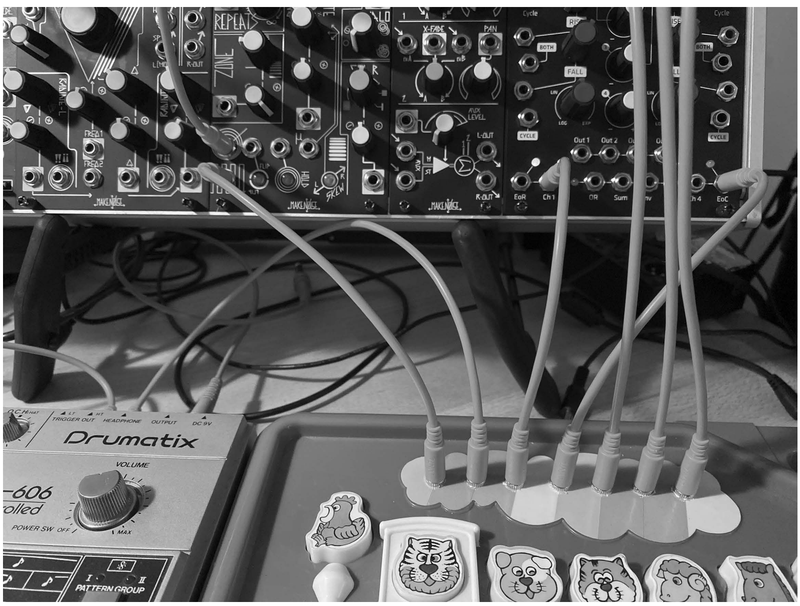


Figure 21.13 Completed toy hack patched to a sequencer and modular synthesizer.

We can also install the LED-photoresistor element inside the toy's case with another 3.5 mm jack. Wire the jack to the LED and the photoresistor to any resistance-controlled parameter you would like to modulate

Your kiddy toy grows up before your eyes (Figure 21.13)....

CHAPTER 22

Video Hacking

LOVID (TALI HINKIS, KYLE LAPIDUS) AND JON SATROM

You will need:

- CD22402 video sync generator chip.
- AD724 RGB to NTSC/PAL Encoder.
- VGA test signal generator module.
- Hackable video devices (toy video paintbox, video camera, etc.).
- Video monitor.
- Three RCA jacks.
- Switches.
- Potentiometers.
- General breadboarding supplies.

HOW DOES VIDEO HACKING DIFFER FROM AUDIO HACKING?

The video signal is a finicky beast. This can make DIY video hacking and synthesis a bit more demanding than audio experiments. In audio hacking, we can simply run any signal into an amplifier, pass the signal to a speaker, and hear the everything. In video hacking, we need to conform to the curious (and demanding) electronic "grammar" of video signals.

At first, like in computing, video systems were large and expensive. Although video was dreamt up as a democratic medium (much like the telephone), it emerged as a broadcast technology. It wasn't until the late 1960s that video became accessible to artists and the public in the complementary forms of prosumer equipment and legislated public access television channels in the United States. From the moment artists were able to get their hands on video gear, alternative perspectives and approaches began to emerge. There's a rich history of video artists who have developed their own systems and inspired many contemporary hackers (see "Pixel Artists" essay on the website).

Video standards were created in the 1940s to display broadcast images on cathode-ray tubes (CRTs), which are much more forgiving than many modern displays. Newer hardware is extremely picky about the video signal it will display, particularly the precise timing of sync information and the frequency of the color burst. Manufacturers of newer flat displays and fancy projectors shield us from a poor signal's glitches and visual noise (admittedly, an acquired taste) with calming blue "no-signal" screens. It's like owning a sound system that shuts off whenever the music is out of tune.

To illustrate the complexities of a video signal, try simply plugging a video cable into an audio input. You will hear a steady drone whose overtones fluctuate in response to the image's content and brightness. The fundamental pitch is a function of the video frame and sync rate, and this remains fixed (as long as you don't move to another country with a different video standard) while the overtone balance represents the image data. By varying the brightness (i.e., pointing a camera at a fan), you can play with interference patterns between the frame rate and the fan speed. Aiming infrared remote controls at the camera can also be fruitful since most video cameras detect infrared light (and show it as a hot white dot). Going from video to audio is trivial because amplifiers are wildly forgiving and simply play any signal they're handed. Going from audio to video is another story because of the amount of discrete information needed within the video signal. There are a number of artists who play with the passing of audio signals to the video domain and vice versa (see a))<>((v in the "Pixel Artists" essay on the website) This often requires sync stabilization through a dedicated genlock device or a time-base corrector.

VIDEO SIGNALS: FROM THE FRONT TO THE BACK PORCH

The video standards created in the US were named after the National Television Standards (or System) Committee (NTSC), a group of engineers and policy makers. NTSC standards were developed for black and white transmission of images in the 1940s and modified in the 1950s to add color information to the video signal. This kludge of color data into the existing black and white signal gave NTSC the reputation of "Never The Same Color." The NTSC standards remained widely used until the twenty-first century, when digital standards became more prevalent. Similar standards (PAL and SECAM) were developed and used in other parts of the world.

The NTSC signal stream provides a rapid sequential series of still images (frames) at a frame rate of approximately 30 images/second. Each frame is comprised of 525 lines (480 are visible), delivered as two interlaced fields of 262.5 lines (odd and even scan lines) in the aspect ratio 4:3. Because NTSC was developed for display on CRT televisions, periods of time without image information were included in the video standard to allow movement of the electron gun back across the screen (like carriage return on an old typewriter) and from the bottom right corner back up to the top left after each frame has been drawn. The video signal contains image information that is divided into three primary colors: red, green, and blue (RGB). Brightness of the video is referred to as luminance while the chrominance includes the hue (tint) and saturation (intensity) of color.

The combined color image stream is called a *composite* video signal. Composite video is commonly found on older game systems and consumer video gear. It often connects through an RCA cable, and the jack is usually colored yellow (to distinguish it from the red and white commonly used for audio right and left). A signal with separate (not NTSC encoded) color channels is called *component* video and is sent via multiple RCA jacks (each carrying a different part of the signal) and is found in higher-end displays. Various component standards are used, including straightforward RGB and YPbPr, where a green cable carries Y (luminance (luma) and sync), a blue cable carries P_B (the difference between blue and luma), and a red cable carries P_R (the difference between red and luma).

Composite NTSC video is 1 volt peak-to-peak waveform and combines color, video, blanking, and sync information, conveyed by changes in voltage (Figure 22.1). The NTSC composite signal for each scan line includes a horizontal blanking interval of 10.9 us (the carriage return), with subcomponents containing specified voltages and timing that occur in this temporal order: the front porch, sync tip, breezeway, color burst (oscillation at 3.579545 MHz) and back porch (don't you love the rustic video terminology?).

Following each horizontal blanking interval, 52.6 us of active video signal is presented with brightness determined by voltage level and color information based on phase difference from the color burst. In between each field, there is a longer, vertical blanking interval, which initially was needed to allow time to adjust fields in the magnetic coils that vertically deflect the electron gun's beam but has subsequently been used to convey additional information (e.g., datacasting, time code, etc.).

Component video involves transmission of the different parts of a video signal on multiple channels instead of a single mix (Figure 22.2). In video synthesis and hacking, the use of RGB component video is very convenient. This typically involves three separate color channels and a separate sync signal. Because component video is not encoded into a single stream, the integrity and intensity of each color are preserved. However, in light of redundancy and extra bandwidth, component video is not frequently used for transmission and recording.

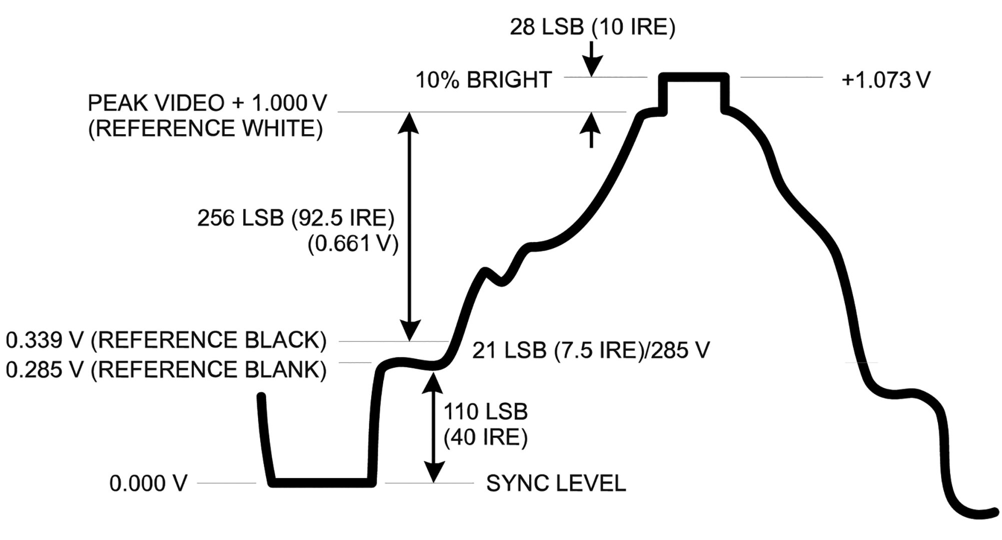


Figure 22.1 NTSC waveform.

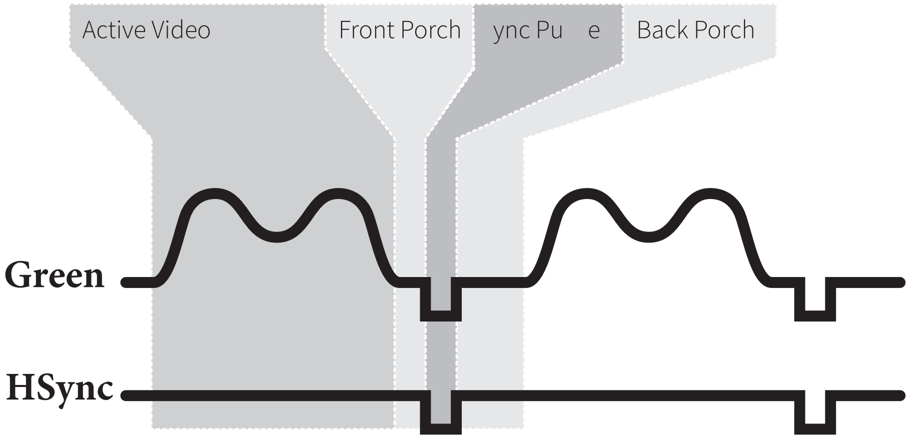


Figure 22.2 Component versus composite video.

SYNC GENERATING

One way to generate video sync signals is by using a sync generator chip (e.g., Harris CD22402).[1] Once the sync and blanking signals are available from this circuit, they can be directly used to generate component video signals, which can then be encoded to create composite video (Figure 22.3). Because blanking signals swing from 0V during low periods on horizontal and vertical sync pulses to 4V to 15V (depending on supply voltage for

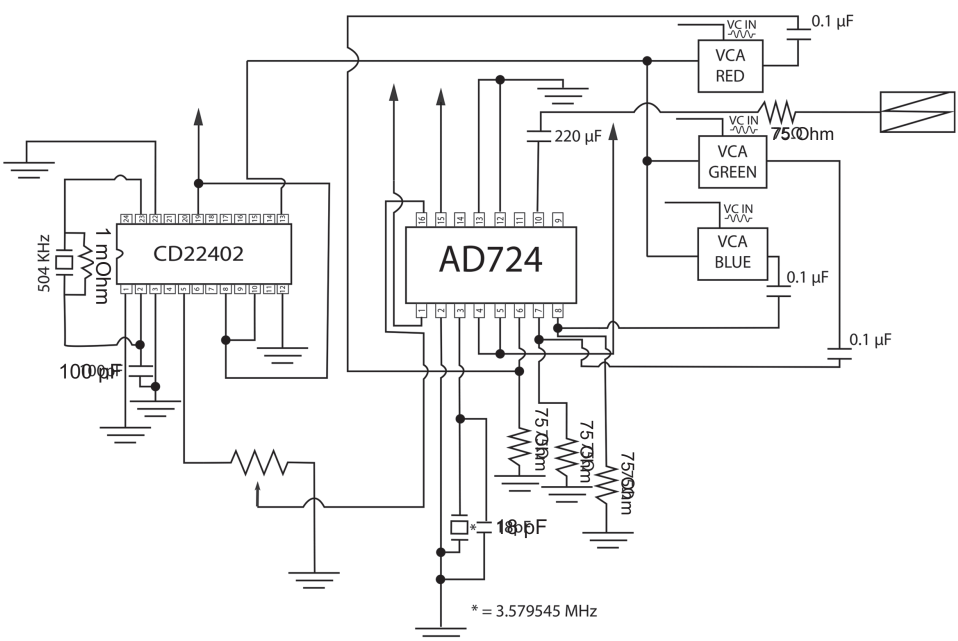


Figure 22.3 Component-to-composite sync video generation, LoVid schematic.

the CD22402) when video information content is acceptable, they can be sent as *signal inputs* to a voltage-controlled amplifier (VCA) in a synthesizer. Any voltage information to be displayed as a video image can be used as a control voltage for a VCA. These can be relatively low frequency, like standard audio frequencies, or the higher frequencies that generate vertical lines and more involved/complex images. A multiple of 60 Hz will display that multiple's number of horizontal lines and a multiple of 15570 Hz (60 × 525/2) will display that multiple's number of vertical lines. The outputs of the three VCAs can then be used as red, green, and blue inputs to a video encoder (e.g., Analog Devices AD724).[2] When a sync input from the CD22402 is also provided, the AD724 outputs composite video. A similar approach can be used with a composite video signal, with sync information being imposed. Gijs Gieskes has built a Time Base Forcer that combines two video inputs and allows for stable messing with the sync to generate glitches that often cannot be captured on contemporary displays or recording equipment.

SYNC PLUNDERING

Another approach for generating your own sync involves taking it from a pre-existing signal generator. One efficient example, developed by Jonas Bers, is the CHA/V (Cheap, Hacky, Audio/Video).[3] It is inexpensive and easy to make and uses commonly available Video Graphics Array (VGA) test signal generators.[4] These generate component video output (usually switchable between color bars and full screen of a single color) that can be displayed on standard computer VGA monitors, as well as any other monitor or projector that accepts VGA inputs. Inexpensive adapters are also available to transform VGA into a composite video signal via an RCA/phono connector. The pins of the VGA output plug (a female 15 pin D-subminiature connector) on the test signal generator are connected to pads on the board, to which you can tack solder

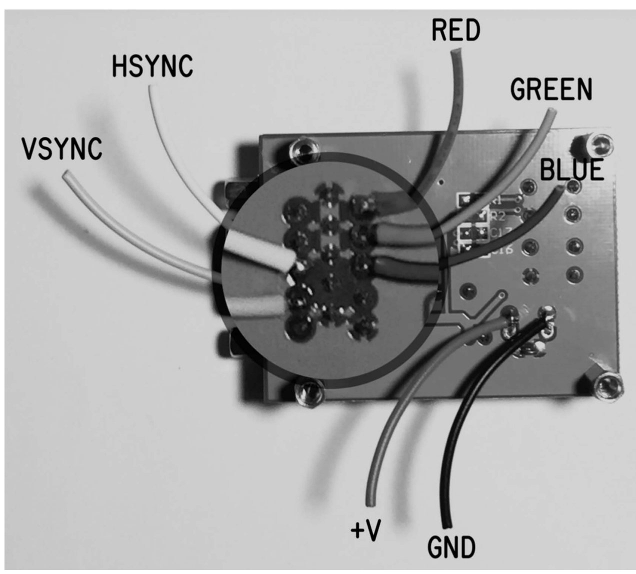


Figure 22.4
CHA/V solder side.

wires (Figure 22.4). Running any signals, via a coupling capacitor, into the red (pin 1), green (pin 2), or blue (pin 3) pins will impact the display of each color: try a separate oscillator from our trusty Schmitt Trigger circuit (see Chapter 13) into each color pad or three parallel tracks from a music session. Horizontal (pin 13) and vertical sync (pin 14) outputs from the test generator chip are also available and can be used to sync oscillators to make stable lines or patterns.

SYNC SNUBBING

You can also extract sync from a hackable device that outputs composite video: old digital cameras, drawing toys, and video mixers already output legit NTSC video. Messing with these devices lets you ignore the engineering (for the most part) and fiddle with hacking methods laid out in this book. When looking at the circuit board, identify the area around the output video signal and steer clear of it (to limit risk of making solidly blue video).

Spending time identifying characteristics of video glitches will help you make decisions on how volatile you want your system to be. The types of glitches roughly break down into three categories: sync glitches, system glitches, and sh**! glitches. Sync glitches tend to be only visible on older CRTs. They often squiggle the image and can skew the image off center, showing it halfway up the screen or off screen. These types of glitches will be difficult to record (or send to modern displays) at a signal level because they confuse capture and display equipment—proof positive of the rigid orthodoxy of sync. The tried-and-true way of capturing sync glitches is to record them off your CRT with a camera. System glitches are summoned from within a device and give you a pallet that hints at the device's original function. Sh**! glitches are when you accidentally short something and the device never wakes up (with the complexity of video devices, you're often only steps away from sh**!). While these artifacts may have become overpopularized in horror films and Instagram filters, nothing beats the feeling of accomplishment of finding unique ruptures born from video hacking.

FINGERING SYSTEMS

When hacking a found video device, look for chips that can contain any of the "logic" of the device and try the wet fingers approach (see Chapter 12). As you identify fruitful areas of the board, take note so you can come back to them later. Once you've developed your glitch treasure map, use a small resistor and begin jumping between pins on the components (as you would in classic circuit bending). From there, you can add details to your map and document new functions. Some jumps may shift the *image* while other areas of the board may deal with the *logic* of the internal program. For example, a television drawing toy like My First Sony™ Sketchpad or the Video Painter can be hacked to take (loose and often unexpected) advantage of the behavior of the drawing tools as well as garbling the image directly.

Video mixers that have effects functions can also be hacked to add more texture and glitchy functions. Retro game systems, from the Atari to the Xbox, can be tickled into barfing pixels on the screen. Computer graphics cards can also be shorted to generate glitches (Figure 22.5).

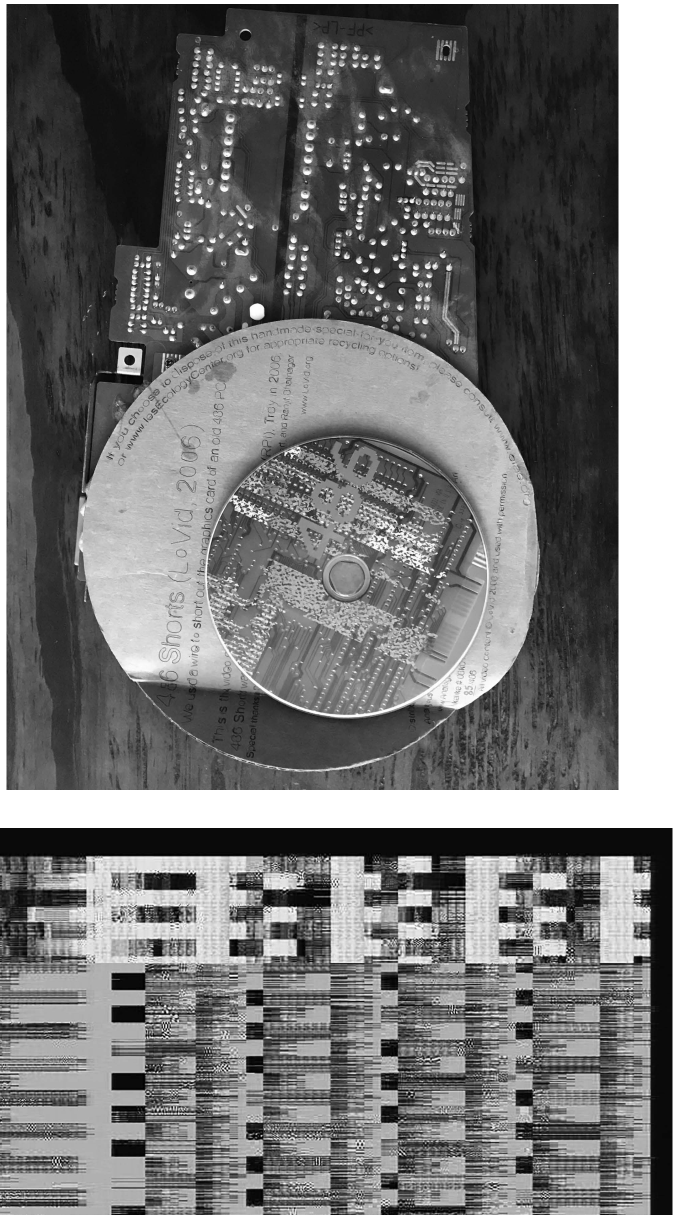


Figure 22.5 LoVid, *486 Shorts*, DVD package and screen shot.

DIGITAL CAMERA HACKING

Finding an area of an old digital camera's circuit that messes with the capture of light through the lens can produce smeary glitchy goodness—in still image and video form. Again, a simple bend or "laying of the hands" can go a long way. (A word of warning for digital cameras with a flash: stay away from the flash capacitor . . . it packs a Taser-like punch!). Explore areas of the circuit board around any of the larger chips. Take a small resistor and poke pins to see what happens when variations are shorted. Often, there are rows of leads that affect the image when connected. By interfering with areas of the board that capture the image from the lens and light sensor, you can trigger the snapping of photos and recording of videos. Many of the effects are smeary and ghostly. If you notice lines and harsh geometry on the device's screen, you may have found the area responsible for the display. If your camera's screen is busted, many cameras have a monitor out, which previews your bends and identifies screen bends versus image sensor bends. You may need to hunt for a proprietary cable. Sometimes it's mini-USB to composite. (Tip: car headrest video screens are great small preview monitors.) When you find a pleasing bend, solder it up to a switch or throw a 1 kOhm pot on it to be able to "sweep" into and out of the bend. Many of these cameras have small buttons that are integrated with their cases; make note of what buttons do what and don't hesitate to solder up "helper switches" that make actions like turning on and off the device easier (Figure 22.6).

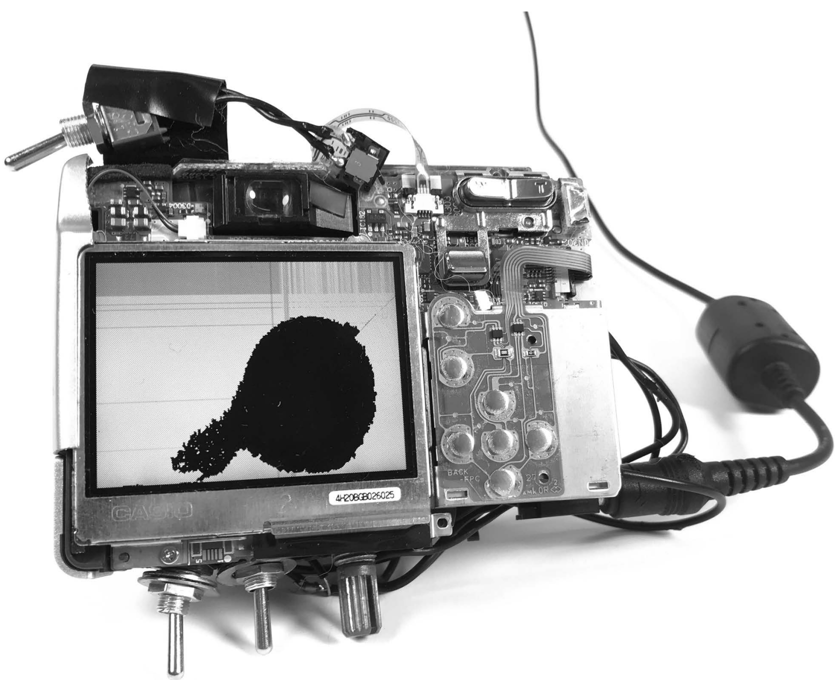


Figure 22.6 Jon Satrom, Fuji FinePix hack.

Figure 22.7
Barbie cam.

The weirder the camera, the better (Figure 22.7). Now that we all have cameras on our phones, older digital snapshot cameras are pretty easy to find cheap.

MIX IT UP

A convenient circuit for mashing two video signals together is Karl Klomp's "Dirty Mixer."[5] Klomp is a Dutch artist who has created numerous creative video instruments from hacked mixers and prosumer gear. The dirty mixer is dirt cheap and super simple. It uses a 1 kOhm potentiometer, a couple of switches, and three RCA jacks. It fits perfectly in an Altoids tin and is a great little tool for adding glitchy textures to any video hacker's arsenal (Figure 22.8).

To make your own little Dirty Mixer, you'll need:

- Three RCA jacks (it's handy to have ones of different color).
- Two switches.
- One 1 kOhm potentiometer.

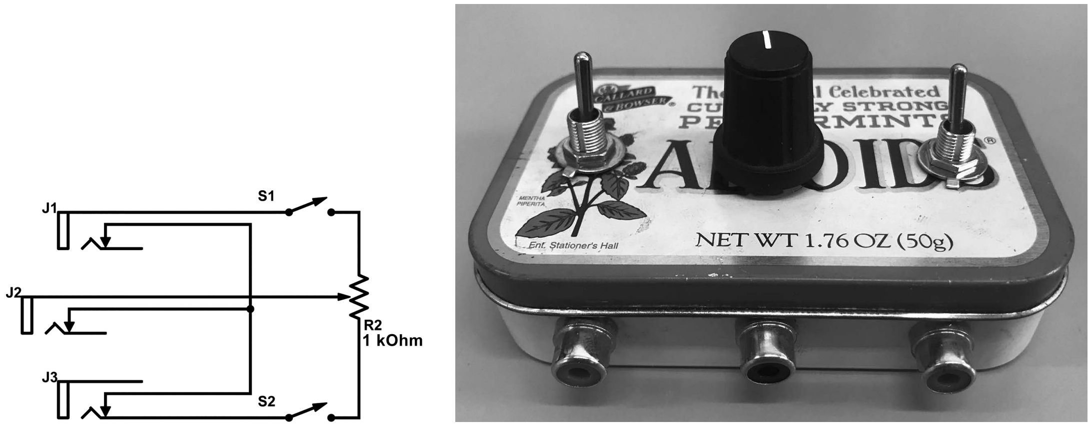


Figure 22.8 Dirty Mixer.

Two of the RCA jacks will be inputs, and one will be an output. In Figure 28.8, the inputs are on each side of the output RCA. Start by connecting the shields (outer lug) of your RCA jacks to each other. Next, connect the tips (inner terminal) of the two input RCA jacks to the switches. The switches allow an input signal to be turned "off" in the case that you want a cleaner signal from one input. Connect the other side of the switch to the outside leads of the potentiometer. The middle lead of the potentiometer will be connected to the tip of your output RCA jack. When both switches are on, signals from input A and input B are mashed together based on the setting of the potentiometer—like a crossfader.

Because the dirty mixer completely neglects sync, the results may vary: you'll be fine on an older CRT, but for many displays you will still need a time-base corrector (TBC) or something to "stabilize" the output signal. Many professional and prosumer video mixers have TBCs built into them. There are a few models of stand-alone TBCs available on sites like Amazon for around $100. One can also find rack-mount TBCs that have been decommissioned from television studios on eBay. They're always too big to tour with comfortably but are great in the studio.

MORE?

On the website you can find our "Pixel Artists" essay and video examples of several artists' projects with image processing, as well as two chapters and an essay from the previous edition of *Handmade Electronic Music* that address other pathways to visual hacking. And in Chapter 35, Nick Briz explains how to do image processing by data hacking.

NOTES

1. http://ee-classes.usc.edu/ee459/library/datasheets/CD22402.pdf
2. www.analog.com/media/en/technical-documentation/data-sheets/AD724.pdf
3. https://jonasbers.com/chav/
4. Usually available on eBay
5. https://web.archive.org/web/20100302135450/www.karlklomp.nl/pro/vbend1.html#dirtymx

CHAPTER 23

An Introduction to Op Amps

You will need:

- Some things to listen to: guitar, contact mike, cell phone, etc.
- A breadboard.
- A few op amps: LM324, TLO72, TLO82.
- Assorted resistors, capacitors, and pots.
- Assorted jacks and plugs.
- A small speaker.
- Some solid hookup wire.
- A 9-volt battery and connector or four AA batteries and a holder.
- Hand tools.

If you've made it this far in the book (congratulations!), you've probably also spent some time perusing other sources of information on things electronic—online and perhaps in print. In which case you've probably seen a lot of the little symbol in Figure 23.1.

The device is an operational amplifier (usually shortened to op amp, opamp or op-amp), and it has been the Swiss Army knife of analog electronics since it first appeared in integrated circuit form in 1963. As the name might suggest, it is an amplifier, providing signal boost (like the 4049 preamp circuit in Chapter 19). But, thanks to a deceptively simple arrangement of input and output connections, the op amp leverages this gain to perform a dazzling array of essential functions in almost every piece of electronic technology in use today, from seismographs to cell phones.

Then "Why," you ask yourself, "is it not in The Book?" I confess that the omission is in part a result of simple contrariness: the old avant-gardener in me enjoys leading you down the deviant path of adapting digital chips to do analog jobs. But I also benched the op amp in the interest of easing your transition into circuitry: its versatility comes at a cost. We built our first oscillator, back in Chapter 13, with just four parts (a chip, a capacitor, a resistor, and a battery). To make the same with an op amp would double that count, square your chances of a wiring error, and make it 10 times more likely you'd walk away from the book then and there. But over the course of 22

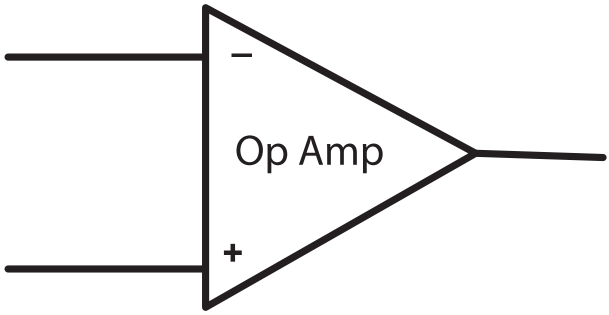


Figure 23.1
Op amp symbol.

chapters you've gained enough experience that I suspect you're ready to deal with the op amp's quirks. Spend some time familiarizing yourself with these chips and you gain access to a wide world of circuit designs.

CHOOSE YOUR CHIP

There are many varieties of op amps. The most relevant distinguishing characteristics are:

- Intended application. Some op amps are optimized for particular jobs: very fast ones for high-frequency radio and video signals; low-noise ones for audio preamplifiers; low-power ones for battery-powered devices. And then there are "generic ones" that serve as a basis for 90% of the circuits you might build.
- Cost. Some of the more specialized op amps are expensive while the general-purpose ones can be had for less than a dollar.
- Format. Op amp integrated circuits come with one, two, or four op amps in a single chip. Although SMD versions now dominate the market, for experimenting on a breadboard you'll want to pick up DIP versions: single and dual op amps in 8-pin format and quad versions in 14-pin packages.
- Single or dual power supply. Some op amps can operate with just a single power supply, like the 9-volt battery we've used in all our designs so far, or a 5-volt USB charger. But others require what is known as a dual or split supply: two batteries or a more sophisticated power supply (or tricking the op amp into thinking one supply is actually two—we'll get to this).

For the purposes of our experiments, I suggest you pick up two or three of either of the following options:

- LM324, a "workhorse" quad op amp.
- TLO82 or TLO72, a slightly quieter, dual op amp.

These are inexpensive, easy to find, and work well with a single battery. You'll also need the usual breadboarding supplies.

CONCEPTUALIZING

You can start by comparing the op amp to a section of the 4049 Hex Inverter that we used as a preamplifier in Chapter 19. Both have an output and an inverting input, labeled "−" on the op amp (Figure 23.1). You recall that we calculated the gain in the 4049 by dividing the value of the feedback resistor by that of the input resistor (if Ri = 10 k and Rf = 100 k the gain = 10). The op amp works the same way, with gain a function of the same two resistors. But the op amp has a second input, labeled "+," and this is where things get more complicated.

You may also remember that the 4049 preamp needed a capacitor at the output because the audio signal was riding on a DC voltage equal to half the supply voltage (4.5 volts in the case of a 9-volt battery). Inside the chip the circuitry raised the incoming audio to this voltage so that both the positive and negative parts of the waveform could be equally amplified. Without this reference the negative part of the wave would have been clipped off since 0 volts is the lowest extreme of the power supply (Figure 23.2) and the output can go no lower.

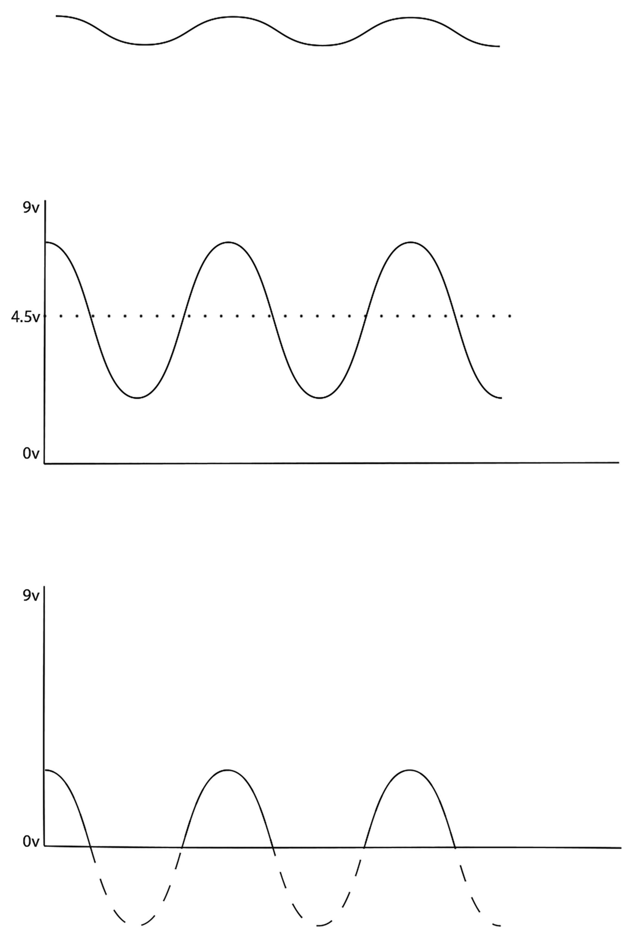


Figure 23.2
Waveform at input (top), amplified with bias voltage (middle) and without bias (bottom).

At the risk of oversimplifying, in lieu of the fixed reference (supply-voltage/2) that the 4049 adds to any input signal, the "+" input is where you set this reference in an op amp. In more general terms, the op amp is always comparing the + and − input signals against each other, and against the output if fed back, but for our first circuits you can think of the "+" input as the place to apply a bias voltage. So to replicate the 4049 preamp with an op amp, you'd set it up as shown in Figure 23.3.

Gain is calculated as before, by the ratio of Rf/Ri (100 k/10 k = 10x) at the − input. Resistors Rb1 and Rb2, both 10 k, form a voltage divider between + voltage and ground (−); since the resistors are equal, the voltage in the middle, where they connect to the + input of the op amp, is half the total supply voltage—with a 9-volt battery, this would be the same 4.5 volts as with the 4049. We added Cb (bias capacitor), around 10 uf, to add stability to this reference voltage. As before, you'll want to add the output and input capacitors to block the DC from thumping anything that follows. If we build this circuit on a breadboard, using a dual op amp like the TLO82, it might look like Figure 23.4.

Admittedly this bias circuit adds a few parts (and complexity) to our preamp design. But if you're using multiple sections of the op amp (perhaps making a stereo or quad preamplifier), you can use one bias circuit for all sections. And if you do step up to using dual power supplies (i.e., using two 9-volt batteries or making a circuit for your Eurorack synthesize system, with its dual supply), you can forgo the voltage dividers. Figure 23.5 shows the same preamp circuit running off of two 9-volt batteries. Note the wiring of the two batteries:

- The + from the left battery is connected to the + bus at the top of the breadboard.

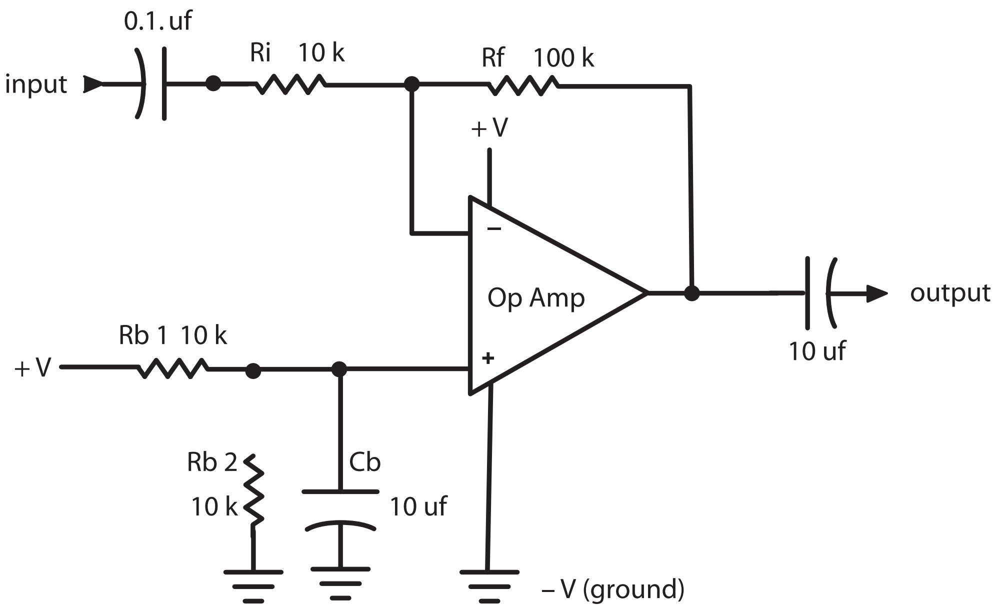


Figure 23.3 Op amp preamp, single supply.

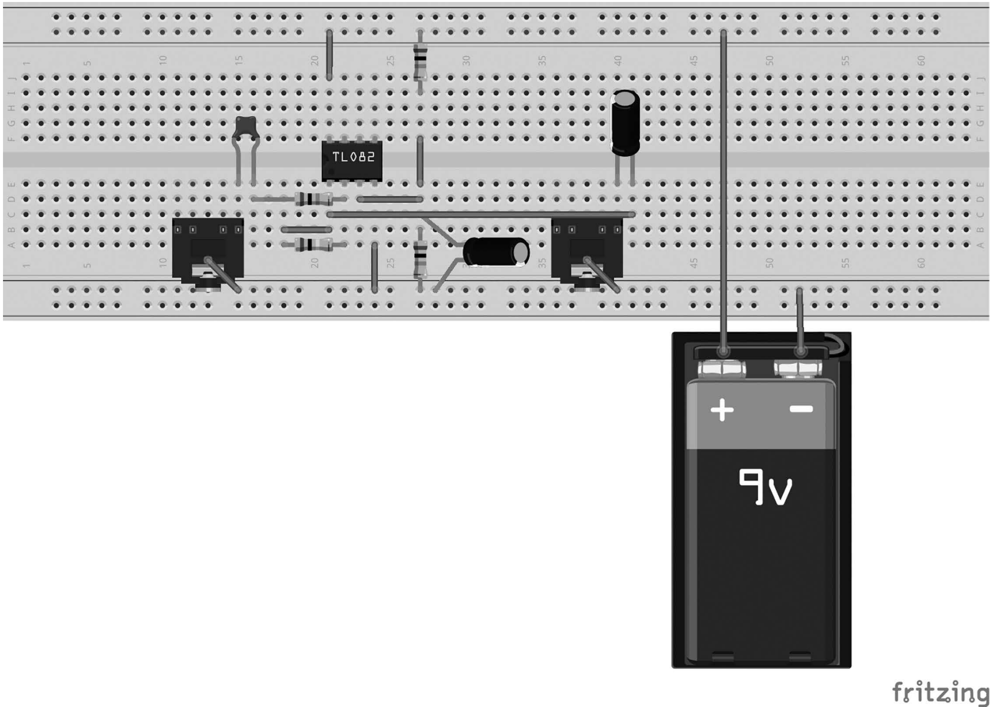


Figure 23.4 Breadboard view of single-supply preamp.

- The − from the left battery is connected to the ground bus at the bottom of the breadboard.
- The + from the right battery is also connected to the ground bus at the bottom of the breadboard.
- The − from the right battery is connected to the second bus at the bottom of the breadboard, which supplies the negative voltage when using a dual supply.

I know connecting + to − is a scary idea, but trust me, this is how you create a ground reference that lies halfway between +9 volts and −9 volts. If you move on to synth circuits, you'll find this is standard practice.

And op amps can do so much more than just amplify. . . .

A MIXER

Figure 23.6 shows a simple three-input mixer running off a single battery. You've already made a passive one in Chapter 18, but that design can't boost low-level signals. This design features a gain of 10 for any input when the pot is full open—good enough for raising contact mikes up to laptop level.

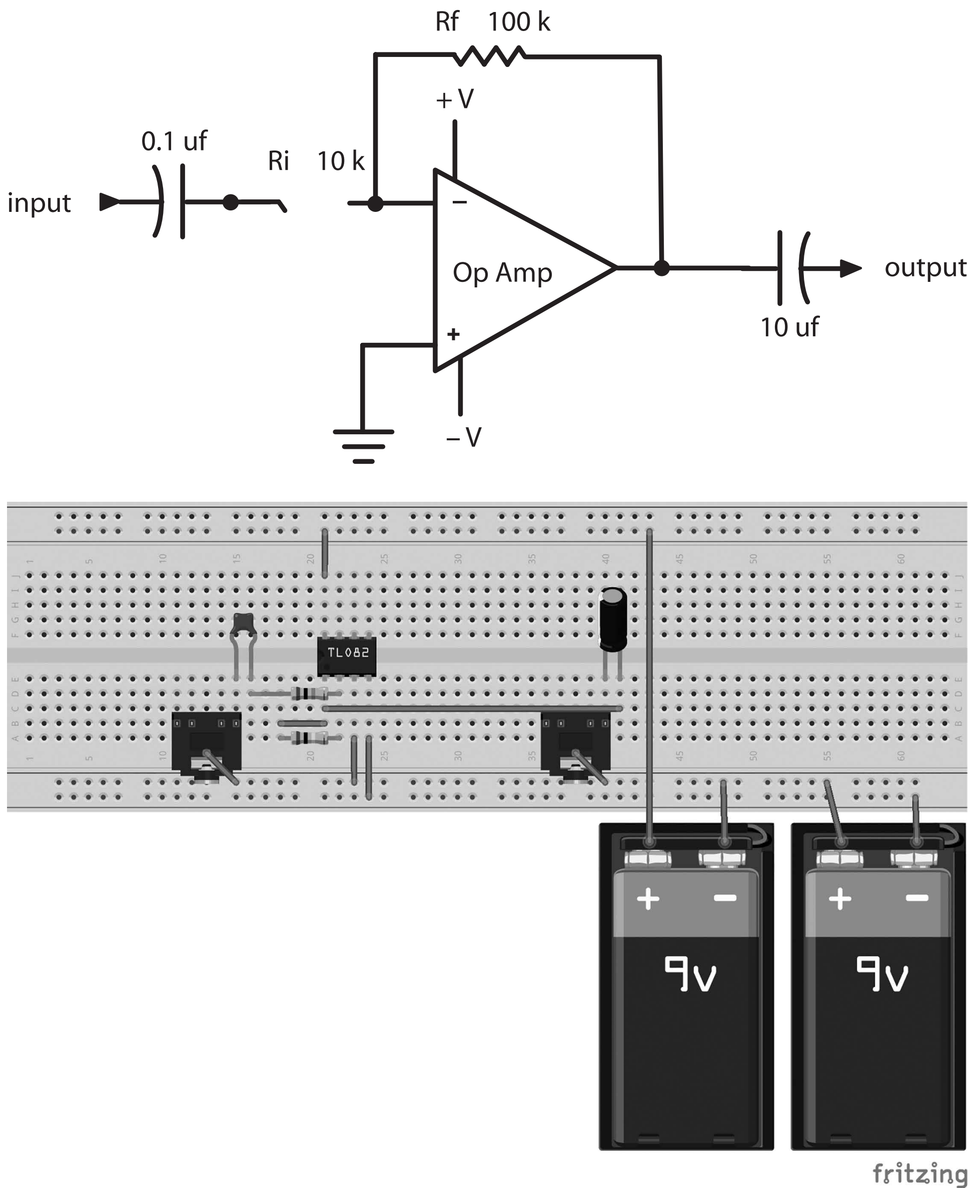


Figure 23.5 Op amp preamp, dual supply.

Want more? The Bissel Function Block in Chapter 30 includes several different op amp applications. In Chapter 31 we build neural networks out of op amps. The internet abounds with resources for op amp circuit design: not just schematics for individual circuits but full-blown tutorials that walk you up the design ladder from basic preamp to multi-mode filter. A good place to begin is with the various free "Application Notes" that semiconductor makers publish for their products. Google "LM324 Application Note" for starters.

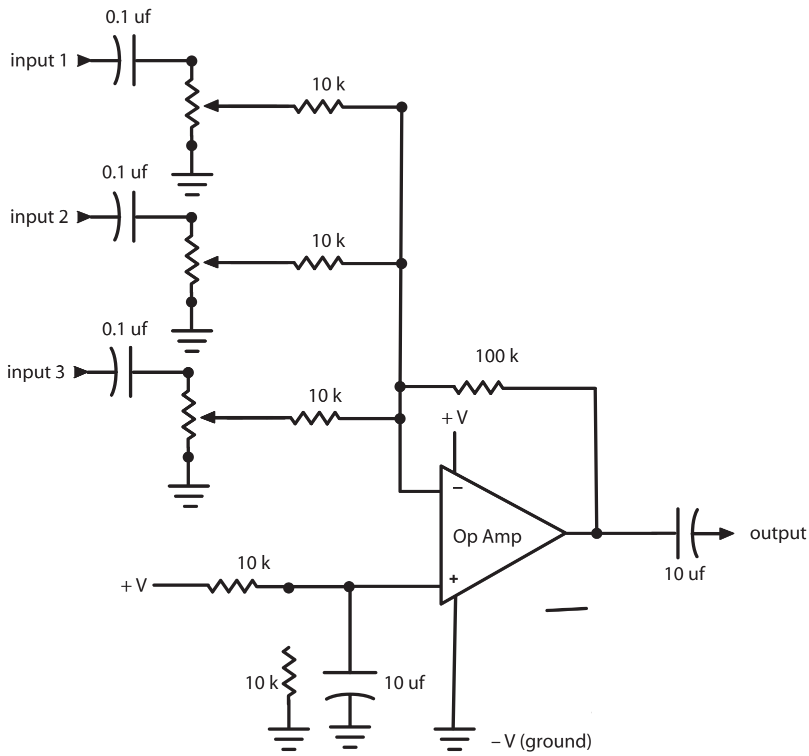


Figure 23.6 Three-input mixer.

CHAPTER 24

A Little Hacker's Amp

You will need:

- Something to amplify: guitar, contact mike, phone, etc.
- A breadboard.
- Audio Power Amplifier chip, LM386.
- Assorted resistors, capacitors, and pots.
- Assorted jacks and plugs.
- A small speaker.
- Some solid hookup wire.
- A 9-volt battery and connector or four AA batteries and a holder.
- Hand tools.
- An appropriately sized box to fit the circuit and speaker.

Whether in pursuit of a self-contained electronic instrument or some form of sound sculpture, one day you will tire of choosing between some generic mini-amplifier and a bulkier, more expensive (and potentially more dangerous) PA system. There are a number of amplifier kits available from online retailers that include the essential integrated circuit, associated passive components (resistors, capacitors, etc.), and a printed circuit board. But if you want to save a few dollars and get some more design experience, consider soldering up your own using the venerable LM386 (see Figure 24.1). At less than a dollar retail, this chip, combined with a few other components in a very simple configuration, makes a cheap but decent low-power audio amplifier. It is the heart of many commercial mini-amps. And it's almost indestructible.

The LM386 is a variation on the basic op amp we studied in Chapter 23, with on-chip gain and bias resistors, to simplify your job. The basic configuration shown in Figure 24.2 gives a gain of 20 and is best for line-level signals such as file players, computers, your oscillators, etc. By adding a 10 uf capacitor between pins 1 and 8 the gain rises to 200 (see Figure 24.3), which is more suitable for contact microphones, coils, and guitar pickups. You can add a switch on this capacitor to select high or low gain for different input sources, as shown in Figure 24.4.

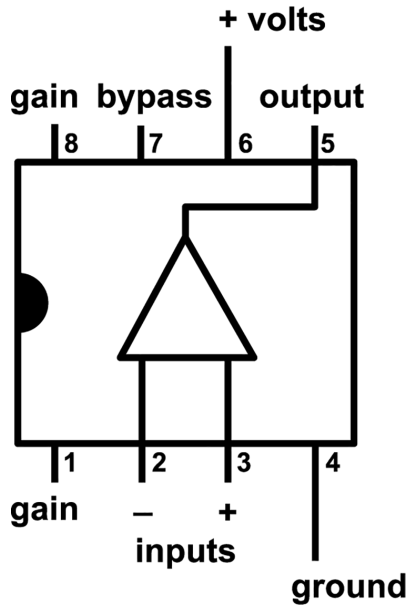


Figure 24.1
LM386 amplifier pinout.

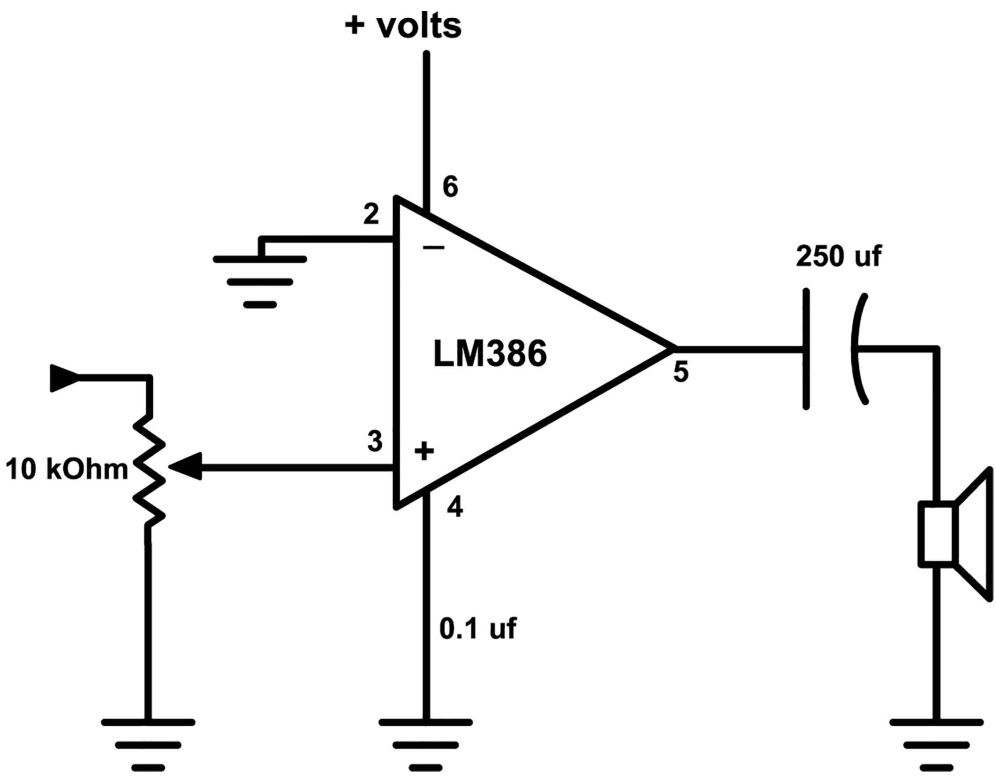


Figure 24.2 Amplifier with gain of 20.

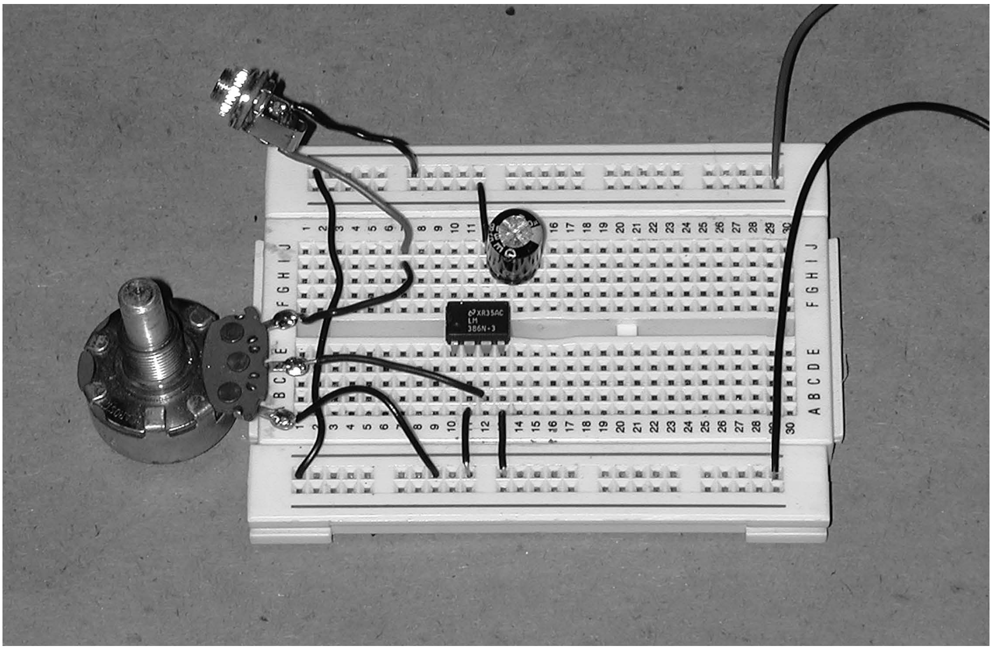


Figure 24.2 (Continued)

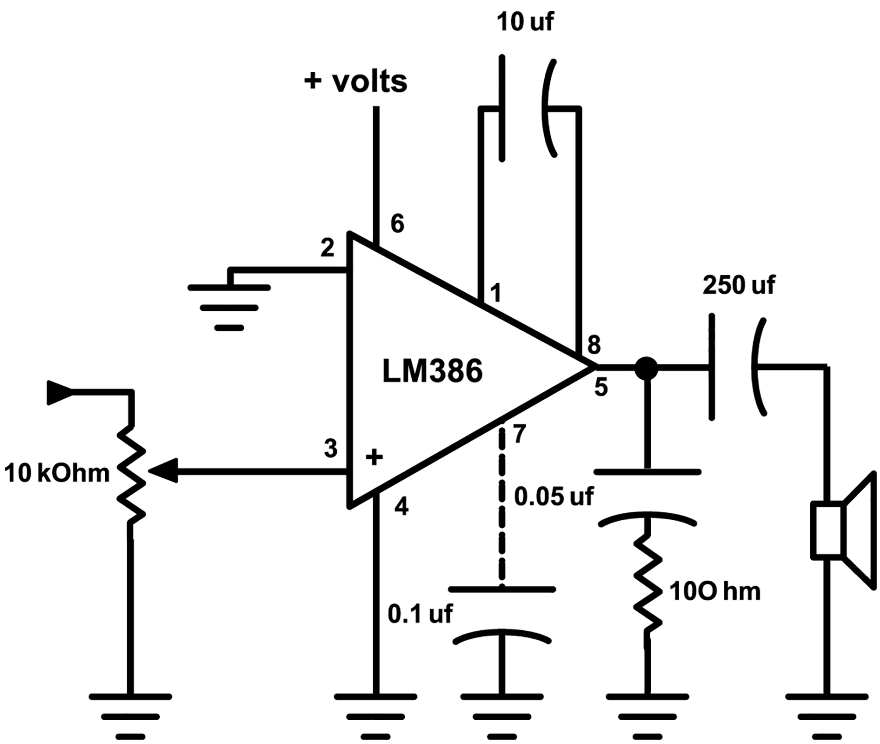


Figure 24.3 Amplifier with gain of 200.

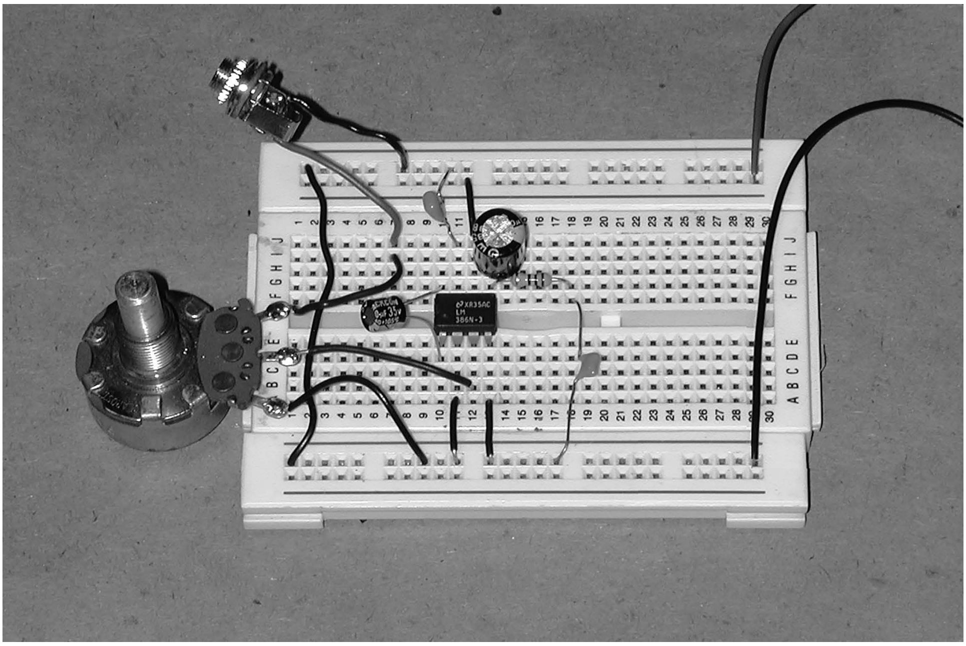


Figure 24.3 (Continued)

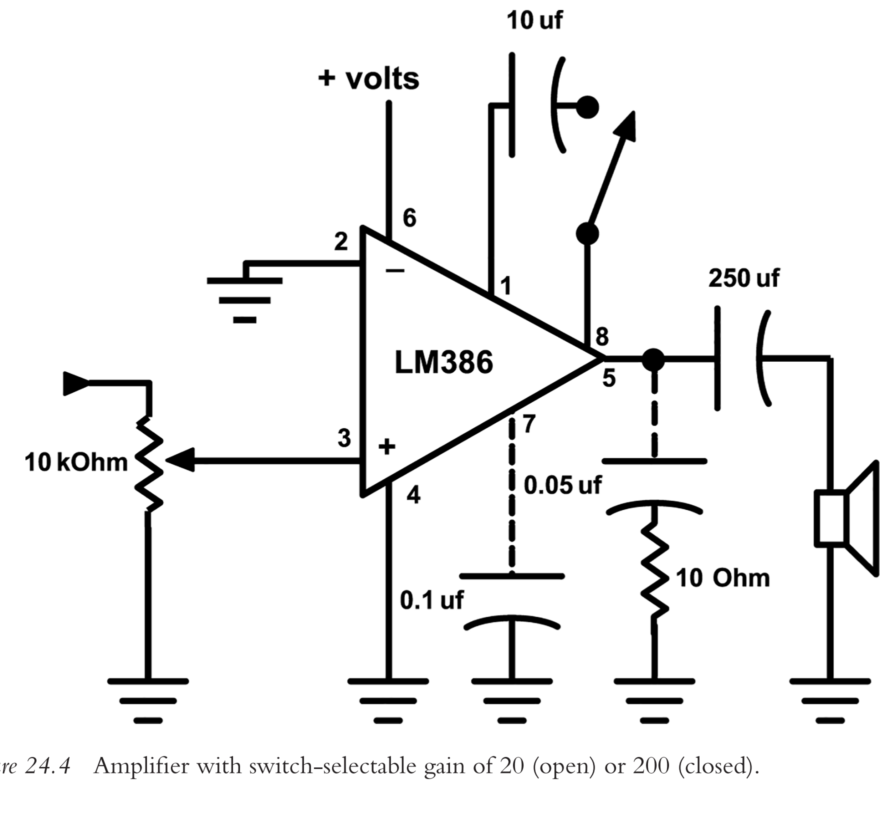


Figure 24.4 Amplifier with switch-selectable gain of 20 (open) or 200 (closed).

The + voltage from the battery connects to pin 6, and the −/ground connects to pin 4. You might want to include a power switch as well or disconnect the battery from its clip when not in use since this circuit drains more power than the others we've made. Pin 1 is also tied to ground, and the input signal goes to pin 3 after passing through a potentiometer used as a volume control. The 0.05 uf capacitor and 10 Ohm resistor shown at pin 5, and the 0.1 uf "bypass" capacitor at pin 7, are optional parts, to be added if the circuit oscillates and whines by itself. Some people add an input capacitor—typically in the range of 0.1–2.2 uf—between the volume control pot and the + input (pin 3).

One cautionary note: for all the apparent simplicity of this design, audio power amplifiers are notoriously finicky—even worse than the 4049 preamp we wrestled with back in Chapter 19. Especially in the high gain setting, this one is susceptible to oscillation. Guidelines for best performance are similar to those we followed with the 4049:

- Keep the jumper wires as short as possible—on the breadboard and between the pots and jacks and the breadboard.
- Use shielded cable to connect to the circuit, speaker cable at the output.
- Install the various bypass capacitors shown on the schematics.

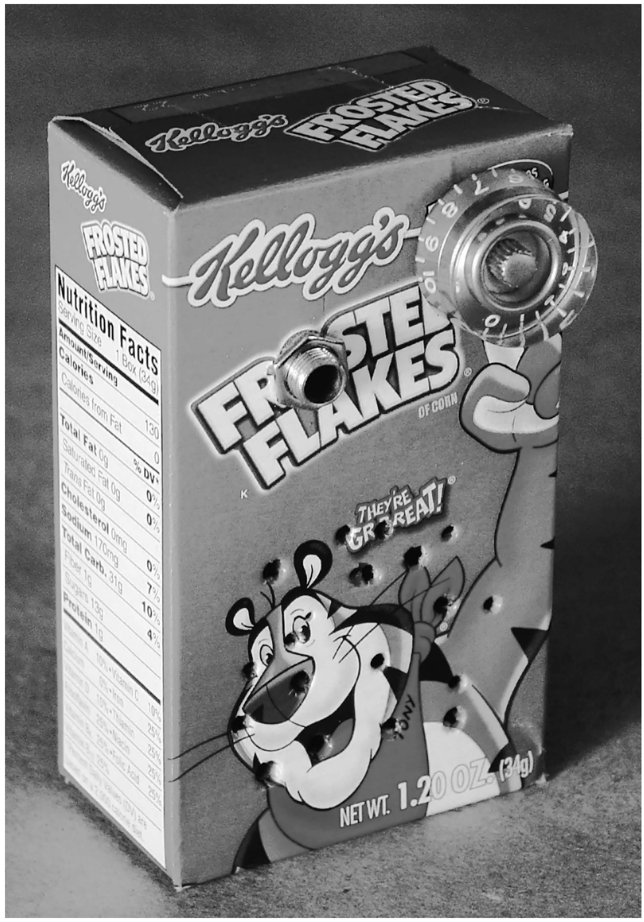
If nothing else works, consider buying a kit that includes a printed circuit board designed specifically for the amp. Such a board will do a much better job of suppressing spurious noises than a breadboard or generic circuit board such as we've used so far.

When you've finished soldering up, you can mount the circuit in a box with a speaker, like a self-contained mini guitar amp (Figure 24.5). Or leave the speaker out to keep the package smaller—you can then connect to various sized speakers to suit the demands of the occasion. Figure 1.2 in Chapter 1 shows an amp in a small tin with jacks for connecting with plugs but also on-board contact posts for quick hookups with clip leads.

This circuit puts out about 1/4 watt of audio power and can be used to drive small speakers or headphones. It runs nicely off of a 9-volt battery or a set of four AA batteries (the latter will last longer). This 1/4 watt may seem pretty puny, but it's enough to drive a piezo disk at pretty high sound levels using the backwards output transformer trick shown in Chapter 8. This amplifier can also drive small transducers, small motors (such as the vibrating motors from cell phones and pagers, also discussed in Chapter 8), or a low-power solenoid or relay. The EBow electromagnetic string driver described in Chapter 8 is basically a pickup coil connect to an LM386 connected to a driver coil extracted from a small speaker—all encapsulated in epoxy so we can't figure out exactly what else the designer threw in to make it sound so cool (but don't let that stop you from experimenting . . .).

There are dozens of amplifier chips available these days, ranging in power from fractions of a watt like the LM386, suitable for battery power, to high-power ICs intended for use in hi-fis, TV sound systems, or car stereos. I've chosen the LM386

Figure 24.5
Mini-amplifier built from LM386.



because of the extreme simplicity of the circuit design and its indestructibility, but if you want more oomph, you can do a bit of Googling to find a more suitable chip.

An amplifier is the most basic and essential tool in electronic music. You can never have too many of them. So knit yourself one whenever you're feeling bored.

CHAPTER 25

The Mumma-Tudor Ring Modulator

MICHAEL JOHNSEN AND YOU NAKAI

You will need:

- Two center-tapped audio transformers, Xicon 42TL002-RC.[1]
- Four germanium diodes, 1n34 or equivalent.
- Three audio connectors of your preference.
- Hookup wire.

David Tudor (1926–1996) was 32 years old when he began delving seriously into electronics (see essay by Nakai and Johnsen on the website). He developed a distinct style of live electronic music in which modular devices and homemade and commercial electronic gear, as well as acoustic instruments and objects, were tentatively assembled to form a compound instrument of his own design for each performance. He did not work in isolation. Exchanges of ideas and instruments were frequent between him and his collaborators. The instrument that Tudor probably used most throughout his career was made by Gordon Mumma (or William Ribbens, with whom Mumma set up his company Cybersonics): the Spectrum Transfer, a versatile ring modulator that appears in diagrams and photographs extensively from the time Tudor received it in 1965 until as late as the early 1990s. We will build a simplified version of this historic circuit.

The ring modulator (often abbreviated "ring mod") is a powerful circuit for transforming sound and yet is one of the easiest to build, if not understand. It plays a vital role in the basic kit of old-school 1950s electronic music, but it's even older than that. The ring modulator is part of a larger family of "double-balanced amplitude modulators" that goes back to electronics year zero. By the 1920s, telephone engineers worldwide were using emerging (but pre-transistor) electronic technology to develop a cheaper circuit to cram ever more signals onto their lines. The catchy term "ring modulator" refers to the daisy chain of diodes at its core.

In the 1950s electronic music adopted the ring mod, along with other bits of radio gear, to perform tasks for which they were never intended. As technology evolved, more complicated circuits performing the same function were developed, with names like "analog multiplier," but the original ring modulator uses just four diodes and a pair of audio transformers. Like the mixers we built in Chapter 18, it doesn't even need a battery. This passive character gives it unique qualities and robust charm.

Every ring modulator has three ports. Two are used as inputs and one as the output. You get to assign their roles, but you must use all three or there won't be any sound. The two input signals are multiplied together by the ring mod into an output with an enharmonic spectrum: the new tones aren't necessarily in the harmonic series, which is why the ur-sound of the ring mod is often described as "bell-like." In addition to generating new spectral components (loads of them!), it strongly suppresses the input frequencies, removing much of their original character from the output. This is different from other processors like filters, which *remove* harmonics, or mixers, which simply combine signals without changing them. While distortion pedals also add harmonics, they sound nothing like a ring modulator.

Specifically, if you input two sine waves of frequency X and Y, you'll get a mixture of two new frequencies (called the upper and lower sidebands) at the output: one will be the sum of X + Y and the other the difference of X − Y, but neither X nor Y itself will be present in the output signal. The output components are determined by an arithmetic process that doesn't preserve the ratios of the tempered scale, so using any two pitches from the piano will produce an output that falls between the keys, hence the term enharmonic. For example, using 440 Hz middle A4 at one input and 392 Hz G4 at the other will produce pitches at 440 + 392 = 832 Hz (a very sharp g#5) and 440 – 392 = 48 Hz (a very flat g1). The only exception is when the two inputs are the same or related by octaves. In that case the output just jumps up by octaves: using A4 at both inputs gives A5 at the output (440 + 440 = 880 Hz, 440 – 440 = 0).

The ring mod's simple arithmetic turns input sounds inside out and creates new ones. You might be tempted to "hear" the arithmetic by using simple oscillator inputs, but something familiar like a voice at one of the inputs will produce a less rational, more clangorous result—familiar as the voice of the malevolent robot in many a vintage science fiction film. If the input pitches are related by octaves and one of them is a sine wave, you can elegantly shift the spectrum of the complex wave but retain a melodic line. Both Mumma and Stockhausen were fond of this method. If both inputs are complex, the results get noisy and unpredictable. Slow glissandos create a rush of swirling heterodynes. The ring mod can also function as a gate or tremolo, for panning effects, as a narrow band filter, or as a frequency doubler.

BUILDING THE RING MOD

The ring modulator has only six parts: four germanium diodes and two center-tapped audio transformers. Companies that supply parts for guitar pedals often sell germanium diodes. 1n34 is a common type, but others are fine, too. A good transformer for this project is the Xicon 42TL002-RC (source in notes).[2]

You'll need some wire and jacks of your choice (although for prototyping you could just clip it all together with alligator leads or stick the parts on a solderless breadboard). However you build the ring modulator, make sure the diodes point in the right direction (Figure 25.1)! It sometimes helps to redraw the diode ring "folded up" into a lattice as in Figure 25.2: follow the path around the ring to prove to yourself that your mechanical assembly matches the schematic.

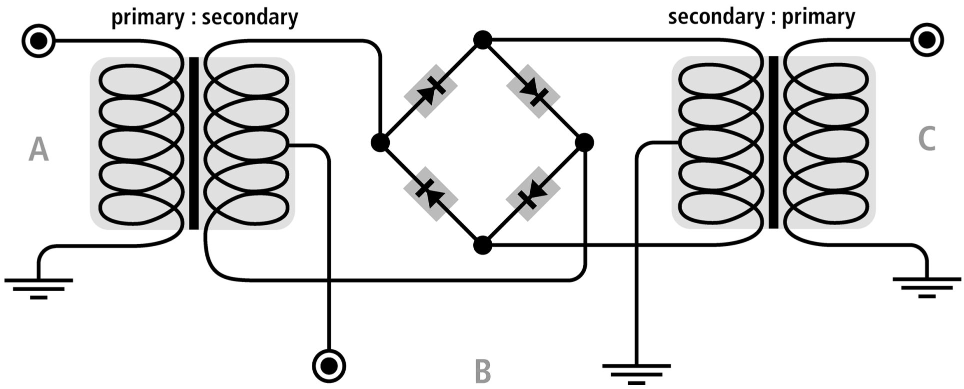


Figure 25.1 Ring version of ring modulator.

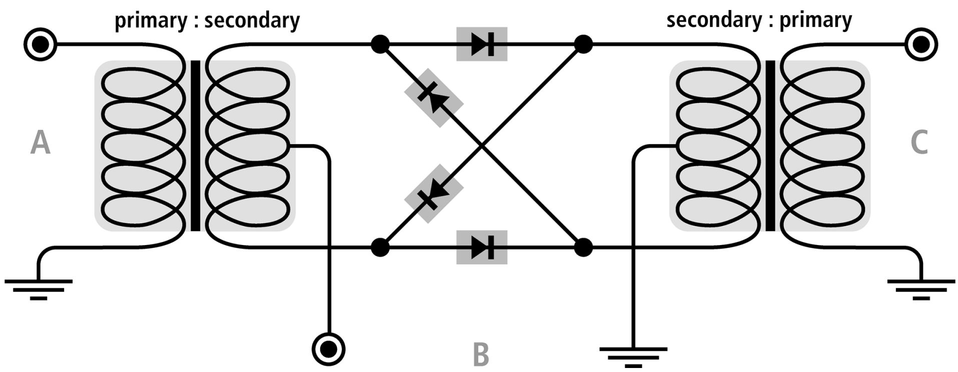


Figure 25.2 Lattice version of ring modulator.

Begin by forming the ring of diodes. Look for the dark bands on the diodes to verify their orientation. Consider twisting the leads together to make the assembly more rigid before soldering. If you're working from the "lattice" drawing, make sure the two "crossing" diodes don't make electrical contact. The finished ring will have four nodes at its corners to connect to the transformers. Solder these four points to holes in your perfboard or insert them into your breadboard on separate buses.

Now, orient the two transformers according to the drawing. The sides with *three* wires should face the diodes. Notice the mirror symmetry of the schematic. We'll explain more about transformers later. Following the drawing carefully, attach each of four nodes in the diode ring to the correct wires on the transformers. Finally, use two-conductor cable to attach your three jacks to the remaining six wires on the diagram. Notice that one jack will attach to both of the transformers. The three ground wires can be connected to one another if you wish, but it's not necessary. Double-check your work according your construction method, wiggling socket connections or solder joints.

If you want to understand how the circuit works, see "How the Circuit Works," following.

TESTING AND CALIBRATION

The oscillators we've built so far in this book generate square waves and triangle waves, which have rich overtone content. But the easiest way to test your ring mod is using two sine wave signals. Hardware sine wave oscillators can be a bit tricky to build, but the waves themselves are easily available in phone apps or on your computer. You can cobble them together in Pure Data, Max, or SuperCollider, and many DAW applications like Audacity include them.

It's best to tune your two source pitches far apart. Try line-level sines of around 100 Hz and 1,000 Hz. Now listen to the output. You should hear two new pitches: one higher than both inputs (this is the sum frequency) and one below (the difference tone) plus a very slight bleed from the sines you put in. If you have access to an oscilloscope, the trace should look like the top image in Figure 25.3, with symmetrical

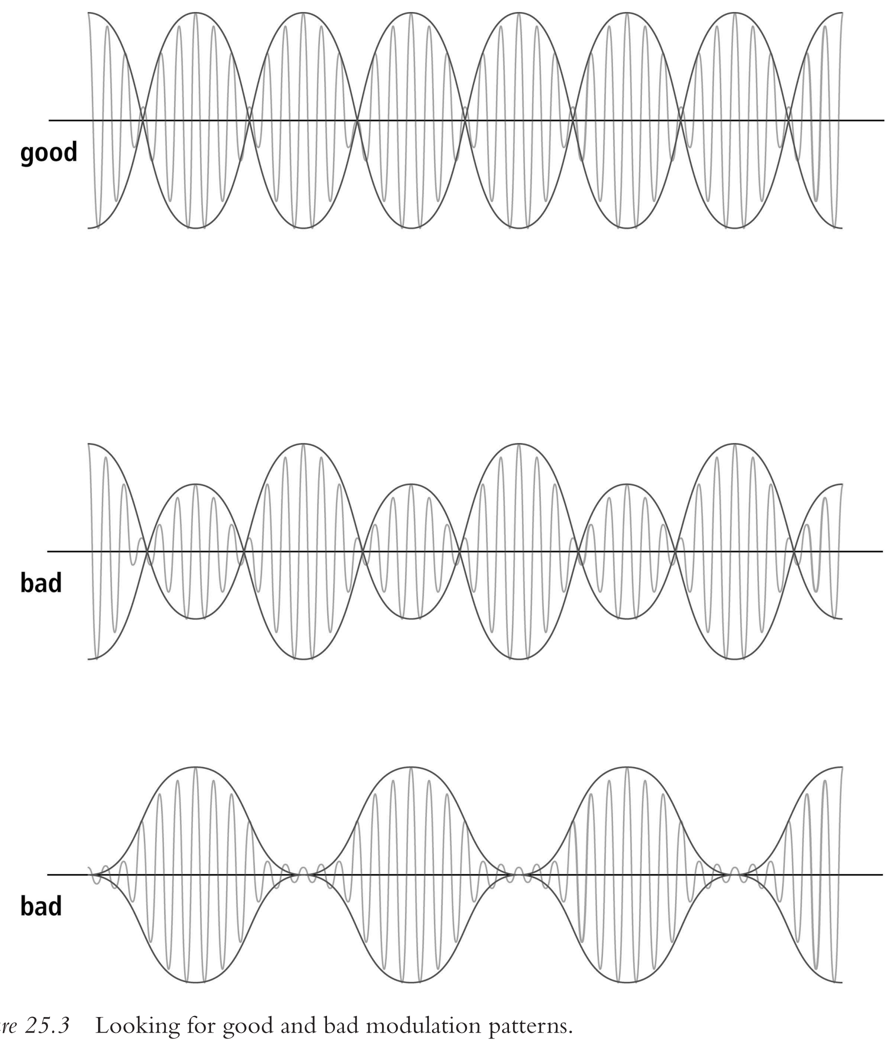


Figure 25.3 Looking for good and bad modulation patterns.

lobes from left/right and top/bottom. If all seems right, start fooling around and sweep the frequency feeding one or both inputs. If you can clearly hear one of the input pitches (or if the trace looks like either of the bottom two images), go back and check your work. You may need to lower the level of one or both inputs to minimize distortion.

SUGGESTED USES

- Use filtered noise at one input to create rasping sounds and hoarse coughs.
- Add attenuators at the inputs for controllable distortion.
- Since it shuffles an input's harmonics, the ring modulator is great at making bell tones and other metallic sounds, especially if one of the inputs is a sine wave.
- Use a slow triangle or square wave at the B input to make soft tremolos (triangle) or a gate (square). Try connecting and disconnecting a battery to gate the audio signal or add a diode in series with one input.
- The ring mod is symmetrical. You can apply signals to any two ports and listen at the third. The difference is that one of them is DC coupled, so if you wish to apply hard, gating control signals, like batteries or square waves, use B as an input.
- Add a Double-Pole Double-Throw switch to switch input signals between ports B and C.
- The ring modulator excels with real-world signals like microphones and pickups. Two live instrumentalists at the inputs will only produce an output when *both* play, and the output won't sound like either of them. Gordon Mumma's *Diastasis, as in Beer* (1966) uses a similar idea.
- Use a low-pass filter at the output to remove the upper sidebands if the output sound too rich, distorted, or noisy.
- Experiment with feedback.
- Send one signal into both inputs for frequency doubling.

HOW THE CIRCUIT WORKS

1. The transformer isolates and balances.

Transformers are made from a pair of coils of wire, insulated and wound around a shared core. They get very close but never touch each other. An alternating signal (AC) applied to one winding (called the primary) induces a current in the other, secondary winding while isolating it from direct current (DC). The ratio of turns in the two windings causes the signal voltage to grow or shrink.

In the ring modulator, we use a secondary coil with a third contact (called a center tap, at its exact mid-point) to enable a clever trick (Figure 25.4). Since there's no electrical contact between the two sides, we can choose our ground reference at will. Reversing ground and signal at the ends of the secondary inverts the signal from

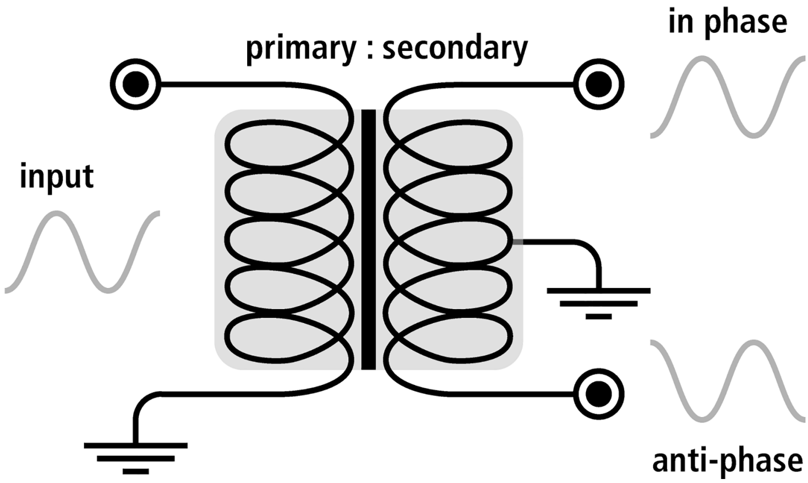


Figure 25.4 Transformer with balanced (or differential) output.

the primary. But if we ground the center point instead, we have both versions at once: in phase and anti-phase. Now, in addition to isolation, the signal is also "balanced" or differential (in phase and anti-phase), like the output from a DI direct box. Balancing is essential to how the ring mod works.

2. The diodes switch softly.

We usually think of diodes as "one-way streets" for electrical flow. But they also function as switches: they conduct only when the applied signal exceeds their DC threshold voltage, usually 0.6 V. Sure enough, the pairs of diodes in the ring use one signal to turn both halves of the other on and off. By limiting the signal voltage to very near threshold (called the "knee" of the diode), we can get a smooth action instead of abrupt switching: more like turning up and down than on and off. Old-fashioned diodes made of germanium are especially good in ring mods because they have soft knees (yup, that's what they're called) while the windings ratio of our transformers is chosen to reduce the signals so that the diodes operate smoothly.

3. Together, they multiply.

If you've followed so far, it's not a big leap to see that a signal applied at A is isolated, turned up and down and inverted by the signal at B, then "induced" to the output at port C. Mathematically, this is a multiplication of two signals. Because the circuit is symmetrical and passive, you can use any two ports as inputs and listen at the third.

If you don't understand this, don't lose sleep, just keep experimenting until you internalize multiplication.[3]

NOTES

1. Available from Mouser Electronics, www.mouser.com/ProductDetail/Xicon/42TL002-RC?qs=sGAEpiMZZMv0IfuNuy2LUWC0DlW3ceP1ZEvrun71JOk%3D
2. This is not the same model transformer we used for the piezo disc driver circuit in chapter 8.
3. For more technical information see Harald Bode, "The Multiplier-type Ring Modulator." *Electronic Music Review* Vol. 1 (January 1967). P. 9ff, https://ubutext.memoryoftheworld.org/emr/periodicals/EMR1.pdf

CHAPTER 26

Paper Circuits

PETER BLASSER

You will need:

- Five bc546b transistors ("−" symbol on circuit diagram).
- Five bc556b transistors ("+" symbol on circuit diagram).
- CD4015 Dual 4-Bit Static Shift Register.
- 9-volt battery connector.
- Resistors, 1/8 w: 10 k, 22 k, 47 k, 100 k, 220 k, and 2.2 mOhm.
- Four capacitors, 0.01 uf (10 nf).

Paper circuits have two mirrored sides, printed adjacent, with a fold line in the middle. A piece of card moistened with glue is inserted between the folded sides. The top side indicates where to pierce by needle and where to then thread components, such as resistors, capacitors, and chips. On the underside, lines trace how to weave the component leads together for soldering.

It is a quick way to craft a sketch of a circuit, but it also serves as a platform for studying crossed or circuit-bent nodes. Component connections form metallic islands, usually kept insulated from each other. The paper leaks some electrons between these islands, which may influence the purity of electronic sounds. It is a noisy and finicky alternative to more conventional circuit boards, a radical technique I first read about on the website of commonsound, a noise-music collective in St. Louis (around 2001).

In contrast with machine-made PCBs, a unique goal in laying out paper circuits is to facilitate a pleasant crafting experience. Because the interconnecting wires are woven together on the back side without any reference to grid geometry, component placement prefers a flowing orientation, often eschewing right angles. When a transistor, resistor, and capacitor link together as an oscillator, it is a union of three, and the designer can naturally explore triangular layouts (Figure 26.1). The goal in using odd angles is not fragmentation but a tightly spaced mesh.

Another unusual design implication of paper circuits arises from the organic nature of the paper, which spreads electronic influence among nodes. The "official" circuit can be described in the schematic, but, once built, the paper (which conducts differently depending on seasonal weather) introduces many unofficial connections

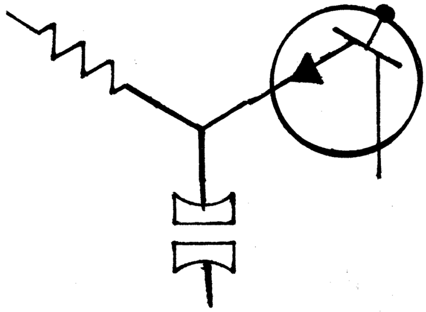


Figure 26.1
Triad*:* three components come together, suggesting a triangular layout.

between each of the nodes. One can think of the paper circuit as a sensor composed of a net of interrelated signals.

A fiberglass circuit board allows different impedances to sit adjacent, but in a diffusive medium such as paper, placement is important. Components are arranged geographically; their orientation and proximity on the paper are important.

Higher frequency circuits are more sensitive to the paper's special properties, and radio circuits may become chaotic or unstable. The best designs work with the medium's natural tendency to mix signals. My earliest paper circuit design, Mister Grassi, is a closed loop: each component feeds into another component, either an oscillator or digital chips configured to warps the signals of the oscillators. The closed loop nature is particularly suited to a leaky medium such as paper circuits because any spurious noise gets absorbed back into the loop. It's hard to tell where errors come from because everything's already affecting everything else.

RUNGLER

When Rob Hordijk explained his Rungler concept to me in 2009, I realized that it described Mister Grassi as well as a host of other electronic musical instruments that generate chaotic, living sounds. The Rungler mangles sound by combining analog oscillators and a primitive digital memory. I am interested not only in the sounds it makes but also the philosophical value of combining analog and digital. Each realm misinterprets the other; this is the same sort of mechanism that generates sublime poetry in translation from one language to another.[1]

The digital memory is a simple shift register from the venerable 4000-series CMOS logic that populates this book. This concept is elegant enough that eight bits of data suffice to generate interesting patterns. A shift register takes two inputs, data and clock. The data moves through the register one slot at a time, at the sampling rate of the clock. For these two inputs, a Rungler uses two oscillators. In turn, these oscillators receive frequency control from the combined outputs of the shift register, forming a double feedback loop.

The Parrot Pleaser, an electronic circuit composition by Paul DeMarinis (1975),[2] uses a 4015 shift register as scrambler, controlling an oscillator from the LM3900 data sheet circa 1972.[3] Rob Hordijk prefers the 4021, which is similar to 4015, but with eight successive stages instead of two blocks of four. In his Benjolin synthesizer, the Rungler has two current-controlled oscillators, employing the stable and precise SSM2164 device. With oscillators and registers, both these instruments enclose a loop around phase uncertainty.[4]

Mister Grassi contained three compact Runglers, each employing simple transistor oscillators and the 4015 device. I have excerpted a single section here, titled "Rungling." The layout drawing in Figure 26.2 is available on the website as a 1:1 TIFF file—print this out first.

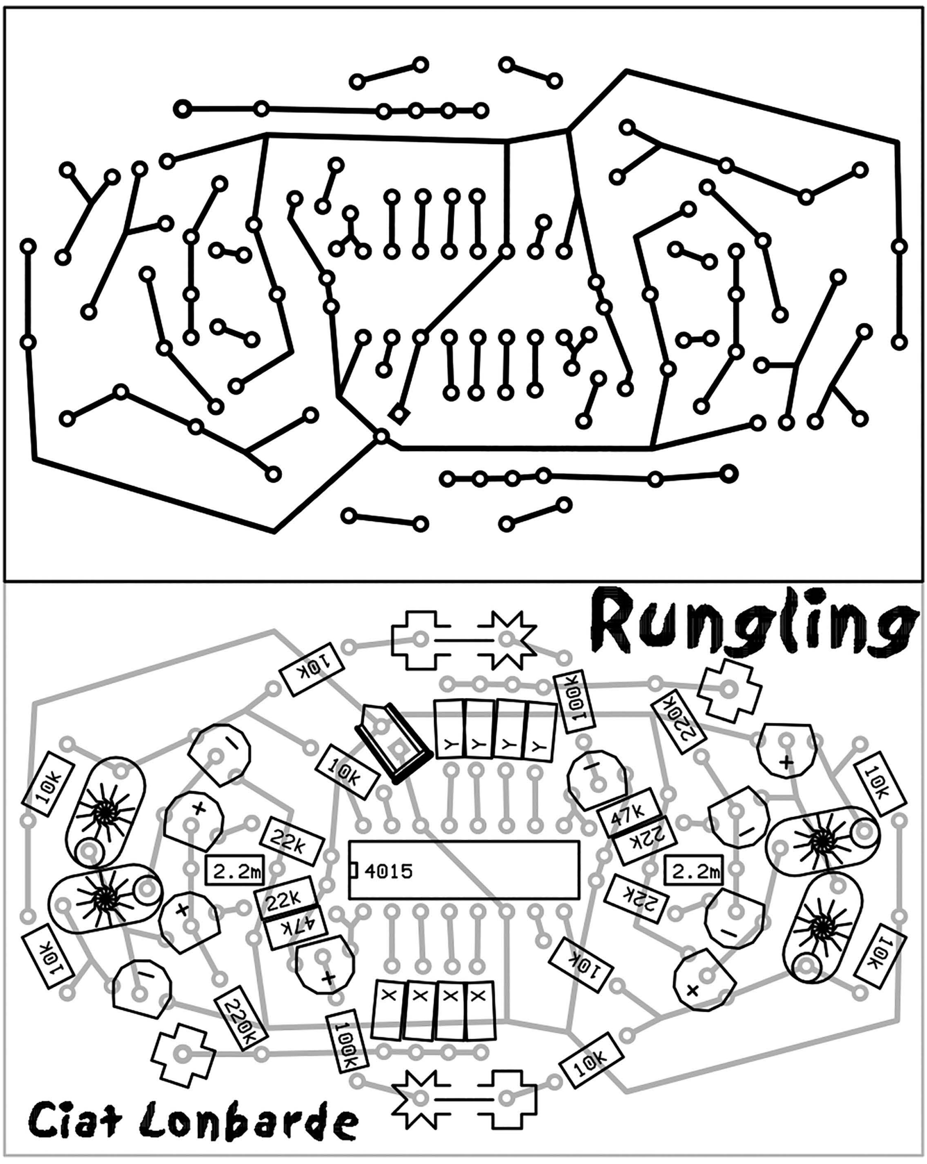


Figure 26.2 Rungling paper circuit layout.

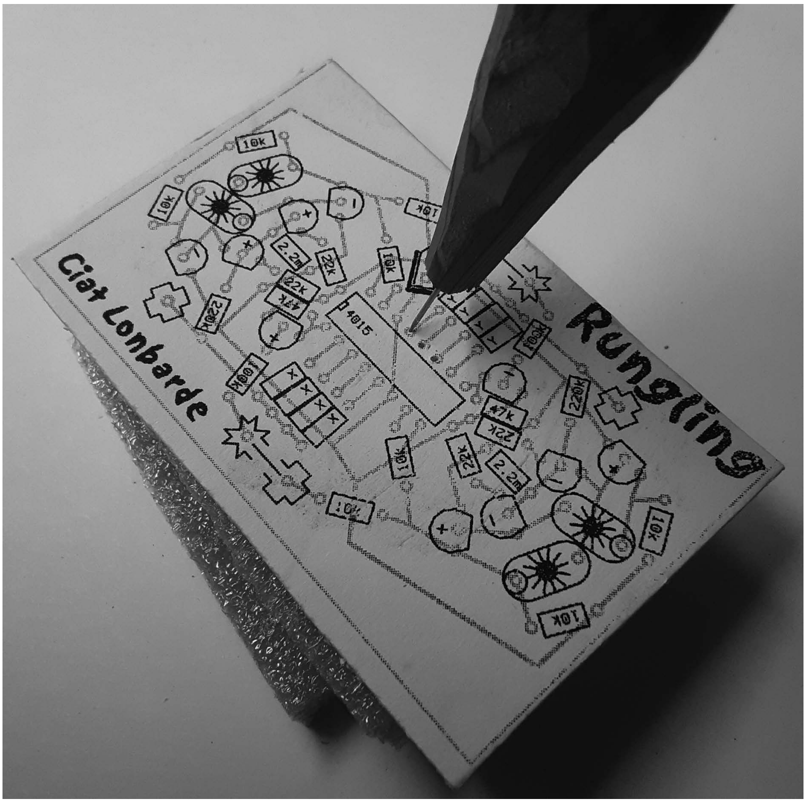


Figure 26.3 A paper circuit: cardstock and glue sandwich. From the top side, pierce each hole with a bodkin.

After you print it, fold it along the middle line and cut a rectangle of cardstock a little bigger than the circuit. Also cut a flat piece of cardstock to squeegee glue on either side of the circuit cardboard, one side at a time. Once the sandwich is formed by inserting the cardboard inside, clamp it to dry between two pieces of wood or under a stack of books. Once the glue dries, pierce each hole with a bodkin, which you can make by mounting a sewing needle in a wooden handle (Figure 26.3). Now it is ready to be stuffed with the components; refer to the parts list at the head of the chapter for symbols (Figure 26.4).

Stuffing paper circuits differs a bit from a traditional circuit board. Instead of placing all the same components at once and going through the list from resistors to chips, first insert all the different components in a small geographical area. That way you can weave their leads together and solder one node at a time. Remember to match the orientation of the chip to its legend; likewise, the flat face on each transistor must match its printed outline.

For the four capacitors, marked with starbursts, you may pick values to change the range of the circuit. I typically use 0.01 uf (10 nf). Smaller values will make the circuit pitch higher, larger for lower values. You customize the mangled data by picking

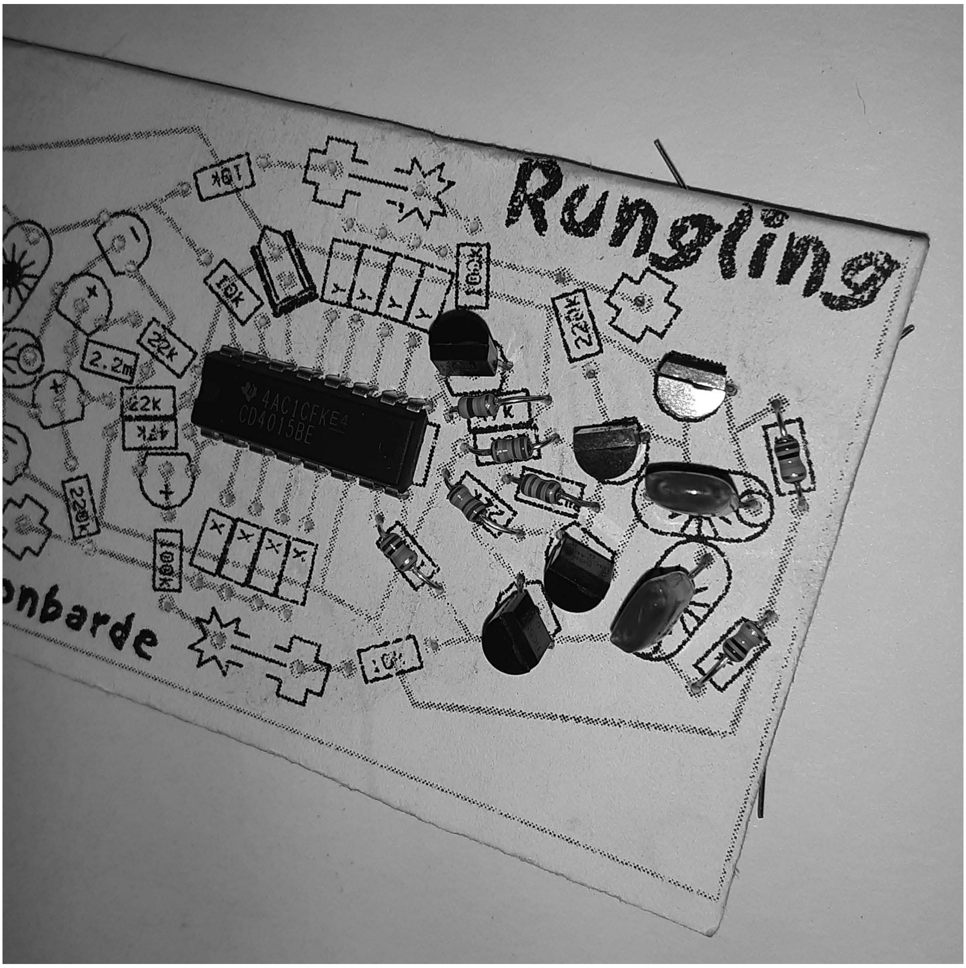


Figure 26.4 Stuffing the paper circuit with components according to its legend.

different "X" and "Y" resistors; no two Runglers have exactly the same data patterns, although they all share the same general flavor. Fill the "X" bank with a range of different values such as 10 k, 22 k, 47 k, and 100 k, and also distribute them in the "Y" bank.

An obelisk symbolizes the 9-volt power input. Its pointy tip is positive, or the red wire from the battery snap, and its flat base is ground, or the black wire. Note where the ground trace extends, as it is needed for most connections to external devices. Connect the longer power traces with a length of bare copper wire. Watch out for bridges—unintentional solder, or wire connections between nodes (Figure 26.5).

The special cross-to-star jumpers open and close the Rungling loops. The jumper contains marks for square wave output and trigger input as cross and star respectively. Start with a wire jumping each so the circuit triggers itself in a closed loop. Signals exist at any point; there is no official output but a range of detection options. Jamming an AM radio with Rungling allows spatial dynamic control simply by moving the radio

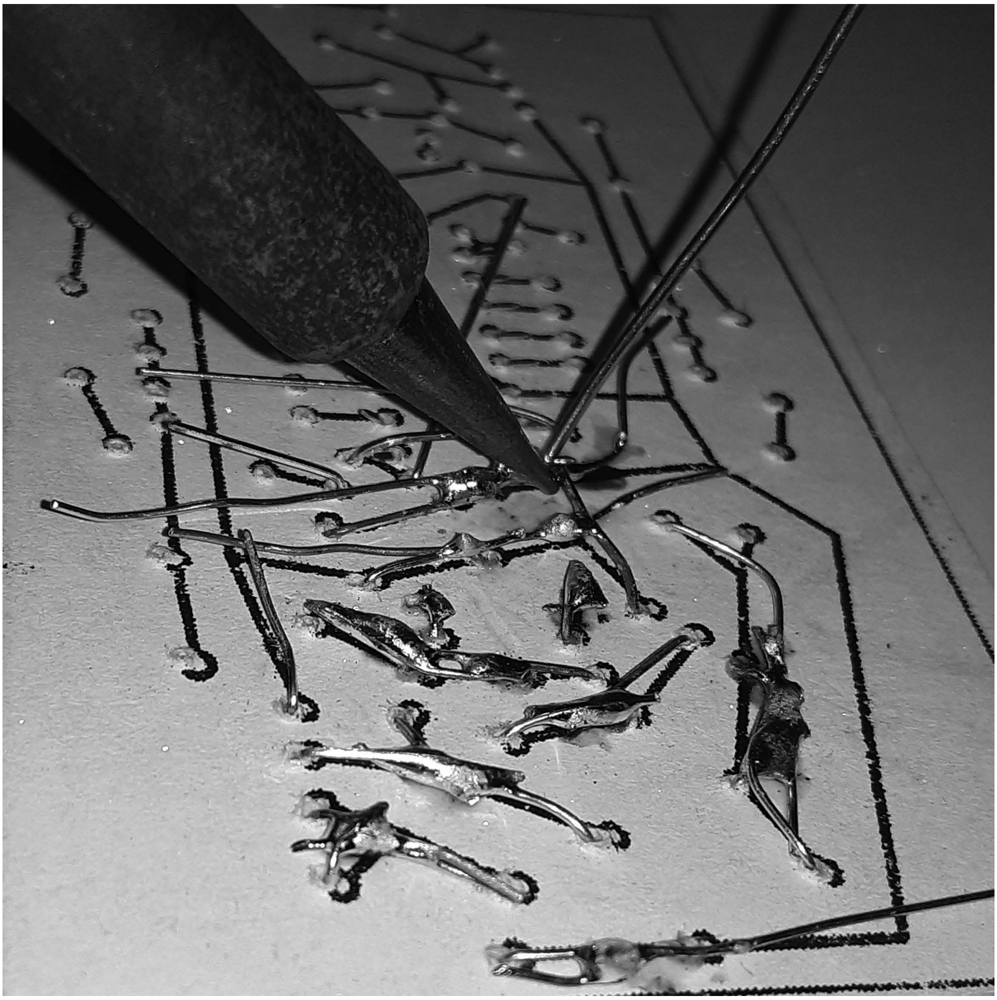


Figure 26.5 Soldering the leads on the bottom, following the traces. The islands in the foreground are nodes.

around. The silicon components, transistors, and chips emit wide spectrum switching noise, especially detectable by AM circuitry (Figure 26.6). If you touch Rungling on its trace side, enjoy the bent sound flavors, but keep metal away so not to burn transistors out by full short circuits.

A single cross marks the mangled digital voltages of each Rungler. If you view the signal here with an oscilloscope, you should see a chaotic, quilted pattern. To connect to an amplifier, take this output through a 0.1 uf capacitor. Trace ground from the battery connection to complete the circuit. It should sound like the chaotic data stream of an old modem. If the circuit passes this test, you can open up the loop jumpers and make the two ends into touch nodes; your finger hence becomes the trigger (Figure 26.7).

Figure 26.6 Jamming an AM radio. Gentle fingering on the nodes will further mangle the sound.

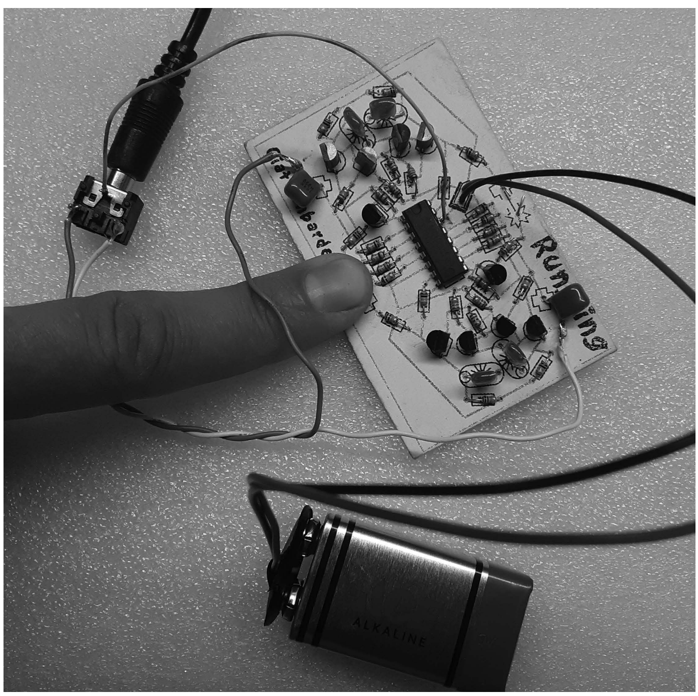


Figure 26.7 Connect ground to a stereo mini jack, and to left and right couple each cross output through a capacitor. Open the loop by cutting the special jumper, then use your finger to trigger it.

NOTES

1. In Rob's own words:

   Feedback of the output voltage to the rates of the two sources implies a myriad of possible sequences with both a pattern on the amplitude axis and a pattern on the time axis. I proposed the name Rungler for this kind of circuit as it had no name, and it is interesting enough to give it a name. And this name now seems to be widely accepted. . . . So, in essence a Rungler circuit is an open ended circuit that can be expanded and added to in many different ways. In this sense "rungling" is more a technique instead of just a particular circuit. The non-linear feedback from the DA, where the shift register and DA create the non-linearity, assumes "chaotic" possibilities on a meta-level over the actual shift register contents, but "statistics" are also a part of the equation. It can also be approached in the sense of "behaviour."

2. https://pauldemarinis.org/Circuits.html
3. www.ti.com/lit/an/snoa653/snoa653.pdf
4. http://electro-music.com/forum/post-281728.html

CHAPTER 27

Rule the Airwaves
Build a Radio Transmitter

BRETT IAN BALOGH

You will need:

- A blank, single-sided phenolic PCB.
- Six 1/2-inch-square PCB "chiclets."
- One 1/2 watt, 10 kOhm resistor.
- One 1/2 watt, 470 Ohm resistor.
- One 1/2 watt, 27 kOhm resistor.
- One 1/2 watt, 330 Ohm resistor.
- One 10 uf capacitor.
- Two 0.01 uf capacitors.
- One 10 pf capacitor.
- One 4–34 pf trimmer capacitor.
- Some 26-gauge magnet wire.
- One LED.
- One BC337 NPN transistor.
- Some 20- to 22-gauge stranded hookup wire.
- One mono (tip sleeve) audio plug to fit your mixer/amp input.
- One 9-volt battery snap.
- Soldering iron and hand tools.
- An FM radio.
- Trimpot tool (optional).
- Clip-on heat sink (optional).

INTRODUCTION

Since about 2005, artist, media theorist, and critic Tetsuo Kogawa has maintained a website where he has published his design for a simple FM radio transmitter.[1] Its fidelity and stability are not suitable for "professional" broadcast: the frequency drifts and is easily affected by the presence of human bodies, metal objects, etc. This is not to say that you cannot get a clear signal on an FM radio receiver, but the idiosyncrasies make it performative in a way that a more sophisticated transmitter is not.

This transmitter has a range of about 10 meters. Because it operates at very high frequencies, I do not recommend building this circuit on the typical prototyping breadboard used elsewhere in this book. For better results we use a technique called "Manhattan style." We take blank printed circuit material as a substrate (ground plane) and glue smaller pieces, called chiclets, on top to create islands insulated from the substrate. Make sure the components are soldered as close to the ground plane as possible—long leads will become unwanted antennas that will degrade the performance of the circuit.

STEP 1: SUBSTRATE

Printed circuit board blanks come in basically two flavors, depending on the material they're made from: FR-1 is paper laminated together with a phenolic resin, coated on either one or both sides with a thin layer of copper; phenolic is brittle and is best cut with a saw, as opposed to shears, to prevent cracking. The other flavor is FR-4, a fiberglass-epoxy laminate; do not cut this with a saw, as it produces very abrasive and irritating particles, but only a PCB shear or snips. Like the FR-1, it is coated on one or both sides with a thin layer of copper. In either case, single-sided boards (copper on one side) are recommended for this project.

First cut the circuit blank to size. I recommend a 3-by-3-inch piece as the substrate. Then cut six 1/2-inch-by-1/2-inch chiclets. Put a dot of super glue on the non-copper side of each chiclet and glue them to the copper side of the substrate in two horizontal rows of three with a 1/2-inch separation between each chiclet (Figure 27.1). Wait a few minutes while the adhesive sets.

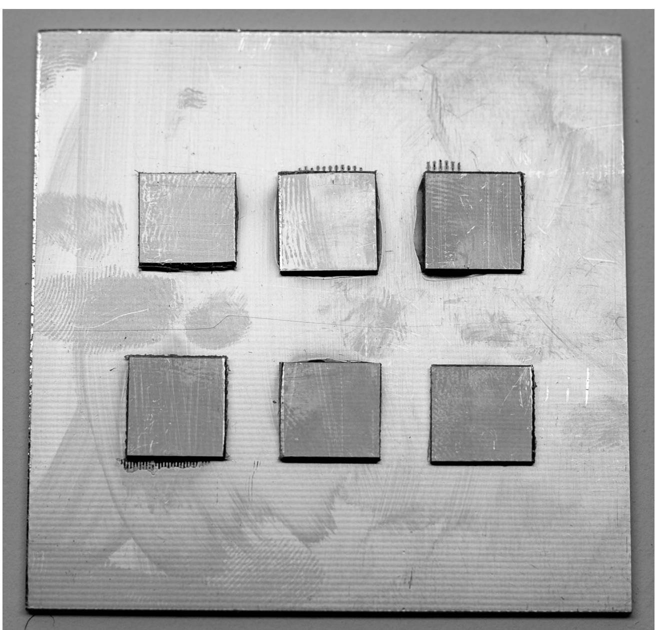


Figure 27.1
Chiclets glued to ground plane.

STEP 2: TINNING

Buff the copper surface with fine steel wool until it's shiny and free of corrosion (which impedes soldering). Melt a blob of solder onto each chiclet by heating the chiclet with the soldering iron and feeding the solder into where the iron touches the chiclet's surface. The pool of solder should be flat (Figure 27.2). You do not have cover the entire surface of the chiclet. If the solder looks like a ball sitting on top on the chiclet, heat it up more with the iron so it flattens out. It also helps to hold the iron so that the side of the tip is in contact with the chiclet (more surface area = more heat transfer). Number the chiclets, keying them to the text. We follow the numbering convention for an IC, where pin 1 is at the lower left and the other pads are numbered sequentially in a counterclockwise manner so that the upper-left chiclet is number 6.

STEP 3: RESISTORS

Now start adding components, beginning with the resistors. Take the 10 kOhm resistor (Black, Brown, Orange) and bend one of the leads with the needle-nose pliers like Figure 27.3.

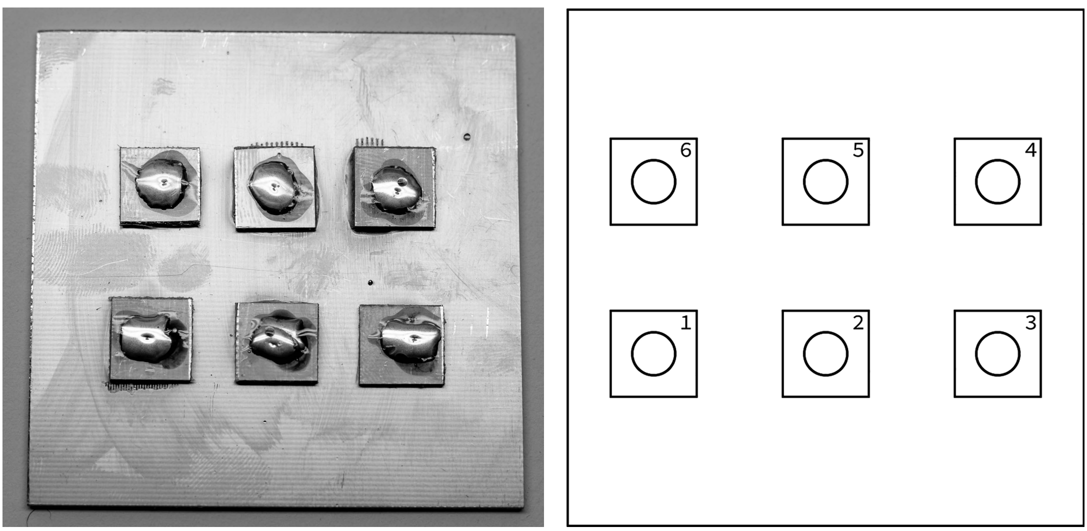


Figure 27.2 Solder blobs on chiclets, numbered.

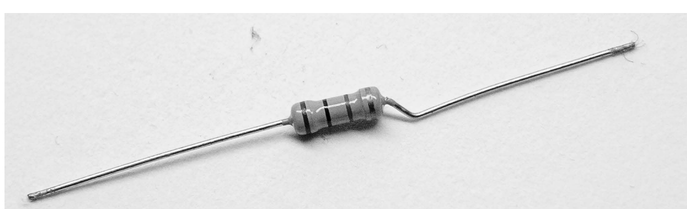


Figure 27.3 Bent resistor.

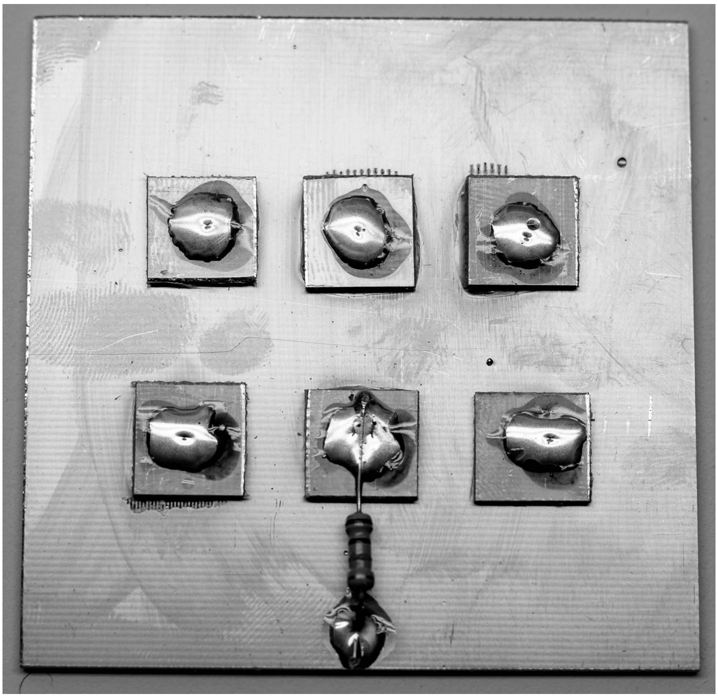


Figure 27.4
10 k resistor soldered in.

Tin both leads. Position the straight lead on top of chiclet #2 and press it into the pool of solder as you heat it with your iron ("tack soldering"). The solder should melt within a few seconds; pull back the iron and hold the component as still as possible until the solder sets. Next, solder the *bent* lead to the ground plane (make a little solder puddle first). Now, return to the first lead and touch up the soldering if it isn't solid already. This resistor now makes a connection between chiclet #2 and the ground plane (Figure 27.4). Use your diagonal cutters to cut the excess leads.

Do the same with the 470 Ohm resistor (Yellow, Violet, Brown) between chiclet #3 and the ground plane.

The 27 kOhm resistor (Red, Violet, Orange) connects chiclets #2 and #5. Lay it between the chiclets, solder, and clip the excess wire.

Do the same for the 330 Ohm resistor (Orange, Orange, Brown), between chiclets #5 and #6 (Figure 27.5).

STEP 4: CAPACITORS

The first capacitor is the 10 uf electrolytic. This is the only polarized capacitor in the circuit, so it is important to solder it with the proper orientation. Electrolytics typically have a stripe painted on the side of the package that indicates which lead is negative, and the leg on the negative side is usually shorter. Bend both of the leads so they are perpendicular to the package as shown in Figure 27.6.

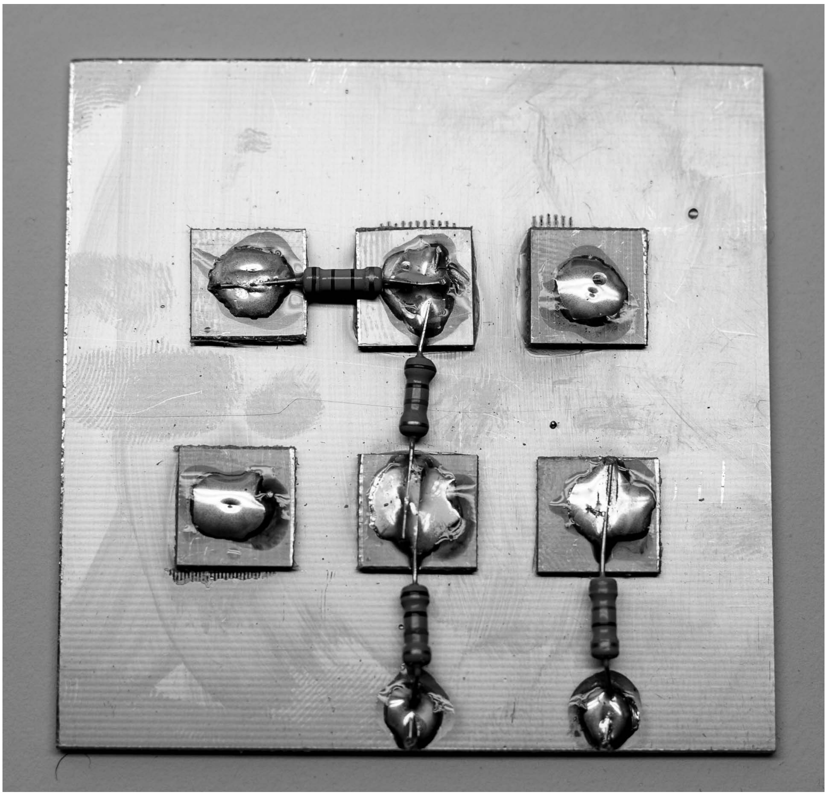


Figure 27.5 Four resistors installed.

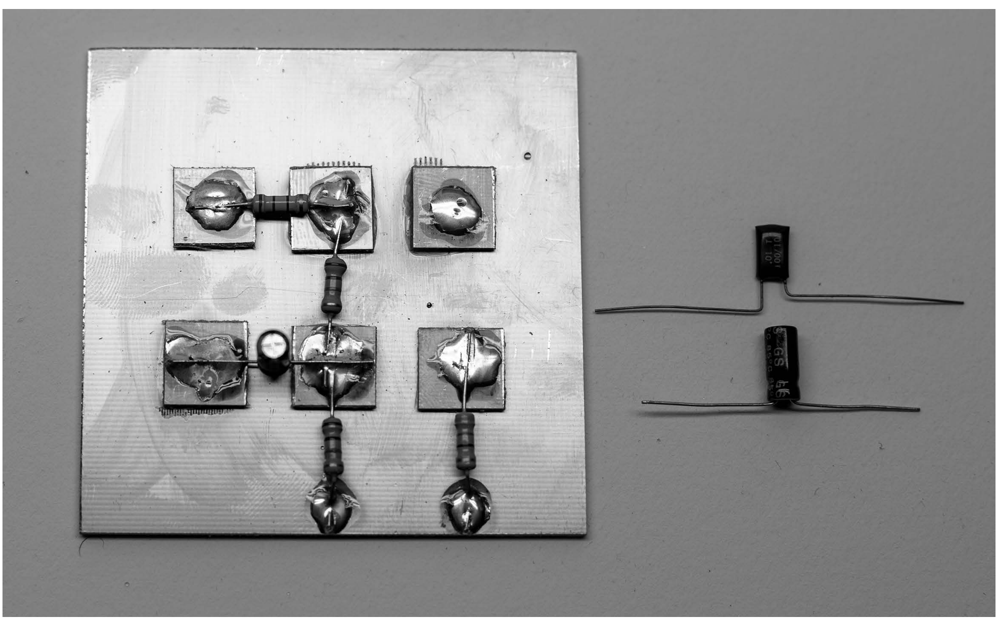


Figure 27.6 Capacitor with bent leads.

Solder the negative lead to chiclet #1 and the positive lead to chiclet #2 and trim the leads. Make sure the capacitor is standing straight up, perpendicular to the board. Ensure there's no slack in the leads, as that increases the possibility of shorting the leads to the ground plane.

The 0.01 uf (103) film capacitor goes between chiclet #2 and the ground plane. Bend the leads so that the body of the capacitor is perpendicular to the ground plane. Solder and trim. The 10 pf (10) mica capacitor connects chiclets #3 and #4. Bend leads so it stands vertically and then solder and trim. The last capacitor is another 0.01 uf (103) film capacitor. Solder it between chiclet #5 and the ground plane and trim (Figure 27.7).

The 4–34 pf trimmer capacitor has a little screw on the top you turn with a screwdriver (a non-conductive one is best) to adjust its capacitance. Turning the screw tunes the transmitter to different frequencies. It is a two-terminal device like the other capacitors, but one connection is doubled up for mechanical stability on the circuit board. This capacitor goes between chiclet #4 and the ground plane. Bend the middle terminal forward and the bottom of the side ones out. Solder the middle terminal to chiclet #4 and solder the side terminals to the ground plane. The trimmer capacitor's screw should be facing up (Figure 27.8).

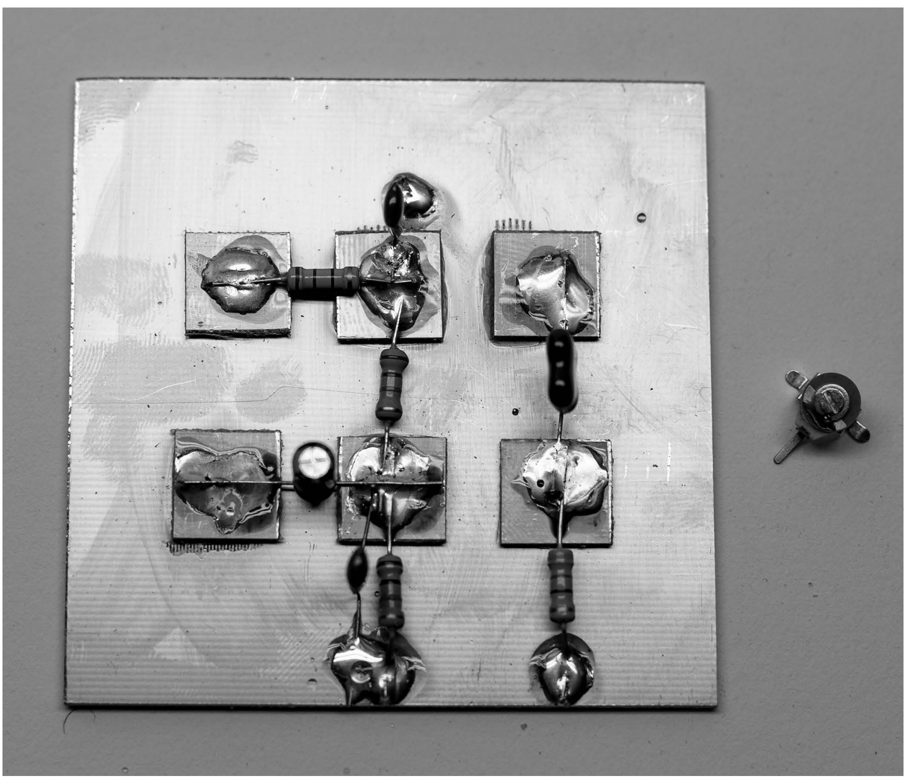


Figure 27.7 Fixed capacitors in place, trimmer cap with bent leads.

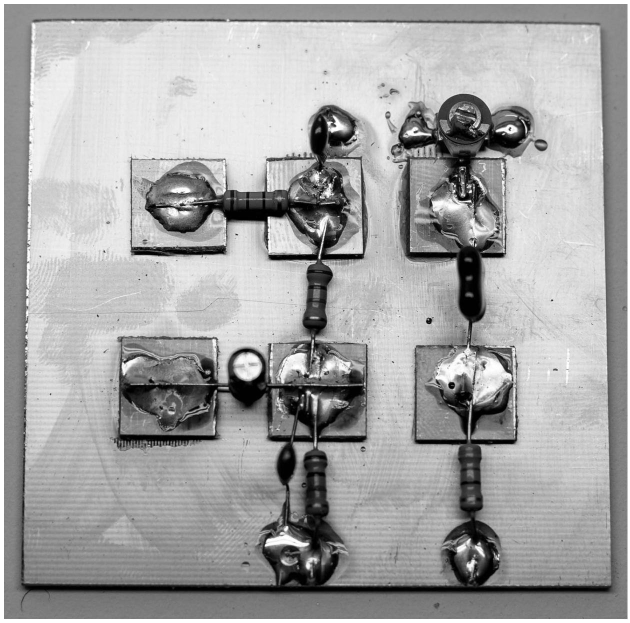


Figure 27.8
Trimmer cap soldered in to place.

STEP 5: INDUCTOR

The next component is one we will make ourselves! We will need a short length of 26 AWG insulated magnet wire 6–8 inches long. Wrap five, closely spaced turns around a 1/8-inch-diameter form—a mini-plug (like the one on earbuds) works perfectly (see Figure 27.9). Make sure you have *exactly* five turns. This coil, in conjunction with the

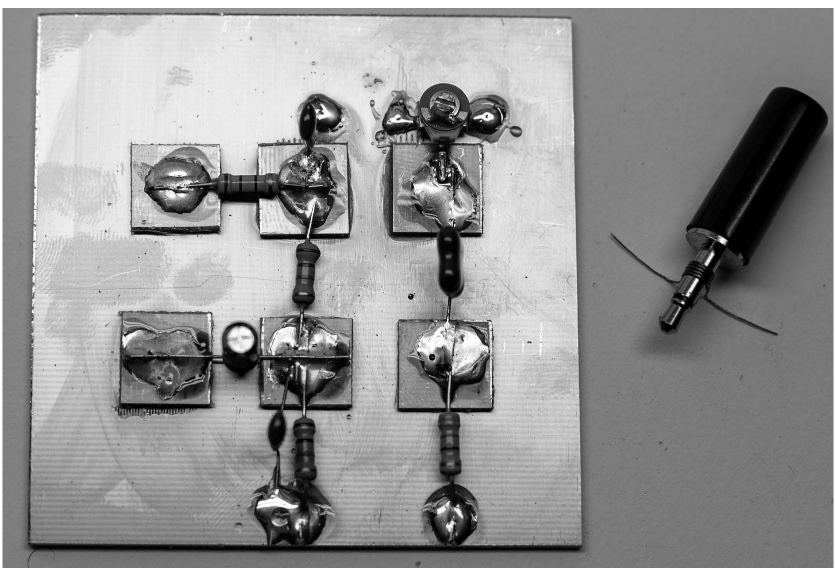


Figure 27.9 Coil wound on audio plug.

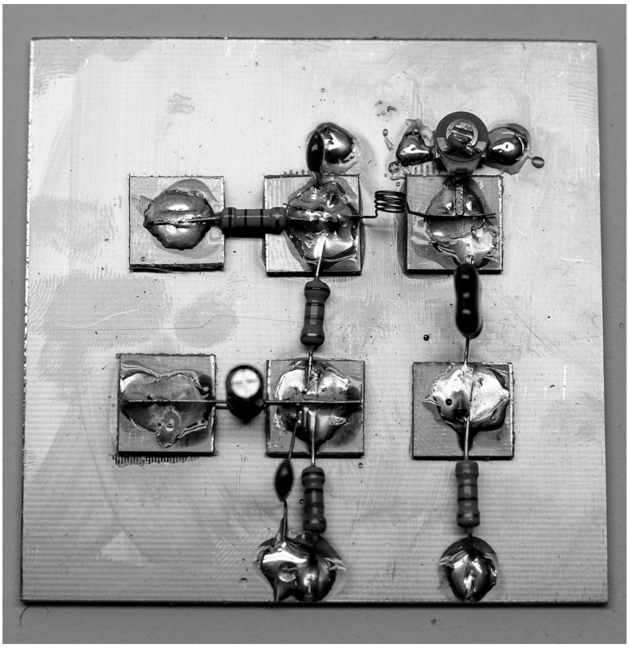


Figure 27.10
Coil soldered in place.

variable capacitor, sets the operating frequency: if you have fewer turns, the frequency will be higher; more turns will result in a lower frequency. These deviations can move the operating frequency out of the FM broadcast band, which will make it difficult (if not impossible) to receive your transmission on a regular radio.

Don't remove it from the plug just yet. Magnet wire has a thin enamel insulation that needs to be stripped from the ends so it can be soldered to our circuit. Clip the leads so that they're extend about 1/2 inch from each end of the coil. Pinch the wire between your thumb and the blade (not too hard to cut yourself) and scrape the leads close to the coil with a utility knife. Look for a change in color that shows that the enamel coating has been removed, revealing shiny copper. If you break the wire in the process, wind another.

Bend the coil ends outward and set them down so they connect chiclets #4 and #5. Solder and trim (Figure 27.10). If you're having trouble getting the solder to stick to the coil leads, you may need to scrape off more enamel (the solder will not stick to the insulation)—be patient, it's important to get this right.

STEP 6: SEMICONDUCTORS

It's useful to know whether our transmitter is on or off, so we'll add an ON AIR light. Take the LED and bend the leads as you did with the capacitors. This is a polarized component, and it must be placed in the circuit in the correct orientation. The shorter leg is the negative terminal, which will be soldered to the ground plane. The positive one goes to chiclet #6 (Figure 27.11).

The other semiconductor in the circuit is a transistor, which does the heavy lifting in our circuit. This is a three-terminal device, and each terminal has a name and a

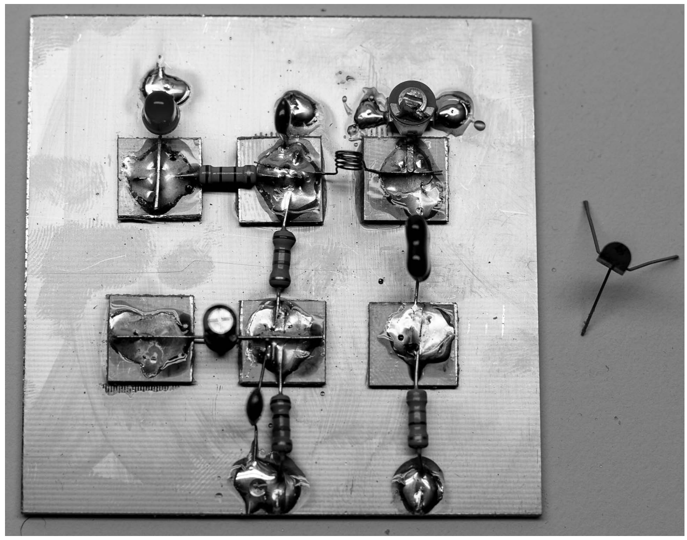


Figure 27.11
Transistor with bent leads.

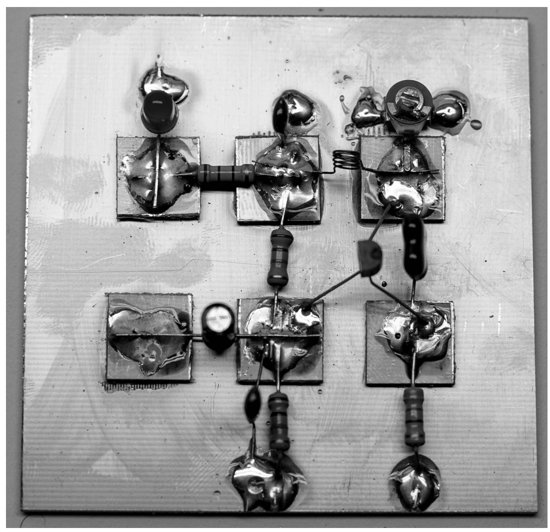


Figure 27.12
ransistor soldered in place.

function. On the BC337 transistor, if you are looking at the flat portion of the package that has writing on it, the terminals are, from left to right, Collector, Base, and Emitter. Bend the leads of the transistor so the Collector and Emitter point back at an angle and the Base points forward. If you set it down, the transistor should stand up like a tripod.

Place the transistor so that the Base touches chiclet #2, the Emitter chiclet #3, and the Collector chiclet #4. Adjust the leads by bending so the transistor now stands stably on the chiclets. Soldering this part is a delicate affair. It's easy to overheat the transistor, so solder each lead to its respective chiclet *quickly* and with the least amount of heat possible while still insuring a strong solder joint (Figure 27.12).

STEP 7: SIGNAL

Our transmitter is monophonic, but many sources of audio are stereo. If we use a mono audio plug to jack into our source, we'll only get one channel. This is usually good enough for our purposes. (You can use a stereo plug if you want, but you should sum the right and left channels through two 10 kOhm resistors.)

Take the 1/8-inch (3.5 mm) mono audio plug you used as a form to wind the coil and unscrew the black plastic housing.

Cut two 8-inch pieces of 20–22 AWG stranded (not solid) hookup wire. Strip about 1/4 inch of the insulation from each end. Solder one wire to the short tab. This will be the signal line. Mark the other end with a piece of tape. Put the stripped end of the other wire partway into the hole on the long tab and solder from the concave side. This will be the ground. Trim the excess wire at the jack and inspect for any shorts. You can fold the wings over with a pair of needle-nose pliers to grip the wires.

Feed the free ends of the hookup wire through the black plastic housing and screw the housing down onto the plug. Solder the signal line (marked with tape) to chiclet #1. Solder the other wire (ground) to the ground plane somewhere near chiclet #1. You can twist the wires to better reject noise and look smart (Figure 27.13).

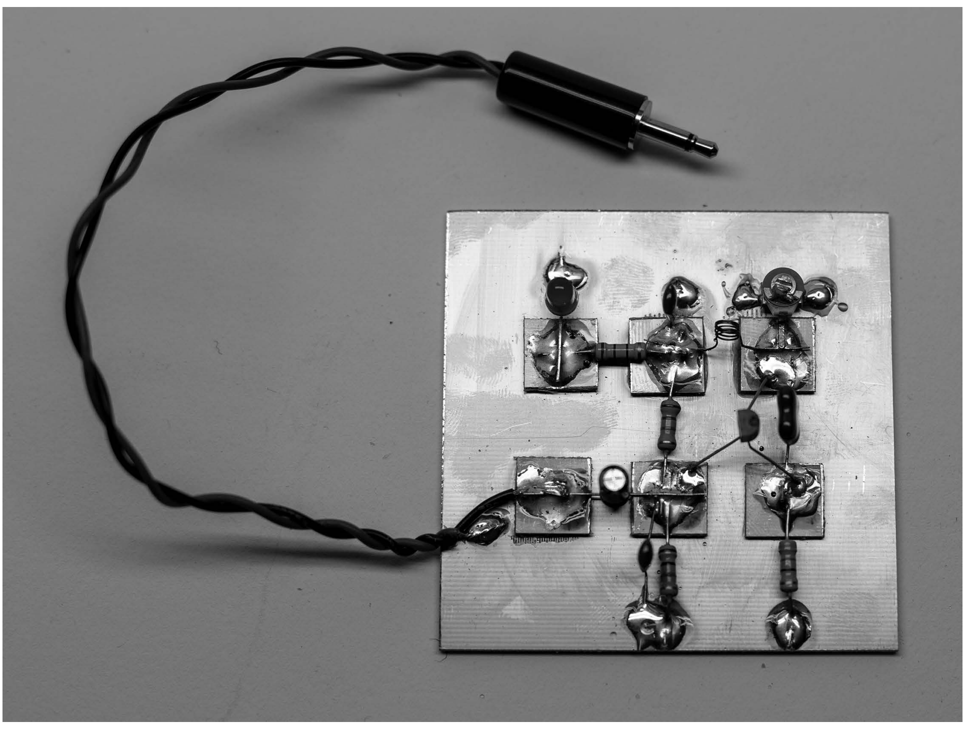


Figure 27.13 Output audio plug soldered in place.

STEP 8: ANTENNA

We need an antenna to get the oscillations of the circuit into the air. The simplest antenna is a whip antenna, which is basically a length of wire soldered to chiclet #3. The length of this wire should be some fraction of the wavelength of the frequency on which you want to transmit so the antenna is resonant at that frequency and radiates the signal more efficiently. The FM broadcast band spans 88–108 MHz, with 98 MHz in the middle. We use the terrifying equation:

(The Speed of Light)/(Frequency in Hertz) = (Length in Meters)

The speed of light is 3×10^8 meters per second, and the frequency is 98×10^6 Hertz, so the length of the antenna should be almost exactly 3 meters. This is the correct length of a full-wave antenna, which is inconveniently long. We can shorten the antenna by cutting it in half and then in half again to make a quarter-wave antenna, which gives us about 30 inches. Cut a piece of hookup wire 30 inches long and solder one end to chiclet #3 (Figure 27.14). Stiffer, solid-core hookup wire works better as it will stand up on its own. If you only have stranded wire, you can clip the free end to something. We want the antenna sticking up perpendicular to the ground plane.

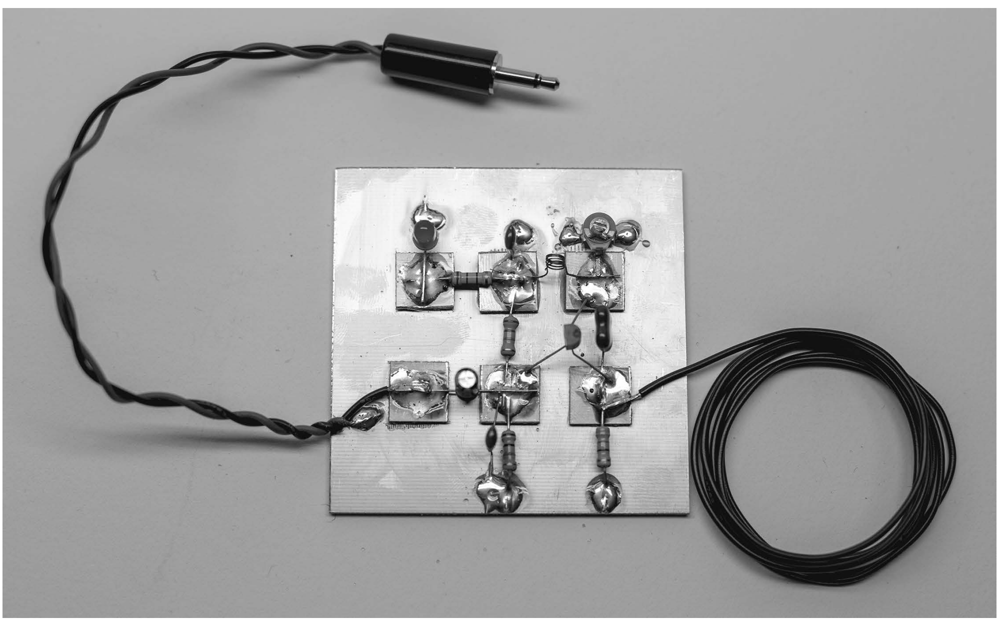


Figure 27.14 Antenna soldered in place.

STEP 9: POWER

We will power the transmitter with a 9-volt alkaline battery. I don't recommend an AC adapter because they are noisy, and you will hear this noise in your broadcast. Solder the black (−) wire of the battery snap to the ground plane close to chiclet #5. Solder the red (+) wire to chiclet #5 (Figure 27.15).

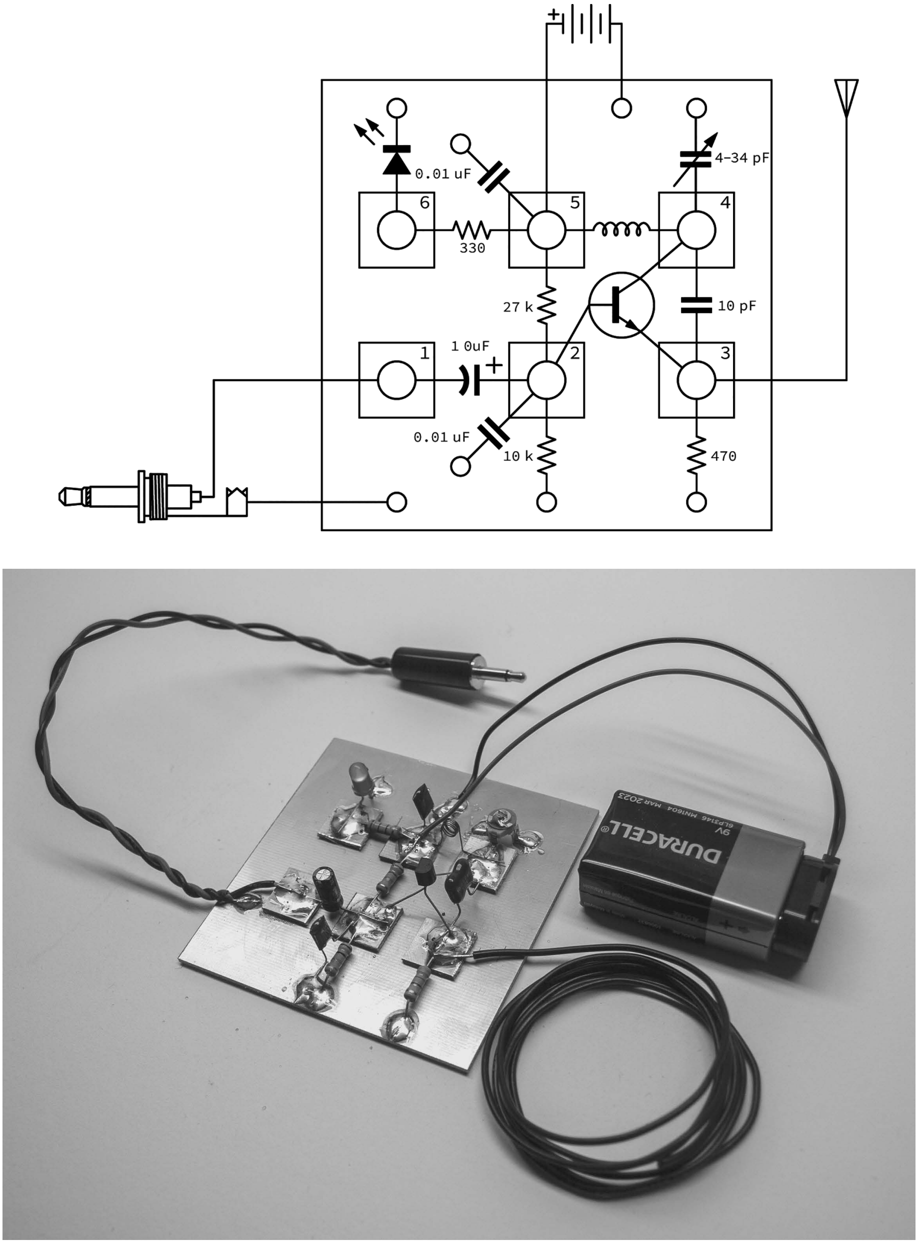


Figure 27.15 Battery clip soldered in place.

STEP 10: TEST AND TROUBLESHOOT

Find an FM radio and place it a few meters away from your transmitter. Connect a 9-volt alkaline battery to the battery clip. If the LED lights up, good. If not, immediately disconnect the battery. Is it warm? It shouldn't be. Check to see if the LED is in the correct orientation. If not, cut it out and solder in a new one. If the LED looks OK, check the rest of the circuit for shorts. Is there solder bridging a chiclet to the ground plane? Are there stray leads making connections where they shouldn't? Is the transistor soldered in the correct orientation? Are all your solder joints solid? Check everything twice and reapply power. Hopefully now the LED is on and the battery is cool as a cucumber.

Now comes the real test: will it transmit? Turn on your FM radio and tune it to a dead spot without a competing signal, somewhere near the middle of the band around 98 MHz. This is easier to do with an analog radio, with a sweepable dial, rather than a digital one, since digital tuners usually refuse to lock onto empty gaps in the transmission spectrum. Without connecting the transmitter to a source of audio, turn it on and use something non-metallic to turn the screw on the trimmer capacitor. As you turn (the screw can turn infinitely in either direction) listen for any small clicks or pops from the radio. It's really touchy, and tiny movements can make a big difference. When you hit the right frequency, the radio static should be replaced by a beautiful silence.

What's happening? The radio is receiving the carrier wave from the transmitter, but we haven't yet added any information, so we hear nothing. Now connect the transmitter to a source of audio such as your cell phone and play some music with the volume about halfway up. The frequency of the transmitter will probably change a bit after connecting the source, so you might have to adjust the trimmer capacitor again. When you get it tuned again, you should hear your music on the radio (Figure 27.16).

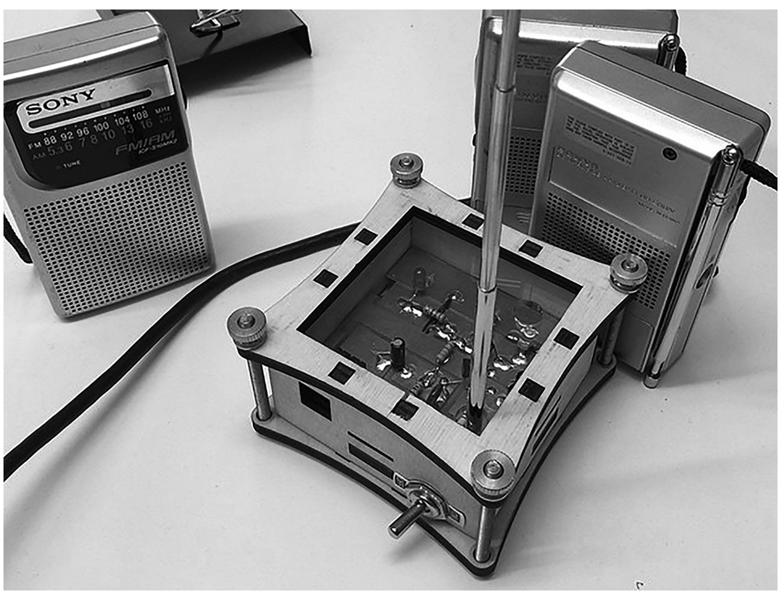


Figure 27.16 Dead-bug transmitter.

Congratulations, you're ON THE AIR! If not, the two most common problems are a burned-out transistor or a bad solder joint on the coil (usually the result of not scraping away enough of the insulation).

STEP 11: PLAY

As I said earlier, the transmitter design is very simple. When conditions are right, you can broadcast a decent audio program to an FM radio within about a 10-meter radius. Many factors affect this distance. One charming aspect of this design is its foregrounding of the physical nature of the radio signal. Reception will be different if the transmitter is outside rather than inside. Try multiple transmitters and/or receivers at the same time: place two transmitters tuned to the same frequency some distance apart and walk between them with a portable receiver—the radio will automatically grab the strongest signal. Audio programs can be spatialized using multiple sources and receivers. Put your hands on the transmitters or walk near them: the electromagnetic fields surrounding the transmitters interact with the human body. Gentle reader: experiment and give thanks to Tetsuo Kogawa for this design and the contributions he has made to the exploration of the aesthetic possibilities of electromagnetic space. Enjoy!

NOTE

1. anarchy.translocal.jp

CHAPTER 28

A Grab Bag of Samples

A Voltage-Controlled Radio Receiver

HOLGER HECKEROTH

You will need:

- A breadboard.
- SiLabs Si4825 FM-receiver chip.
- LM358 op amp.
- LD1117–3.3-volt voltage regulator.
- An Epson C-002RX crystal with a frequency of 32.768 kHz.
- A fixed inductance aka coil with 56 nH.
- Assorted resistors, capacitors, and potentiometers (see text).
- Some solid hookup wire.
- An SMT adapter PCB (see text).
- 2.54 mm/0.1-inch spacing pin headers.
- Assorted jacks and plugs.
- A power supply: a 9-volt battery, wall wart PSU with 9–12 volts and connector, or two AA or AAA batteries.
- An amplifier for monitoring sound.
- Hand tools.

Numerous composers and performers have been fascinated by radio transmissions. These signals either represent the local broadcast culture or, in the case of shortwave transmissions, a faraway place. They are topical and linear. You'll never know what you will get, but it will always reflect the *now*. Radio gives us a luxurious grab bag of samples with which to play.

In Chapter 27 we built our own radio transmitter. Now let's build a radio receiver from scratch and add voltage control so we can combine it with other synthesizer modules.

PREPARATIONS

Modern radio receiver chips, like the Si4825 we will use, integrate in a single chip nearly all the necessary parts for a high-quality multiband radio receiver. We will only need a minimum of common external components to make it work. No winding coils or searching for obscure parts, like in the old days. But there is one drawback: like most modern ICs, the Si4825 is only available as a confoundingly small SMT (surface mount technology) part. So we will start by soldering the tiny chip onto an adapter PCB so that you can plug it in your trusty breadboard for prototyping and experimenting. Our chip comes in a 16-pin SOIC package, so you'll need a 16-pin SOIC to DIP board—Figure 28.1). You can get them cheap from various online shops.[1] Many of them have two sides for two different SMT package formats.

SMT soldering requires practice and time. Practice with cheap or broken parts, relax, take your time, and don't drink too much coffee. Once the chip is soldered into place, solder pin headers to the PCB—this is the easy part, but make sure the long part of the header extends from the *underside* of the board, not the side to which you soldered the chip (Figure 28.2).

Whereas most of the chips in this book can be run on anything from 3 to 15 volts, the Si4825 demands a low supply voltage within a narrow range: between 2 and 3.6 volts. As we may want to interface our radio to circuits running on higher voltages later on (for example, our CMOS circuits or a modular synthesizer), it's convenient to lower our other, higher supply voltage to the right range for the Si4825. For that we will use the LD1117–3.3 voltage regulator (see "Power Supplies" in the Technical Bootcamp section of the website for information on how voltage regulators work).

COMPANION @ WEBSITE

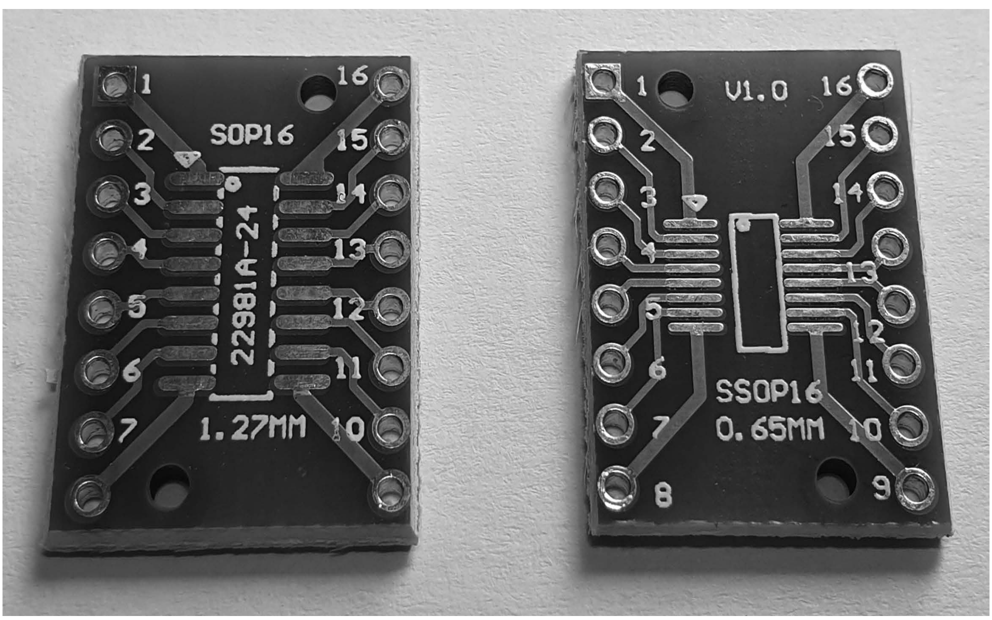


Figure 28.1 SMT adaptor.

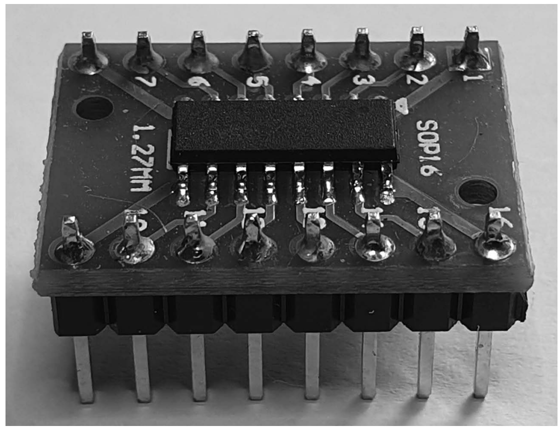


Figure 28.2
Si4825 soldered onto adaptor board.

If you are running all your circuits on 3 volts (i.e., two 1.5-volt AA or AAA batteries), you can leave out this part—just power everything off of the same batteries and you're good to go. But if you're using the regulator, start by connecting a 100 nf (0.1 uf) ceramic capacitor from the Vin of the LD1117 pin to your ground and a 10 uf from Vout to ground (these capacitors help stabilize the power supply). Connect the Vin pin to the positive voltage of your wall wart or 9-volt battery and the breadboard ground to the negative terminal of your wall wart/battery (Figure 28.3). If you have a multimeter, after powering up you should measure 3.3 volts between the Vout pin and ground.

In this application we can omit a heatsink for the voltage regulator as the circuit will only draw a modest current of 20–30 mA.

THE RECEIVER

There is a complete schematic of a Si4825-based receiver and lots of other useful information in the data sheet provided by the manufacturer, Silicon Labs—download it now.[2] We will build a simplified version of one design in the data sheet, one that only receives FM transmissions. This lets us throw out a big portion of the antenna and band selection circuitry, which makes your job much easier. You can always expand this design later to include shortwave and AM transmissions—just follow the data sheet and application notes. Figure 28.4 shows the schematic of the receiver we're going to build.

First connect the chip to your voltage regulator if you're using one (but make sure the power is not yet turned on!). Connect the 3.3-volt output to pin 14 on the header and ground to pin 15. Digital chips need decoupling capacitors, so put a 100 nf (0.1 uf) ceramic capacitor between pins 14 and 15 and put it as close to the chip as possible. The Si4825 won't start straight away if you turn the power on; it needs a *reset* signal for 100 us after the power has stabilized. (If you're curious, have a look at the Timing Characteristics in the data sheet.) A 100 kOhm resistor from pin 9 (RST—the designation in the data

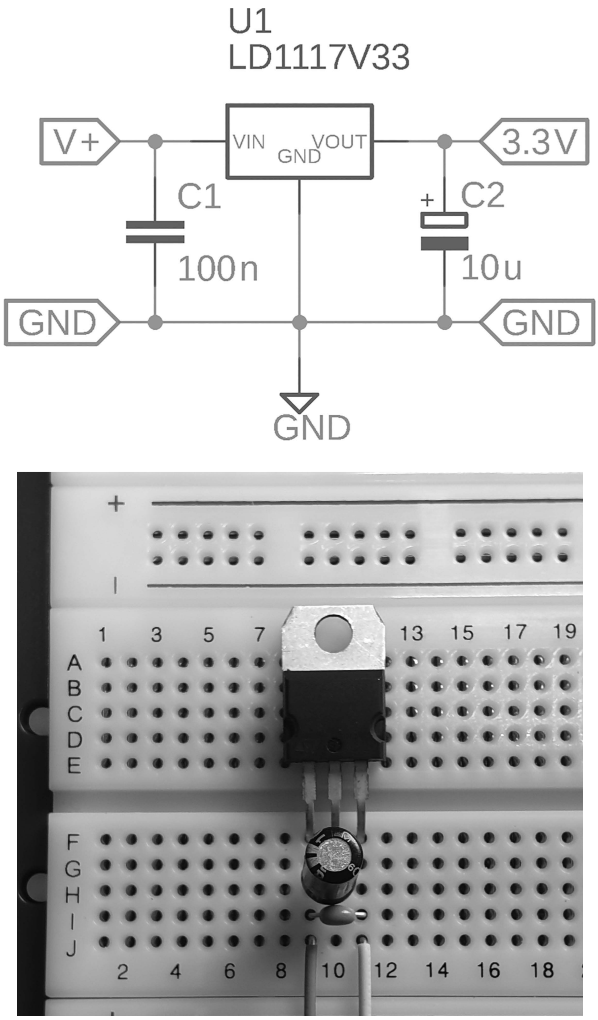


Figure 28.3 Voltage regulator schematic and breadboard.

sheet for "reset") to our positive supply voltage and a 100 nf (0.1 uf) capacitor from the same pin to ground generates a "slowly" ramping up voltage. Since the RST input being "active-low," this will keep the chip reset for a few microseconds after the power is turned on. We won't use the integrated volume control of the chip, so connect pins 10 and 11 to ground to disable it (if this function sounds tempting, you can always implement it later).

The Si4825 needs a stable time base to be able to tune in radio stations. For that you connect the 32.768 kHz crystal commonly found in quartz watches. Connect the crystal between pin 12 (XTALO) and pin 13 (XTALI). To force the crystal to oscillate, you need to add a 22 pf ceramic capacitor from each pin of the crystal (12 and 13) to ground. See Figure 28.5 to check your connections so far.

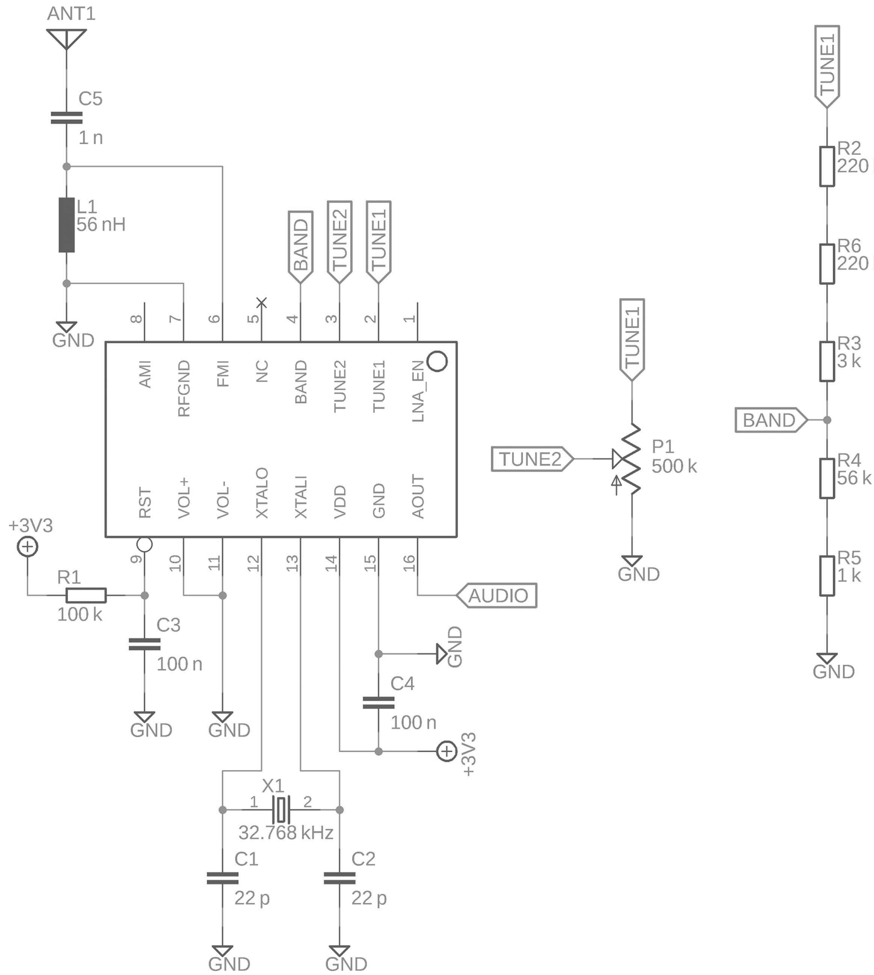


Figure 28.4 Voltage-controlled radio receiver schematic.

We need an antenna to pick up radio stations. Connect a 56 nH coil from pin 6 (FMI) to ground and an approximately 20-inch piece of wire with a 1 nf (0.001 uf) capacitor to the same pin. Now connect pin 7 (RFGND) to ground and the antenna is finished. (The application notes AN738 contain more information on building antennas.)[3] See Figure 28.6.

We have to tell the chip that we want to pick up FM transmissions, rather than AM or shortwave. We do this with a voltage divider, between pin 2 (TUNE1) and GND, whose output voltage is connected to pin 4 (BAND) to set the band we want to receive (a detailed description on how to set different bands can be found in application note AN738). The band we want to receive in Europe is FM2, which we select with a voltage divider consisting of a resistor 443 kOhm 1% from pin 2 (TUNE1) to a 57 kOhm 1% resistor connected to GND. The voltage from between these two resistors is connected to pin 4 (BAND). As 57 k is not a common resistor value, we

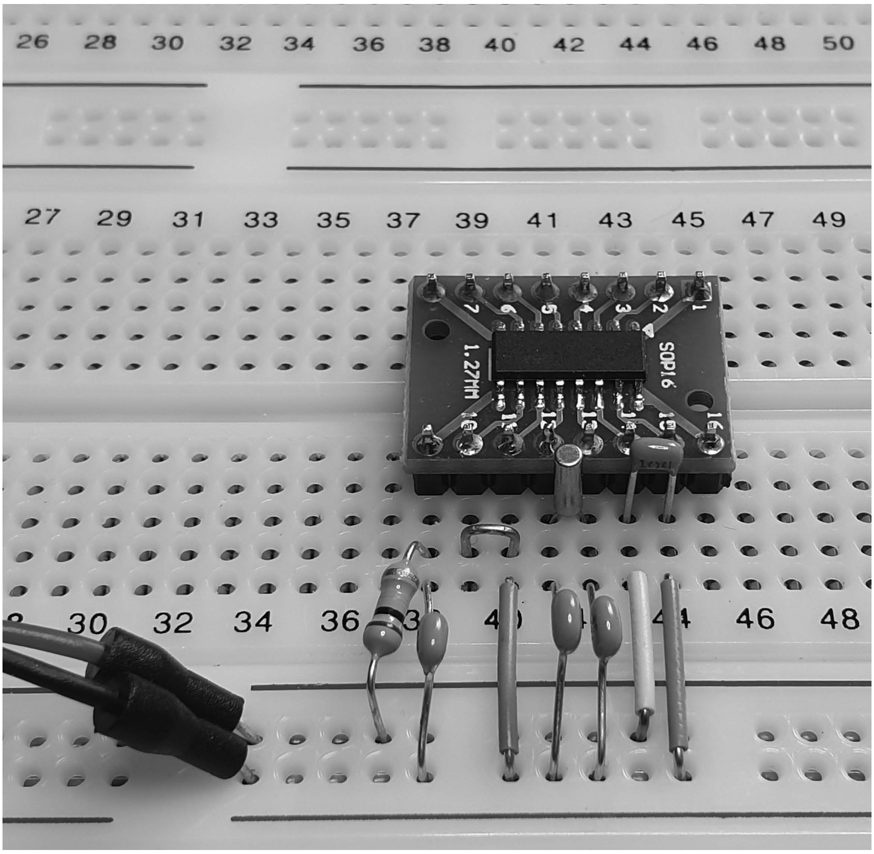


Figure 28.5 Step 1.

put 56 k and 1 k in series. For the 443 kOhm we use 220 kOhm + 220 kOhm + 3 kOhm. (This is one of the rare instances in this book where the resistor values must be precise—you can't get away with "something close" here.)

If you are in the Americas or South Korea, you would use band FM3 because of the different deemphasis. The resistor values for FM3 voltage divider are 433 kOhm 1% and 67 kOhm 1%. As these don't exist either, we again put resistors in series. For 67 kOhm we'll use 47 kOhm and 20 kOhm, and the 433 kOhm resistor is a 432 kOhm resistor and 1 kOhm resistor in series. See Figure 28.7.

For testing purposes connect a potentiometer in the range of 10 kOhm–500 kOhm to tune the radio (Figure 28.8). Connect pin 1 of the pot (one ear) to pin 2 of the chip (TUNE1), pin 3 (the other ear) to ground, and the wiper (the nose) to pin 3 (TUNE2).

Tip: if you cut away the mounting tabs of a 9 mm vertical potentiometer, it fits perfectly on a breadboard. See Figure 28.9.

Connect the tip of an audio jack to pin 16 (AOUT) and the sleeve to GND on the breadboard (Figure 28.10). Plug a cord from the circuit to your amp and power

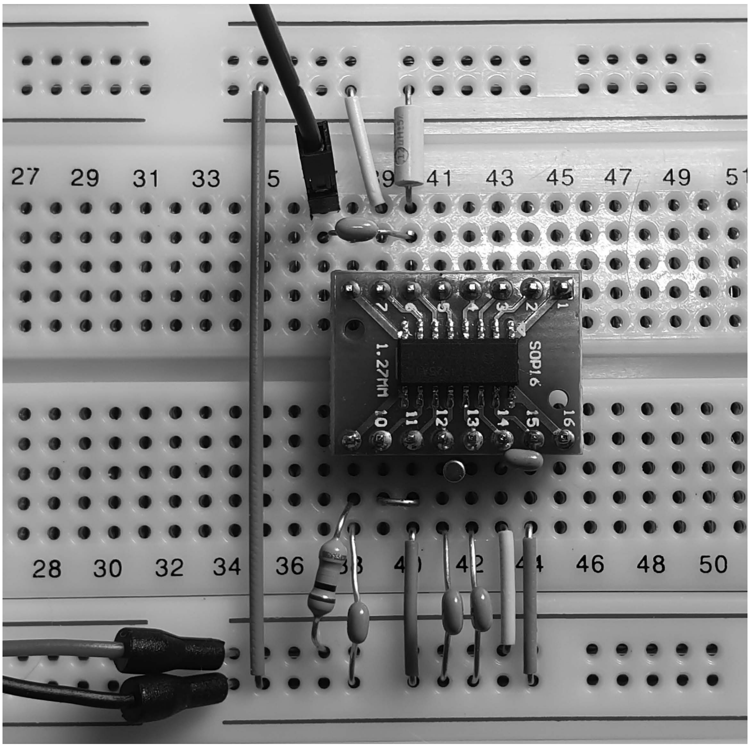


Figure 28.6   Step 2.

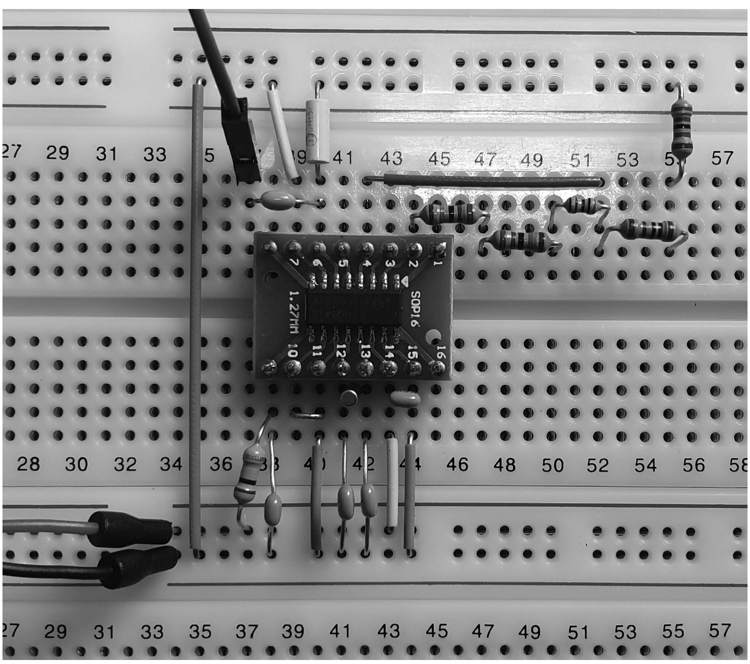


Figure 28.7   Step 3.

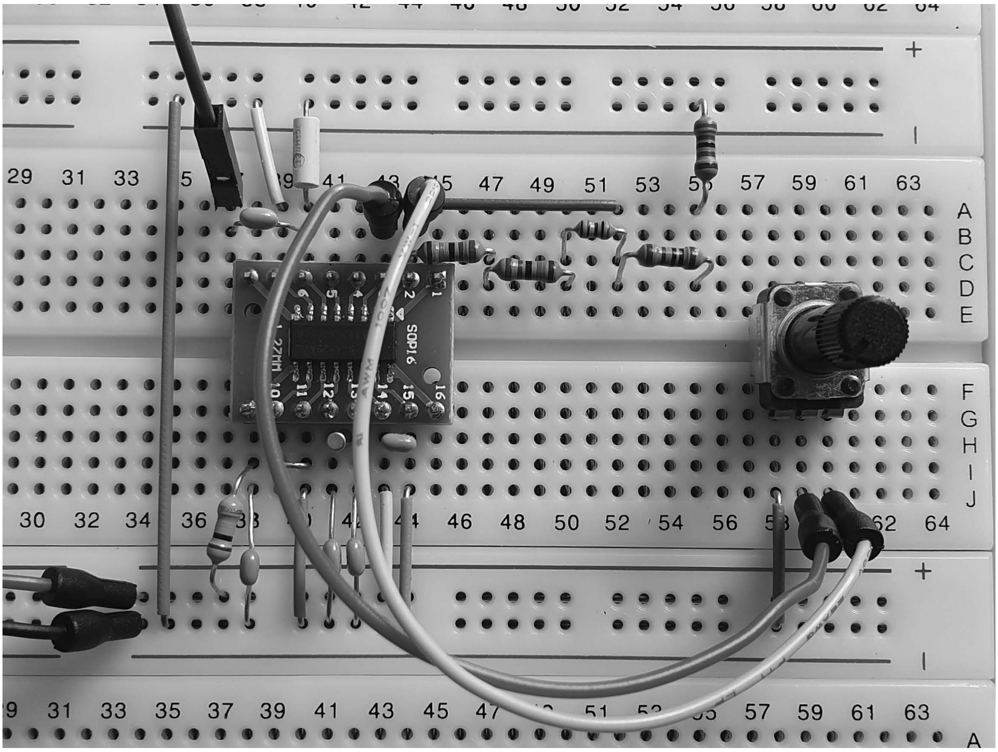


Figure 28.8 Step 4.

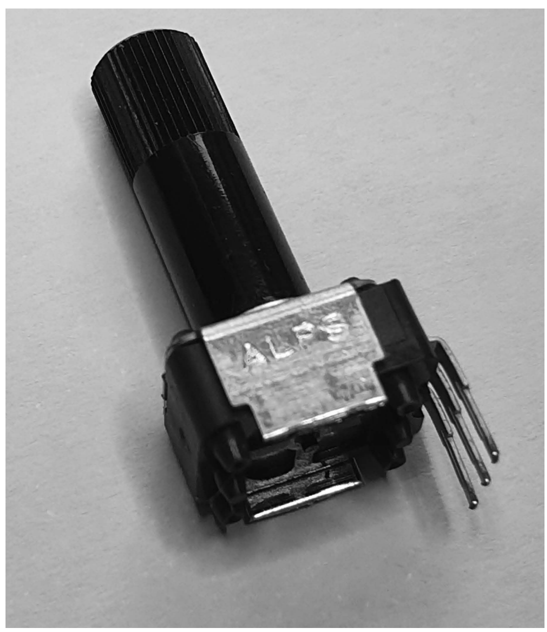


Figure 28.9 Trimmed potentiometer.

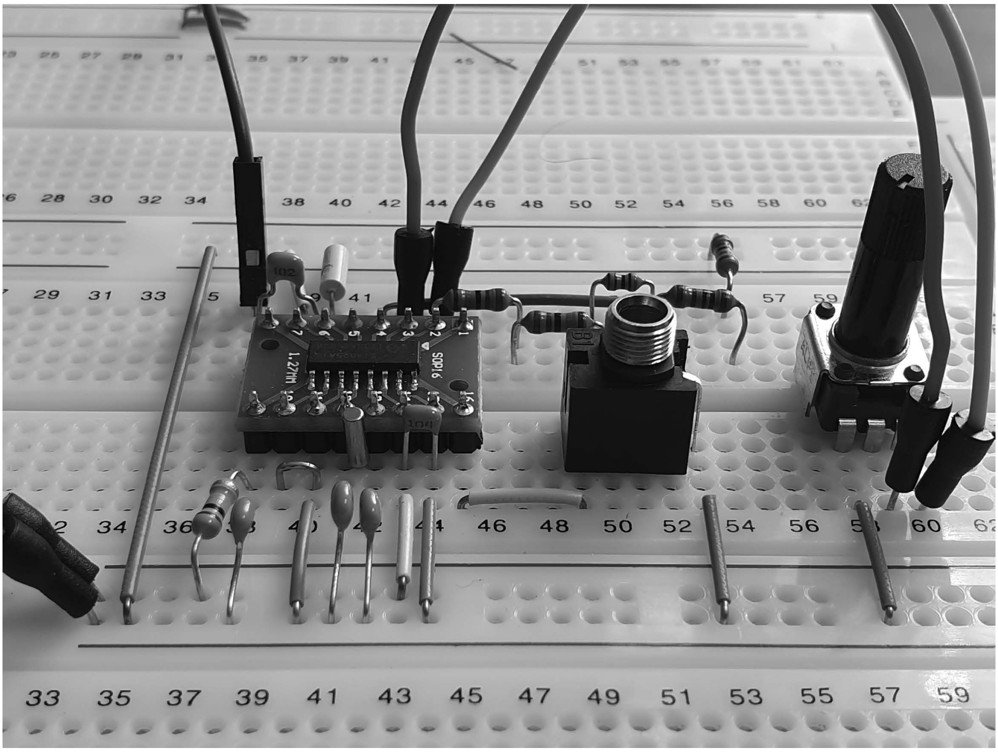


Figure 28.10 Step 5.

up the circuit. You should hear a radio signal. If not, disconnect the power supply and check the following:

1. Is the chip hot or is smoke escaping? Then there is really something wrong and the chip is probably broken.
2. Check the power supply and regulator. Is there a voltage of 3.3 volts between pins 14 and 15 when it is turned on?
3. Check and double-check all connections and part values.
4. Is your amplifier connected correctly and turned up?

MAKING IT VOLTAGE CONTROLLABLE

The next step is to make your radio voltage controllable. Actually, it already is voltage controlled, we're just not yet taking advantage of that feature. Unfortunately, we can't simply hook it up to *any* control voltage source as this might destroy the chip: the Si4825's supply voltage has to be between 2 and 3.6 volts, and no other voltage we connect to the chip should exceed the supply voltage (which also means it can't be negative). To be extra sure to not destroy anything, we will try to keep the control voltage in the range of 0 to 1 volt.

To achieve this, we will use an approach often seen interfacing ADCs in micro-controllers. The circuit consists of three parts. First, we *scale* the input voltage to the desired range, which for us is from 0 to 1 volt. Then we *clamp* the resulting voltage to the supply voltage of the chip to make sure no voltage outside that range reaches it. In a final step, we *buffer* the voltage since the ADC work best on low impedance sources. Figure 28.11 shows the schematic of circuit we are going to build.

The two resistors, R1 and R2, form the voltage divider to *scale* our control voltage source to the desired range. The formula for calculating a voltage divider is:

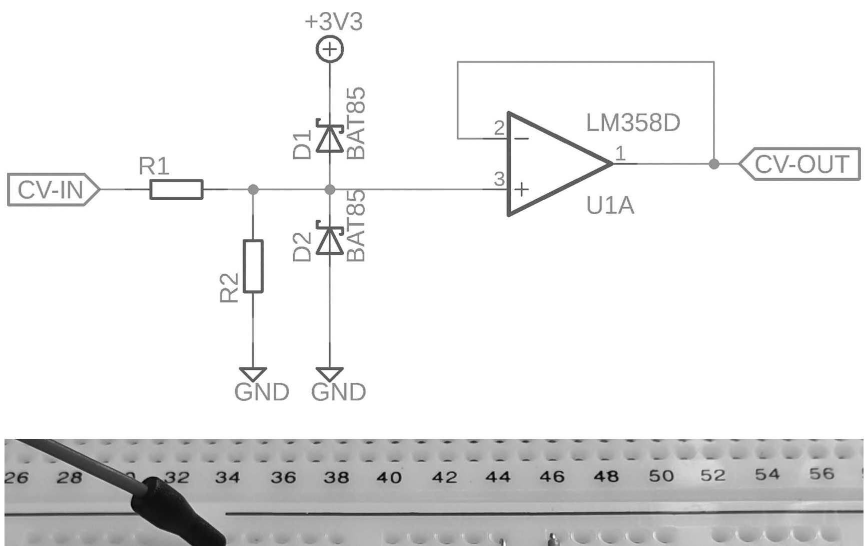


Figure 28.11 Control voltage input circuit schematic and breadboard.

Vout = Vin * R2 / (R1 + R2)

If our maximum control voltage is 10 volts, and we want the scaled voltage to be from 0–1, we could make R1 90 kOhm and R2 10 kOhm. That would give us:

10 V * 10 kOhm / (90 kOhm + 10 kOhm) = 1 V

The two diodes, D1 and D2, after the voltage divider *clamp* our scaled input voltage within the supply voltage range. The diodes work like a brick wall limiter (or fuzz-tone). If the voltage is a few tenths of a volt above 3.3 volts, D1 starts conducting, clipping the voltage. The same goes for D2 when the voltage is below 0.

The last part of the circuit is an op amp configured as a voltage follower, which *buffers* our scaled and clipped input voltage to feed the ADC of the Si4825. As we are working from a single supply (only a positive supply voltage) and within a small voltage range (0 to 3.3 volts), we are going to use the LM358, an op amp designed specifically for low-voltage, single-supply operation.

Now that you have a voltage control input, remove the tuning pot from the breadboard and connect the CV-OUT from our buffer (pin 1 of the op amp) to pin 3 (TUNE2) of our radio.

CONTROLLING IT

Simple Tunable Keyboard

You will need:

- A handful of pushbutton switches.
- Same number of potentiometers 10 kOhm–100 kOhm.
- Same number of diodes 1N4148 or similar.
- Some solid hookup wire.

As a first controller for our FM receiver, we'll build a simple, tunable keyboard. It does not work like a traditional keyboard, where each key has a fixed voltage. The voltage each key puts out is set by a potentiometer—like a cross between a keyboard and an analog sequencer. With it you can switch rapidly between pre-set stations, like on a car radio. The circuit is pretty straightforward (Figure 28.12).

One terminal on each pushbutton is connected to the +V supply of our receiver; the other is connected to an outside lug of a potentiometer (one ear). The other lug/ear of the pot is connected to ground, and the wiper (nose) is connected to the anode (positive side) of a small signal diode. The cathodes of the diodes are all wired together as the control voltage output. When you press a button, its potentiometer acts as a voltage divider, and the corresponding voltage will be present at the summing node of the diodes. The diodes are necessary to prevent interaction between the pots (this arrangement of voltage dividers and diodes is similar to the sequencer circuit in Chapter 20). If you press more than one button, the highest voltage of all

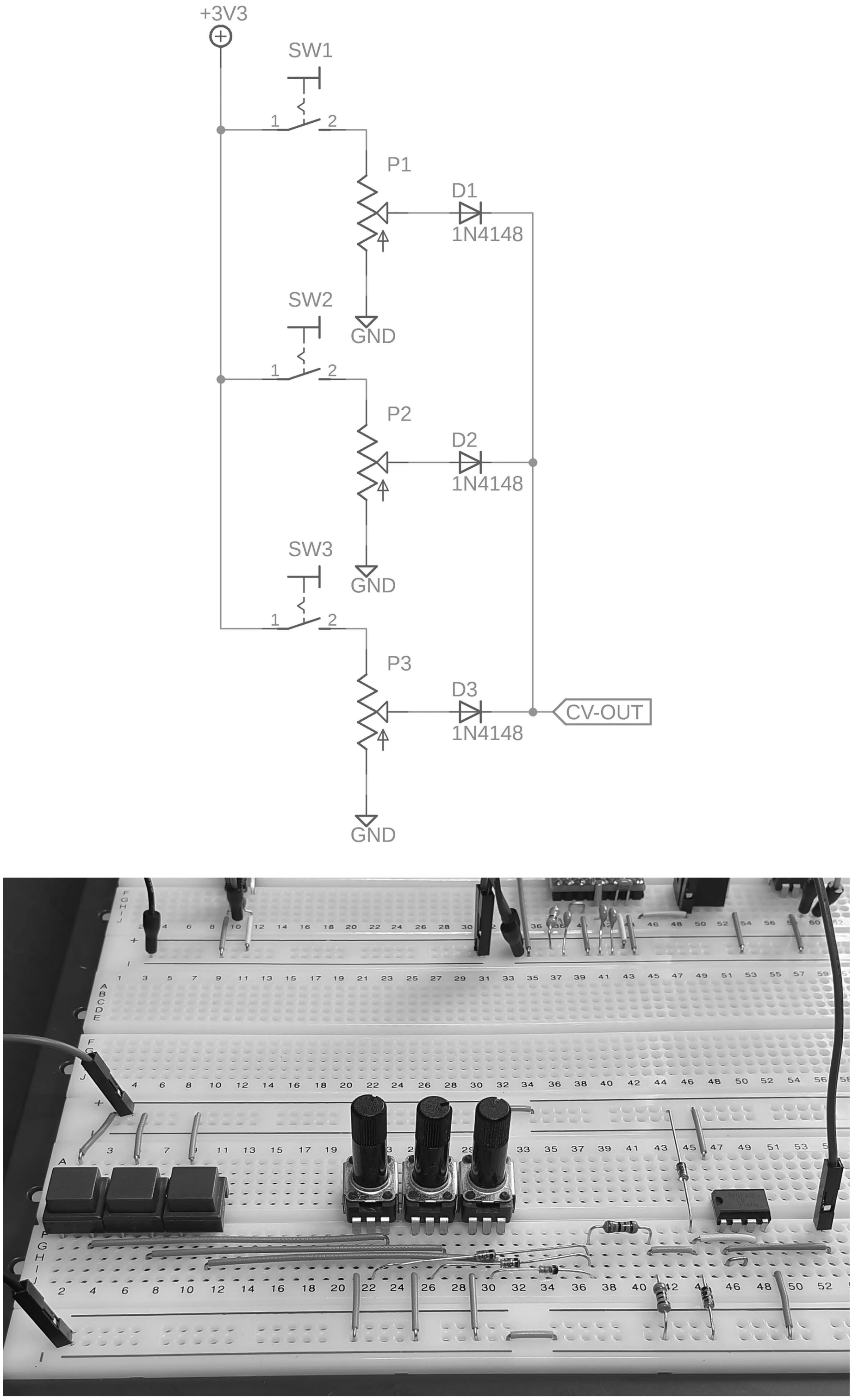


Figure 28.12 Keyboard circuit schematic and breadboard.

buttons pressed will be present at the output. If you don't press any button, the output voltage will be 0 volts, tuning the receiver to the lowest possible frequency in the selected band, and you will probably hear nothing more than static.

The last thing we have to do is to choose the right resistors for our control voltage input circuit. Since you are powering the keyboard off our 3.3-volt supply, you will have to get the output down to roughly one-third. Use 20 kOhm for R1 and 10 kOhm for R2 in our control voltage input circuit and you're good to go. You can expand this circuit with more switches, pots, and diodes.

Low Frequency Oscillator

With some automation things get more interesting. You will need one of the Schmitt Trigger oscillators chips from Chapter 13 (74C14, 40106, or 4584). For this circuit you will use a split voltage supply. Our oscillators will be powered by 9 volts and our receiver, via its voltage regulator, from only 3.3 volts. Because of this we have to adapt the two resistors of our control voltage input circuit to translate from the higher voltage. The output of the Schmitt Trigger swings fully between its supply rails, so we have to attenuate 9 volts down to 1 volt. We'll use 150 kOhm for R1 and a 20 kOhm potentiometer for R2—the trimmer potentiometer "tunes" the control voltage range to exactly 0–1 volt (Figure 28.13).

Select the capacitors of the oscillators to be in the sub-audio frequency range of a typical LFO.

The LFO switches rhythmically between the lowest frequency of the receiver and the frequency you dial in with the potentiometer. This makes it jump back and forth between static (0 volts) and the frequency selected with the potentiometer.

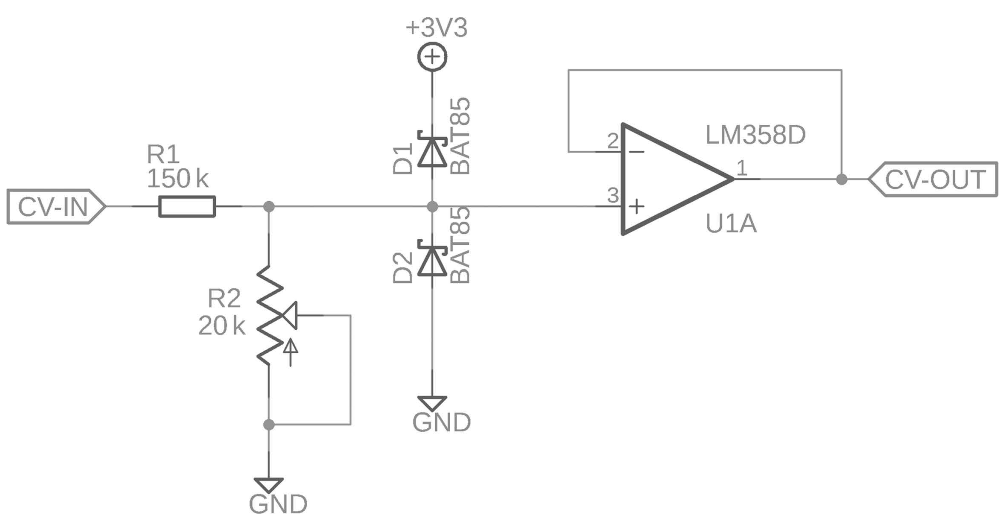


Figure 28.13 Control voltage attenuator circuit.

Experiment with different oscillator designs from this book (the 4093 designs in Chapter 15 can generate unusual patterns).

Sequencer

You will need:

- The 10-step sequencer from Chapter 20.

Interfacing the sequencer works similarly to the Schmitt trigger oscillator array described prior. Again, we have a split supply, and the output of the sequencer will be in the whole range of its supply voltage. With R1 = 150 kOhm and R2 = 18.7 kOhm, we get it down to our desired range of 0–1 volt.

Now you have a sequencer playing the receiver by switching between stations.

Modular Synthesizers

WARNING: if you're going to connect anything to your beloved (and expensive!) modular synth, you should know what you're doing. You can potentially damage the modules.

How can you control your VC radio from a modular system? We have already discussed all the building blocks you will need, so you might guess what's coming up next. First, you should power the receiver via the 3.3-volt voltage regulator connected to the power supply of your modular. This way both circuits share a common ground so we don't have to worry about that. Then you have to adapt the voltage coming out of the synth to our desired range of 0–1 volt.

For Eurorack systems, the voltage outputs of modules are typically:

- Audio signals: −5 to +5 volts.
- Pitch CV: 0 to +10 volts.
- LFO: −2.5 to +2.5 volts.
- ADSR: 0 to +8 volts.
- Trigger, Gate, Clock: 0/+5 volts.

But don't bet on that. A lot of companies try to adhere to this standard, but some don't. Check the documentation on each module. Still, we can be sure of one thing: the output voltage will always be in the range of the power supply in your rack, which is typically −12 to +12 volts. Our input circuit protects the Si4825 from any voltage above 3.3 volts or below 0 volts. So we only have to worry about attenuating it to optimize the control voltage range. Start off with R1 110 kOhm and R2 10 kOhm, thus attenuating the maximum 12 volts down to 1 volt. Then you can gradually increase R2 until you like how it responds.

Figure 28.14 Early incarnation of the VC FM receiver controlled by Arduino for mapping stations to keyboard (made for Antonio de Luca).

Figure 28.15 VC FM receiver controlled by Teensy 3.6 with graphical user interface and keyboard mapping.

Integrating a voltage-controlled radio in your modular synth opens up whole new sonic worlds (Figure 28.14 and 28.15). Experiment and enjoy!

NOTES

1. www.amazon.de/gp/product/B00O9W6RLQ; www.amazon.com/uxcell-SSOP16-TSSOP16-Adapter-0-65mm/dp/B00O9W6RLQ; www.amazon.co.uk/SSOP16-0-65mm-1-27mm-Double-Adapter-Green/dp/B01EZQAM6G
2. www.silabs.com/documents/public/data-sheets/Si4825-A10.pdf
3. www.silabs.com/documents/public/application-notes/AN738.pdf

CHAPTER 29

A Lo-Fi Sampler and Looper

HOLGER HECKEROTH

You will need:

- A breadboard.
- ISD1820 voice recorder chip.
- Assorted resistors, capacitors, and potentiometers.
- Some solid hookup wire
- Assorted jacks and plugs.
- A 9-volt power supply (wall wart or battery).
- An amplifier for monitoring your sounds.

INTRODUCTION

This is a quick and dirty sampler and looper circuit. It doesn't sound like a computer program, a full-blown hardware sampler from the 1990s, or even a contemporary looping pedal. It's lo-fi, gritty, pauses at the beginning of the loop, and clicks on playback, but it has lots of character, it is cheap and fun—and you will build it yourself!

THE ISD1820

The ISD1820 integrated circuit is designed to record and play back voice messages in greeting cards, warning signals, vending machines, etc. Its circuitry is optimized for recording and playing back the constricted frequency range of the human voice, so don't expect hi-fi. It sounds like a cheap radio. The most challenging part of this project is getting hold of the IC. If you search the internet, you can find a few shops specializing in DIY music electronics, as well as some auction platforms, that sell these chips.[1] But it's easier to find a cheap, ready-made circuit board with the ISD1820 on it. So if you can't find the IC itself, search for one of these boards online and pull the chip out (just make sure it's socketed so you don't have to de-solder it—Figure 29.1).[2]

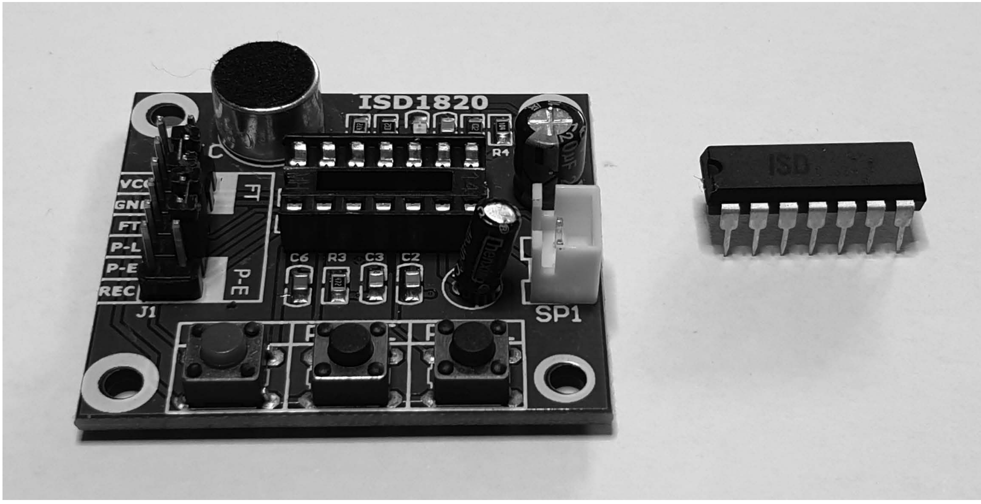


Figure 29.1 ISD1820 pulled from a circuit board.

THE LOOPER

Figure 29.2 shows the schematic of the circuit we are going to build. Note that it has very few parts, so it should not intimidate you. This chip runs off a power supply somewhere between 2 and 5 volts. We're going to use a 3.3-volt voltage regulator like the LD1117, wired to a 9-volt supply. This could be either a wall wart PSU or a 9-volt battery. This only adds three more components and makes expanding the sampler much easier (Figure 28.3 in the previous chapter). Alternatively, you can power the circuit from any 5-volt USB supply left over from your last cell phone or two or three 1.5-volt batteries (i.e., AA or AAA).

Connect the 3.3 volts from the regulator to pins 11 and connect pins 8 and 14 to ground (Figure 29.3).

You must control when the chip will record and when it will play back. This is done through pins 1, 2, and 3.

- To record, connect a positive voltage (3.3 volts) to pin 1. As long as this voltage is present, the chip will record.
- Pin 3 (PLAYL or "Play Level") plays back the last recording for as long as the voltage is held high. The playback will stop automatically once the end of the recording is reached.
- Pin 2 (PLAYE or "Play Edge") is what is known as a "one-shot trigger." After a transition from 0 volts to a positive voltage (aka positive edge, hence the pin name), the playback will start and continue until the end of the recording, even if you disconnect the voltage.

We connect the REC pin to our supply voltage via a pushbutton and add a 100 nf (0.1 uf) ceramic capacitor to ground. The capacitor is necessary to debounce the button

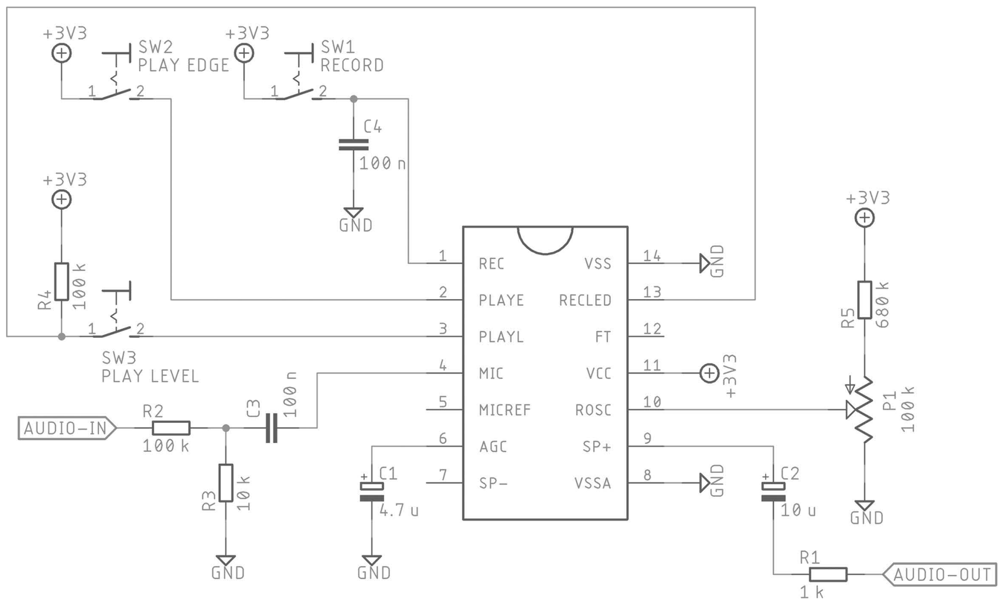


Figure 29.2 Loop project schematic.

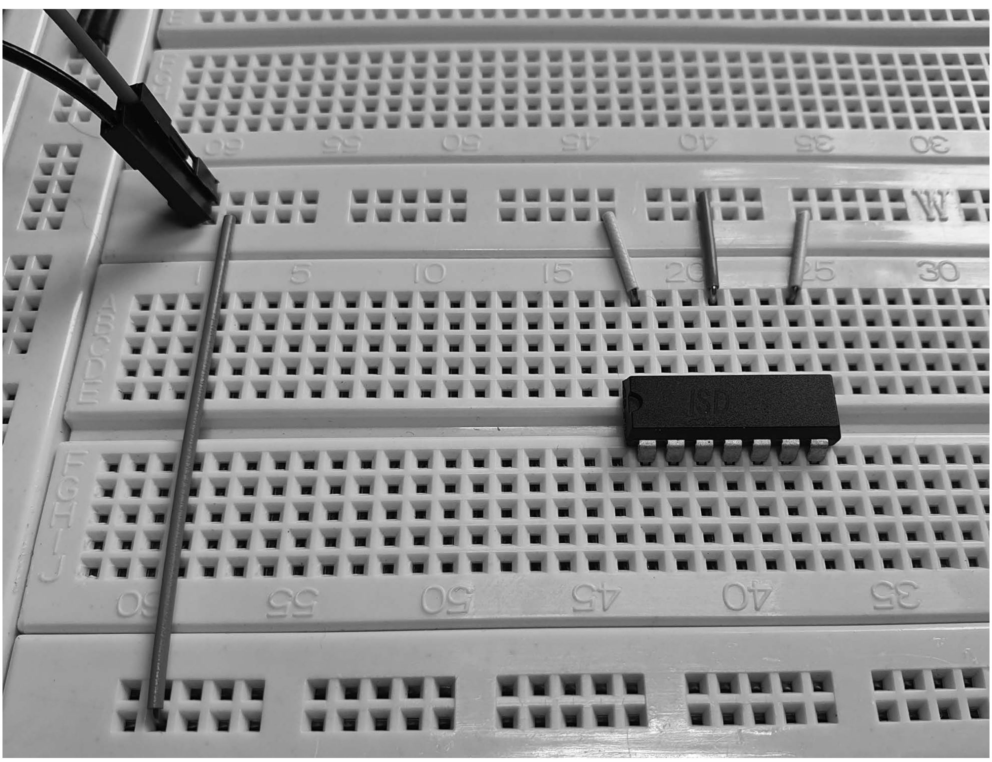


Figure 29.3 Step 1.

(otherwise it might retrigger the recording for multiple very short times and not record properly). The PLAYE input is even simpler. Just connect a pushbutton to the positive supply voltage. This play mode, always having the same duration, is not very interesting. You can leave it out if you want.

PLAYL will play the sample as long as it is high, more like a proper keyboard sampler, but won't loop once its end is reached. You can use the pin 13 output (RECLED) to automatically retrigger and loop the sample. RECLED is normally used to turn on a LED when the chip is recording, but it also blinks the LED when the end of the recording is reached during playback. We will use this blink signal for the retriggering. The pin doesn't output a positive voltage, but the cathode of an LED whose anode is connected to +V through a resistor is pulled to ground while the chip is recording or when the end of the sample playback is reached, which lights the LED. If we connect the other terminal of the switch connected to pin 3 (PLAYL) to the positive supply voltage with a 100 kOhm resistor, and also to pin 13 (RECLED), the pushbutton will pull the pin high and the RECLED function will pull it low for a short time at the end of the recording. After that momentary low, PLAYL bounces back up to high, and retriggers the playback (Figure 29.4).

(For more information on this kind of kluged triggering, see Chapter 21 on advanced circuit bending.) If you get tired of holding down the button, you can replace it with a toggle switch (you can even wire a momentary switch and a toggle switch in parallel—Figure 29.5).

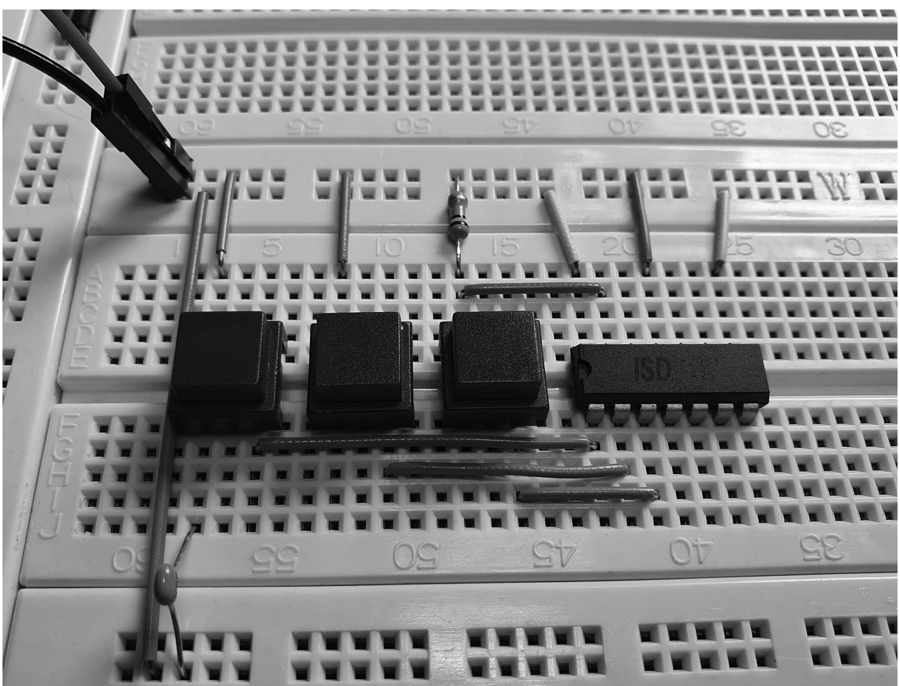


Figure 29.4 Step 2.

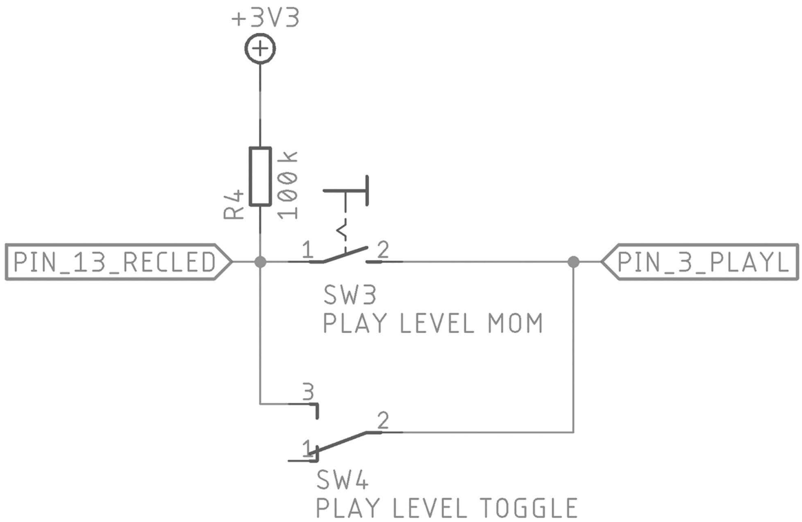


Figure 29.5 Momentary and toggle switches in parallel.

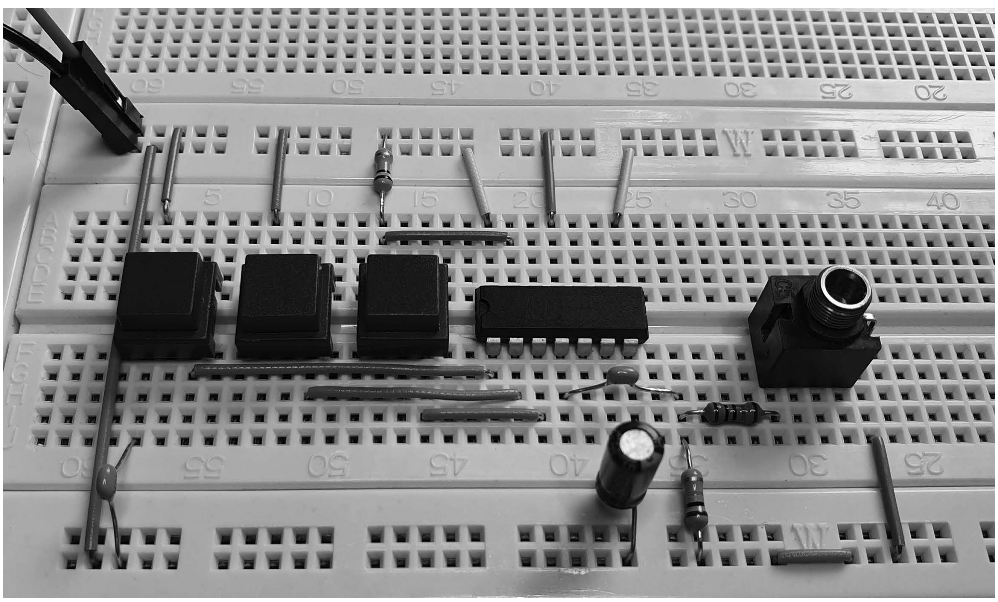


Figure 29.6 Step 3.

Now that you can trigger the Record and Playback, the next step is getting an audio signal in and out of the chip. This IC is designed to be directly fed from a microphone, which puts out a low-level signal. If you want to record a line-level signal (from a mixer, computer, etc.), a CMOS noisemaker, or a synthesizer, you will need to attenuate the signal through a voltage divider and couple it with a 100 nf (0.1 uf) capacitor. Values of 100 kOhm and 10 kOhm for the divider will lower the signal to one-tenth, which should get you in the right range (Figure 29.6).

Don't worry about the signal being too *low*. The ISD1820 has a built-in automatic gain control (AGC), which will boost signals that are too weak. To configure the AGC, connect a capacitor from pin 6 (AGC) to ground. An electrolytic cap of c. 4.7 uf gives good results.

The audio signal is output from pin 9 (SP+). The 10 uf capacitor filters out unwanted DC voltage, and the 1 kOhm resistor limits the current flow. Connect the tip of an audio jack of your choice to the free end of 1 kOhm resistor, the sleeve to ground, and plug into your amp or mixer (Figure 29.7)

Pin 10 (ROSC) is maybe the most interesting pin on this chip. It controls the sampling and playback speed of the recorder chip. This can be set either *fixed* by connecting a known resistor to ground or made *variable* with a 100 kOhm potentiometer wired as a voltage divider (with an additional resistor to the positive supply—Figure 29.8).

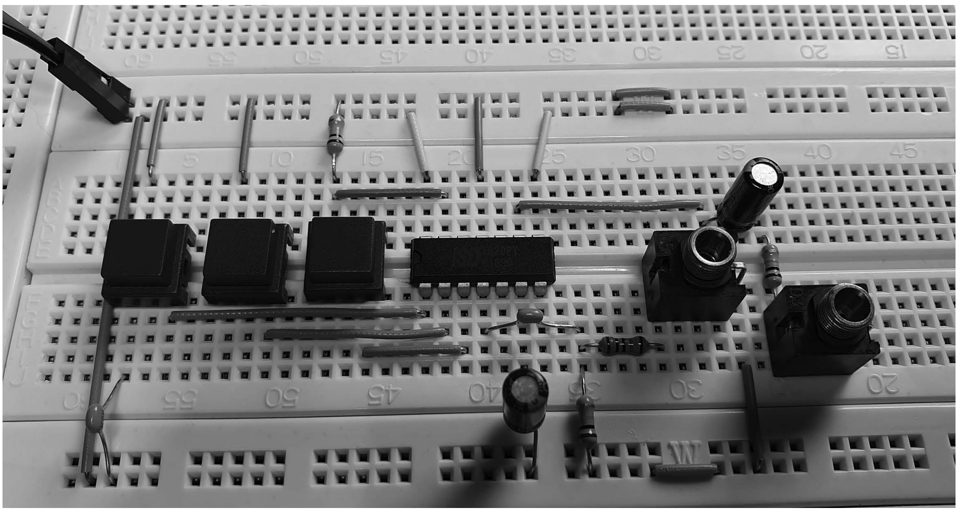


Figure 29.7 Step 4.

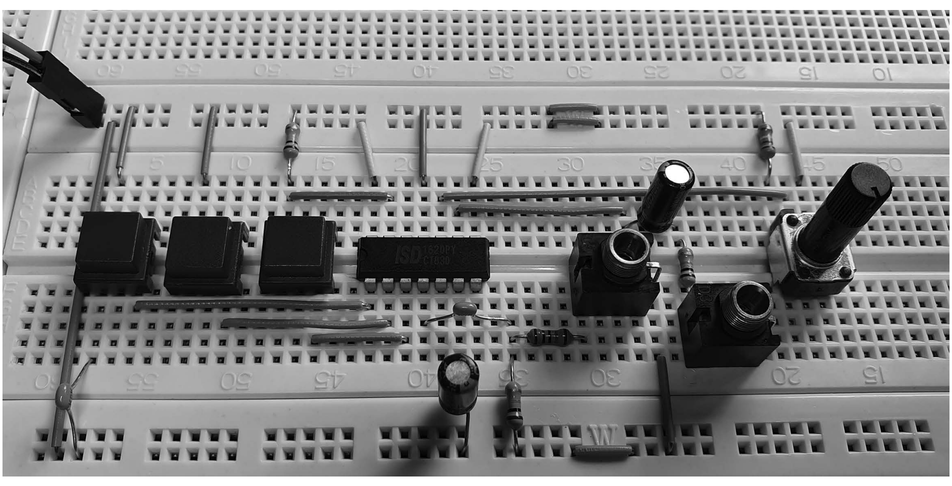


Figure 29.8 Step 5.

There are some quirks to this speed control. It not only controls the playback speed but also the recording speed and sample rate. The setting has a direct effect on the sampling quality and maximum recording length. The maximum recording length will be somewhere between 5 and 20 seconds, depending on the position of the pot. That's more than enough for some glitchy loops (if you don't mind reduced bandwidth).

Sampling at a *low* speed will result in lower quality, but you can speed up the sample more on playback. A *high* sampling speed will give us better audio quality and the option of lowering the playback speed. It's a matter of taste. If you set the sampling speed too high, you will only hear a buzz on playback. Try different settings to get a feel for the response.

You can change sampling speed while recording, and these speed changes will be present on playback—like a reel-to-reel tape recorder when you change speed while recording. Experiment with varying the recording and playback speed. It will result in unusual looping patterns, drones, screeches, and drunken cartoon character voices. Replace the potentiometer with a photoresistor for a Theremin-sampler hybrid.

EXTERNAL CONTROL

Things get even more interesting if we control the ISD1820 from external circuits instead of pressing buttons. Controlling the playback from an external source is easy. You only have to connect to pin 3 (PLAYL) a positive control voltage within the range of the supply voltage of our ISD1820. You can use a trusty Schmitt Trigger oscillator (see Chapter 13) to trigger the playback with a regular beat. Tune the oscillator into the sub-audio range, using an electrolytic cap 4.7 uf or larger.

As we will run it from our primary 9-volt supply, its output voltage will be too high for the ISD1820. We will use our Swiss Army knife circuit, the voltage divider, to bring the output into the range of our 3.3-volt supply voltage. A divider with 20 kOhm at the input and 10 kOhm to ground should work for 9 volts (Figure 29.9).

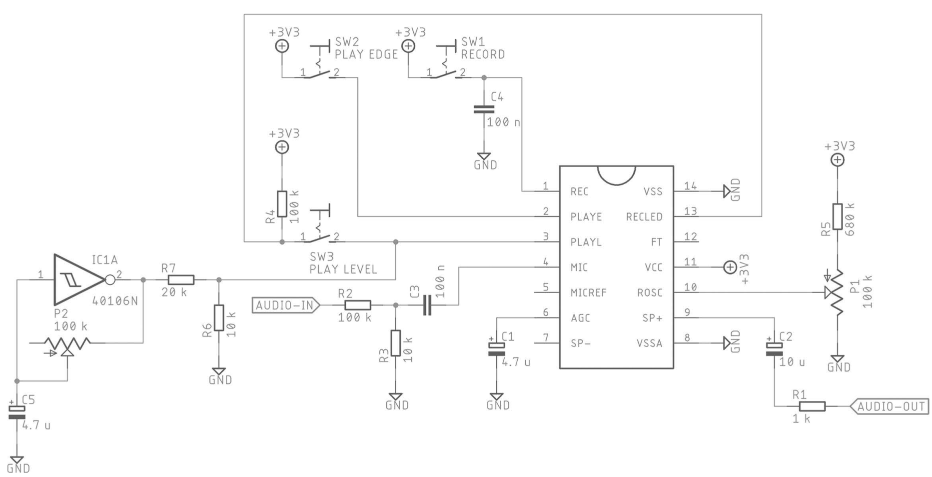


Figure 29.9 External triggering circuit.

Every time the oscillator's output is high, the recording will play back the recording from the beginning. With the pot of the oscillator, you can control the period of the trigger.

Try different settings for the trigger and the playback speed. Pressing the playback button will override the trigger signal from the oscillator. Press the buttons and turn the knobs and you'll get some groovy loops and beats. When the playback is running, you can still hit the record button to make a new recording, as this has priority over playback. You can sample a new loop on the fly that will be triggered after you let go of the record button. You can experiment with all oscillators from this book that put out a square wave. Try triggering with multiple oscillators mixed with diodes. Try cascaded NAND oscillators and modulate them (see Chapter 15).

NOTES

1. www.musikding.de/ISD1820
2. www.amazon.de/gp/product/B07L9B9M3J; www.elv.de/joy-it-sound-rekorder-und-abspielgeraet-isd1820.html

CHAPTER 30

The Bissell Function Block

A Lag Processor

PETER SPEER

You will need:

- One LM324N quad op amp IC.
- Two 100 kOhm linear potentiometers.
- Seven 1 kOhm resistors.
- Three 10 kOhm resistors.
- One 100 kOhm resistor.
- One 1 mOhm resistor.
- Two LEDs.
- Four jacks (if needed).
- Five 1n4148 diodes.
- Two 2n3904 NPN transistors.
- One 1 uf electrolytic capacitor.
- One 2.2 uf electrolytic capacitor.
- One 4.7 uf electrolytic capacitor.
- A breadboard.
- A 9-volt battery.
- A battery clip.

In a synthesis context, the lag processor is a circuit that goes by many names: slew limiter, portamento, glide. . . . You typically find one behind the knob on your keyboard that makes notes slide from one to the next. Freed from the constraints of this stereotypical usage, however, the lag processor can be so much more: an envelope generator, a simple filter, even an LFO.

The lag processor appeared alongside some of the earliest synthesizer modules: the Moog 928 Sample & Hold from 1965 included a glide section. Serge (1973) and E-Mu (1976) pioneered more advanced functionality with their Slew Limiter designs, including individual rise and fall times, response curve adjustment, and even voltage control. The Slew Limiter in this chapter is designed with the 1970s Slew Limiters in mind but includes bells and whistles to pad out its flexibility and encourage experimentation (Figure 30.1).

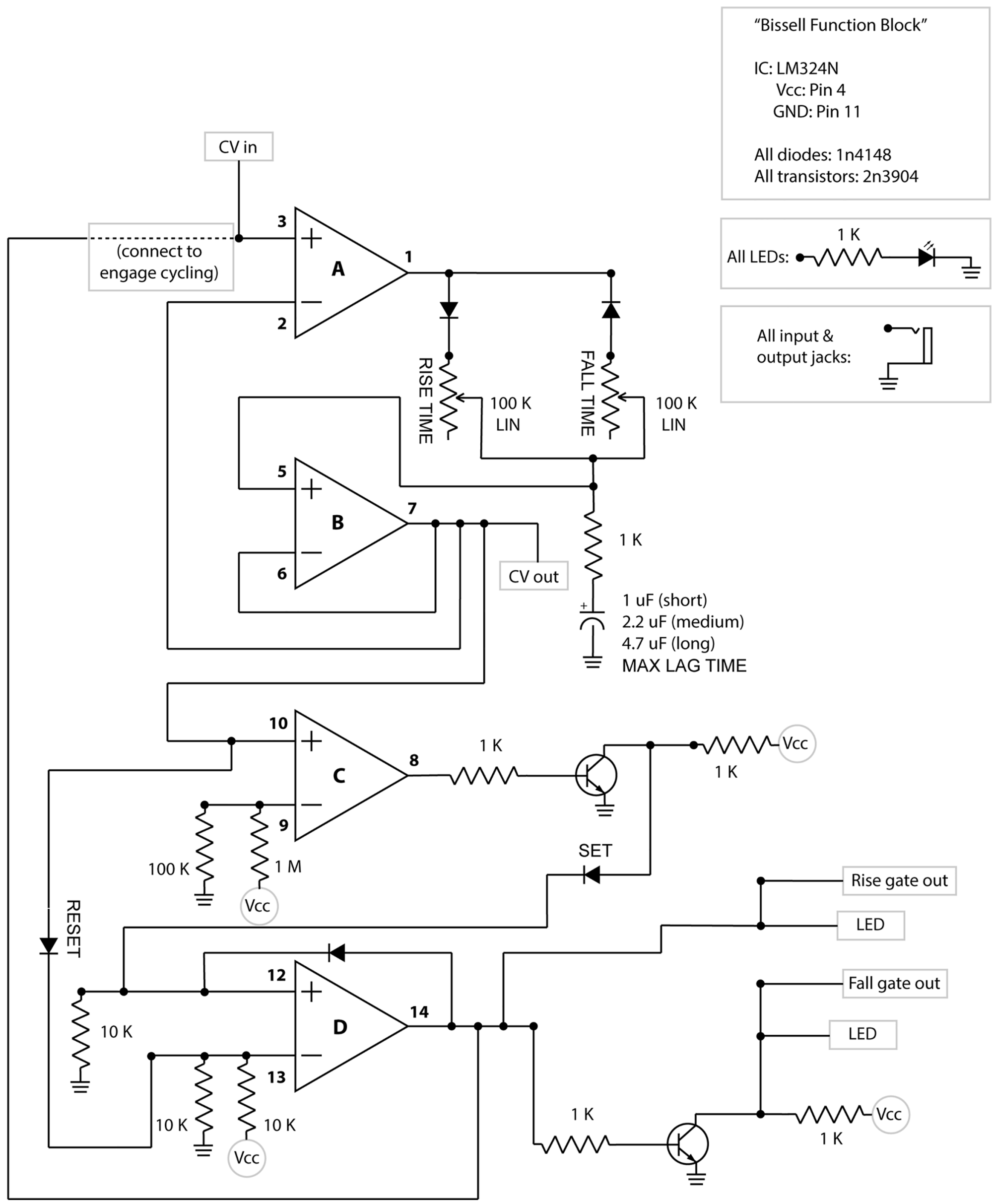


Figure 30.1 Bissell Function Block schematic.

I call this circuit the *Bissell Function Block* for three reasons:

1. Harry *Bissell*, whose "Morph Lag" circuit has been a staple in the DIY synth community for years, directly informs the rise and fall time components of this work.
2. The *Function*, a contemporary synthesizer module by Make Noise designed by Tony Rolando, whose gate outputs inspired the ones featured here.
3. Building *blocks*, conceptually, from which this circuit is comprised. Any portion of the design can be abstracted and incorporated into your project as needed.

This entire circuit centers around a single quad op amp, the LM324N workhorse (see Chapter 23 for an introduction to op amps). This chip contains four identical amplifiers, which will be put to various uses: the slew limiter mentioned prior, a voltage follower (or buffer), a comparator, and a flip flop. A couple of NOT gates (like the CMOS inverters introduced in Chapter 13) are built with transistors. If this looks like a tall order, remember, you can always just build what you need now and leave out what you don't.

The slew limiter and voltage follower use op amps A and B (Figure 30.2). For this section, you will need:

- One LM324N quad op amp IC.
- Two 100 kOhm linear potentiometers.
- One 1 kOhm resistor.
- Two 1n4148 diodes.
- One 1 uf electrolytic capacitor.
- One 2.2 uf electrolytic capacitor.
- One 4.7 uf electrolytic capacitor.
- Two jacks (optional).

The slew limiter is built around an RC (Resistor-Capacitor) network, with a buffered output. The RC network creates an adjustable lag time, and the buffer ensures that the voltage at the output matches the one at the input. For flexibility, the circuit incorporates individual rise and fall time controls, as well as the ability to set the maximum lag time.

Leaving the negative input (pin 2) unconnected for now, connect the output (pin 1) to our various timing components: rise time, fall time, and the maximum lag time.

The rise and fall time controls are isolated from one another with diodes, one forward biased (rise) and the other negative biased (fall), each feeding a 100 kOhm pot. The arrangement of these diodes allows only increasing (rise) or decreasing (fall) voltages to pass through to their potentiometers, which sets the respective rates for rise and fall. The opposite lugs (ears) of these pots should remain unconnected. The wiper lugs (noses) of these pots are summed through a 1 kOhm resistor and then a capacitor, which is connected to ground—this forms the RC network.

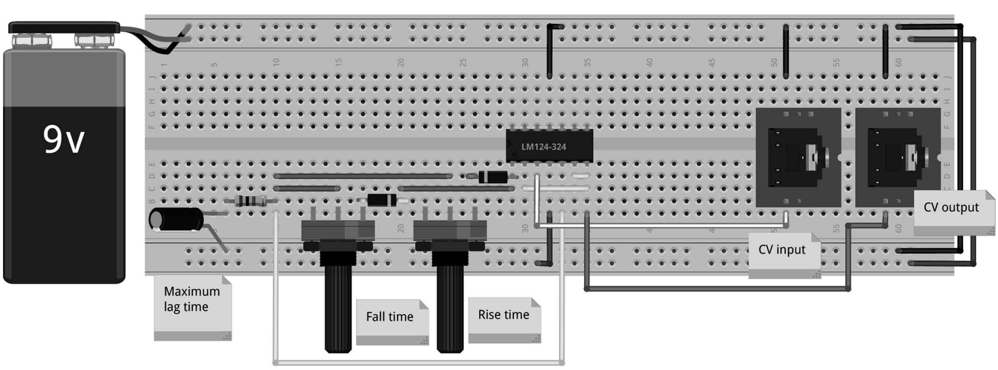


Figure 30.2 Slew limiter and voltage follower.

The value of this capacitor determines the overall time base for these rates: experiment with different values and find one that works best for your needs. A 1 uf is a good value for fast slew times, 2.2 uf for medium ones, and 4.7 uf handling longer time ranges. If you can't make up your mind, put all three on a triple-throw switch and select them at will.

Now also connect the wiper lugs to the positive input of op amp B (pin 5). To buffer the output with a voltage follower, connect the output of op amp B (pin 7) to the negative input of the same amp section (pin 6)—this ensures that the voltages at the output (pin 7) match exactly those at the input (pin 3). Finally, close the feedback loop of op amp A by connecting the output of op amp B (pin 7) to the negative input of op amp A (pin 2). This feedback loop (pin 7 to pin 2) helps linearize the rise and fall times of the lag processor, and with this connection you have completed the first building block of our module.

Send a stepped control voltage (such as the output of a sequencer, from your modular synth or the one in Chapter 20) to the positive input of op amp A (pin 3). Connect the output from op amp B (pin 7) to a voltage-controlled oscillator to test the lag variables—you can breadboard one with the 4046 Phase Locked Loop as shown in Chapter 20 or use any VCO in a modular synthesizer. Adjust the rise and fall times with the 100 k pots and select the maximum lag times with the cap of your choice.

If all you need is a slew limiter, you're there. Congratulations.

Let's try some other input sources. For example, patch a gate signal or slow square wave into the CV input (pin 3), instead of the output of a sequencer. Connect the output of the Slew Limiter to a VCO, a VCA, or a VCF. Observe how the Slew Limiter transforms the on-off/high-low nature of the gate into a more gradual waveform, which rises from 0 volts at a speed set by the rise time pot, holds at a maximum value while the gate remains high, and falls back to 0 volts once the gate is removed at a rate set by the fall time pot. What's another term for a control voltage that rises, holds, and falls? An envelope! Not only have you built a lag processor, but you've also built an envelope generator.

Try connecting an audio-rate signal to the positive input of op amp A (pin 3), something with a good deal of harmonics (such as a square or saw wave). Listen to the output of op amp B (pin 7) and set the rise and fall times to their minimum values (i.e., the shortest lag times). Increase the rise and fall times and note how the sound at the output (pin 7) loses harmonics as the times are increased. Just as with the envelope example prior, an audio-rate square wave can be shaped into a triangle wave using a slew limiter, with the rise and fall times transforming the abrupt sides of the square into the linear slopes of a triangle. What is another term for something that removes harmonic content from a signal? A low-pass filter! A slew limiter is that as well. (Note: as the rise and fall times are increased, the output of this simple filter will decrease in level. For tweaking audio signals, small capacitors work best in the RC network—try values even smaller than 1 uf.)

In these examples, the slew limiter's rise section is only engaged for as long as a rising voltage is present at pin 3. When it disappears, the fall section takes over. In the case of a gate source, it will rise while the gate is high and fall once the gate is low. If we could create a gate source that is synchronized with the rise and fall times—such that it remains high until rise completes its journey up from 0 volts to its maximum level, and then goes low while fall decreases from this point back down to 0 volts—we could cycle the slew limiter between these two states, rising and falling, indefinitely. What is another term for a modulation source that rises and falls indefinitely? A low-frequency oscillator, or LFO! Let's add this functionality to our circuit.

Fortunately, we have two more op amps at our disposal, which is just enough to make that happen. To add the cycling LFO behavior described prior, we'll need the following additional parts:

- Six 1 kOhm resistors.
- Three 10 kOhm resistors.
- One 100 kOhm resistor.
- One 1 mOhm resistor.
- Two LEDs.
- Two jacks (optional).
- Three 1n4148 diodes.
- Two 2n3904 NPN transistors.

To recap: op amp A is forming the slew limiter, taking an incoming signal and making its changes from one value to the next more gradual, and op amp B is both buffering this signal and linearizing the response of op amp A's slew times.

Creating a timing signal that is related to the output of the slew limiter requires us to compare the activity of the circuit against a reference voltage: if it is *above* that amount, this will be reflected in a *high* gate output; if it is *below* that amount, the result will be a *low* gate output. This is the action of a circuit known as a comparator, which is what we will be using op amp C for (Figure 30.3).

Patch the output of op amp B (pin 7) to the positive input of op amp C (pin 10). We will compare the voltage level of this signal with the voltage level present on the negative input (pin 9) of the same amp section, which will be set to a low voltage using a voltage divider circuit (two resistors connected to pin 9, with one tied to ground and the other to the positive voltage of the power supply—the divisor value is determined by the ratio of their resistance values). Connect a 100 kOhm resistor to ground and a 1 mOhm resistor to the positive voltage source, which produces a voltage on pin 9 that

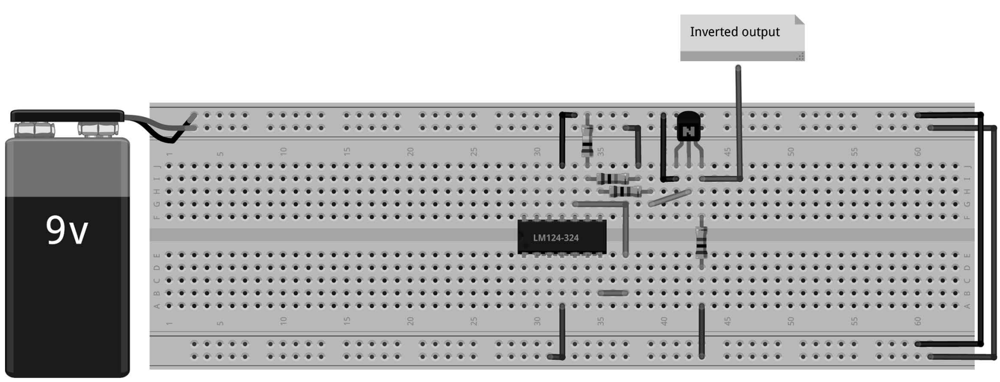


Figure 30.3 Comparator.

is one-tenth the voltage of the power rail (1,000,000 / 100,000 = 1/10—remember your algebra class?). So long as the output of the slew limiter is above this amount, the output (pin 8) will be high.

Why do we want an output that is almost always high? Because what we are really after is the small moment at the beginning when the output is low. By inverting this output, we can have access to a short pulse at the very beginning of the slew limiter's travel, between 0 volts and one-tenth of the supplied power, which will be very helpful in a moment. But first, how do we invert the output of the comparator? With an inverter or NOT gate.

A NOT gate only deals with logic signals—high and low states (think: on and off)—and will invert one value for the other. A high gate at its input will be a low gate at its output and vice versa. Earlier in the book, we used sections of a CMOS Schmitt Trigger for their inverter functions as we built our first oscillator (Chapter 13), but now we'll build one with a single NPN transistor and a pair of 1 kOhm resistors.

Route the output of op amp C (pin 8) through a 1 kOhm resistor to the base of the transistor. Connect the emitter to ground and the collector to the other 1 kOhm resistor and from there to the power rail. (If you're not familiar with transistors, look carefully at the breadboard illustrations.) The output of this NOT gate will be available at the collector leg and, as expected, will provide us with a high gate when the output of the slew limiter is below one-tenth the voltage of the power rail.

Imagine a triangle shape, rising and falling. What we have just created is a moment related to the point at which it begins rising—a starting point. Our goal again is to create a gate that is high while the lag processor is rising and low when it is falling. This start trigger will allow us to synch the rising action to this point.

The next step is to create something that alternates between high and low states and to use this start trigger to create the switching from low to high. Again, think of the triangle shape: the start trigger is at the beginning of the action upward. Once the triangle reaches its peak, we want to switch the gate to a low state and let the slew limiter's fall time dictate the downward trajectory back to 0 volts.

This type of logic switching circuit is called a flip flop (Figure 30.4). It alternates between high and low states at its output in response to a voltage received at a Set input (switch to high) or a Reset input (switch to low). We'll use our remaining op amp (D) to build one of these.

The start trigger will provide our Set input. Connect the output of op amp C's NOT gate (the collector leg of the transistor) to the anode leg of a diode; connect the cathode leg to the positive input of op amp D (pin 12). Connect the positive input of op amp D (pin 12) through a 10 kOhm resistor to ground.

For the flip flop's Reset signal, connect the output of the slew limiter (pin 7) through a forward-biased diode to the negative input of op amp D (pin 13). We add a voltage divider reference signal to the same negative input (pin 13), using 10 k resistors tied to ground and the power supply respectively. Finally, the output of op amp D (pin 14) will be fed back into its positive input (pin 12) through a diode, with pin 14 on the anode and pin 12 at the cathode ends.

What does this peculiar circuit do? The voltage divider sets the negative input (13, the Reset pin) at half the voltage of the power rail, and the reverse-biased diode

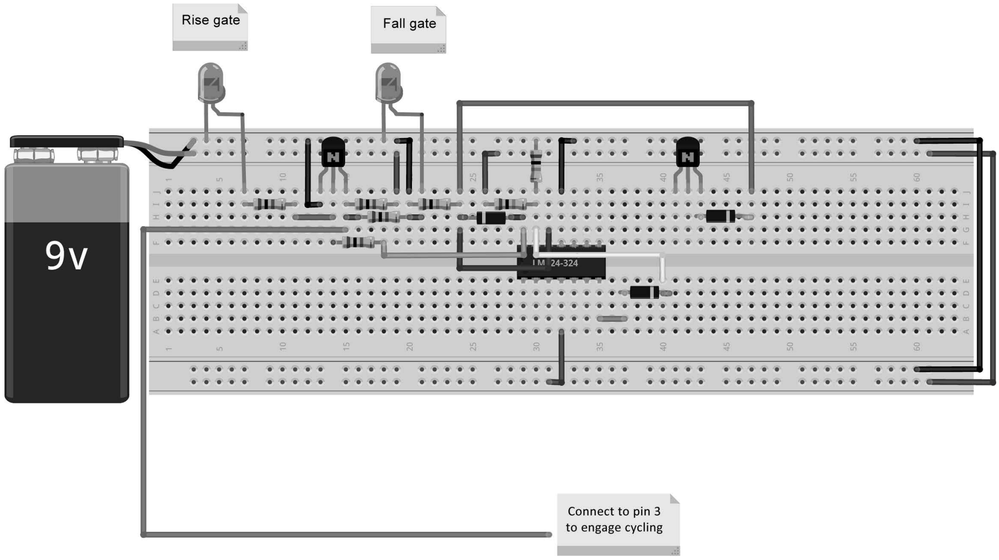


Figure 30.4 Flip flop.

feedback loop ensures that the positive pin (the Set pin) remains low, so long as the Set input does not see a sufficiently high voltage. Once the Set pin *does* receive a high voltage (one higher than that set by the voltage divider), the output is locked in a high state and remains there. However, once the Reset pin receives a voltage above half the power supply, this forces the circuit to return to its default state and switch its output low again. Flipping high, flopping low.

To make this flip flop start flipping and flopping, we need to engage cycling. To do this, simply connect the output of the flip flop (pin 14) to the positive input of the slew limiter (pin 2). Once this connection is made, the slew limiter will begin endlessly rising and falling with the rise and fall time pots controlling the rate and shape of oscillation. Cycle time not long enough for you? No problem! Simply swap out the capacitor in the RC network of op amp A for a larger value. Try interrupting the pin 14/pin 2 connection with a switch to create on-demand cycling!

Finally, we can add indicators and output connections for the action of the Flip Flop to give us a visual readout of the behavior of the cycling (and who doesn't like LEDs?), as well as access to gate outputs reflecting the rising or falling stages. The rising indicator is easy—that's the output of the flip flop. Connect an LED in series with a 1 k resistor to pin 14 to light on the rising action gate signal (connect the LED cathode to ground). The falling stage happens when the Flip Flop goes low, so if we want a lit indication of this state, we need to build another NOT gate.

As before, connect the signal to be inverted (pin 14) to the base of an NPN transistor and connect the emitter to ground and the collector to the power rail via a 1 k resistor. The collector leg is our inverted output: connect an LED in series with a 1 k resistor to the anode (with the cathode to ground) and access this collector pin for a timing signal related to the falling portion of the cycling Slew Limiter. (Figure 30.1, at the head of this chapter, shows the full circuit.)

CHAPTER 31

Sounds From Neural Networks

WOLFGANG SPAHN

You will need:

- Resistors: two 330 kOhm, four 1 kOhm, one 10 MOhm.
- Assorted capacitors, 100 pf to 1,000 uf.
- LED.
- Op amp TL074.
- IC socket, 14 pins.
- 3-pin header strip.
- Jumpers.
- Three 2-pin headers, female.
- Two 4-pin headers, female.
- 2-pin socket header, female.
- 2-pin header, female 90°.
- 3-pin header, female 90°.
- 2-pin header, male 90°.
- 3-pin header, male 90°.
- 2,100 kOhm potentiometers.
- A copper stripe board.
- A paper printout of the Confetti Neuron (from the website).
- Double-sided tape.
- Soldering iron and tools.

INTRODUCTION

There comes a time in the life of most devotees of electronic sound when one acknowledges the fact that circuits make very cool noises but computers often can produce more interesting control structures. Many of us are content to link our computers to our circuits (not so difficult to do these days, thanks to things like Arduinos—see Chapter 33) and enjoy the mix. But others recoil from the prospect of analog and digital cohabitation. One solution to this conundrum lies in neural networks—a form of analog computing modeled on the behavior of synapses in the brain.

CONFETTI, AN ANALOG COMPUTER

Analog computation was surprisingly widespread until the 1970s, when digital technologies took over. Usually, an operational amplifier (op amp) is the core of most analog computers. Apart from being capable of high amplification, an op amp is able to perform various arithmetic operations such as adding, integrating, multiplying, and more (see Chapter 23). Depending on how one wires up the op amp, one can add 3 volts to 2.5 volts to get 5.5 volts as a result. It is also useful for carrying out integration, which is most attractive for designing analog synthesizers. For instance, the integration of a square wave will be a triangle wave and the integration of a triangle wave will be a sine wave. Often these analog operations offer a simpler, more elegant solution to a problem than lines of computer code.

"Confetti" is a multi-connect modular electronic computation system. Each single module is capable of one specific operation. When combined, these modules can solve quite complex formulas, such as strange attractors.[1] Besides its unique mathematical functionality, each board comes with a number of inputs and outputs that can be patched with normal breadboard jumpers. At its core all boards use the standard operational amplifiers TL072 (two op amps on a chip) or TL074 (four op amps on a chip).

BUILDING THE CONFETTI NEURON

To build your own Confetti Neurons simply follow these steps:[2]

- Download and print the paper design from the website: "Confetti501_Confetti Neuron_20.tif" (Figure 31.1).
- Use some double-sided tape to stick it on a one-sixth of a copper stripe board. Copper stripe board is different from the breadboard-mimicking printed circuit boards used elsewhere in this book (introduced in Chapter 14): the copper traces form continuous straight lines; circuit pathways are configured by cutting and bridging these lines (Figure 31.2). As you can see, the paper printout guides you through the correct configuration of the circuit.
- Poke some holes through the paper for all the components.
- Make the designated cuts through the copper stripes (follow the red lines).
- Put in the components and solder them (Figure 31.3).

CONFETTI NEURON

Developing neural networks that imitate the complexity of the human brain starts with a simple core module, which can be multiplied until it resembles a complex structure. A possible model for neurons within the brain would be a device that is able to output or suppress a signal ("fire" or "not-fire") when a certain input level is reached. Connecting a great number of these devices would result in them performing complex functions or even simulating life. (Nevertheless, it is still just

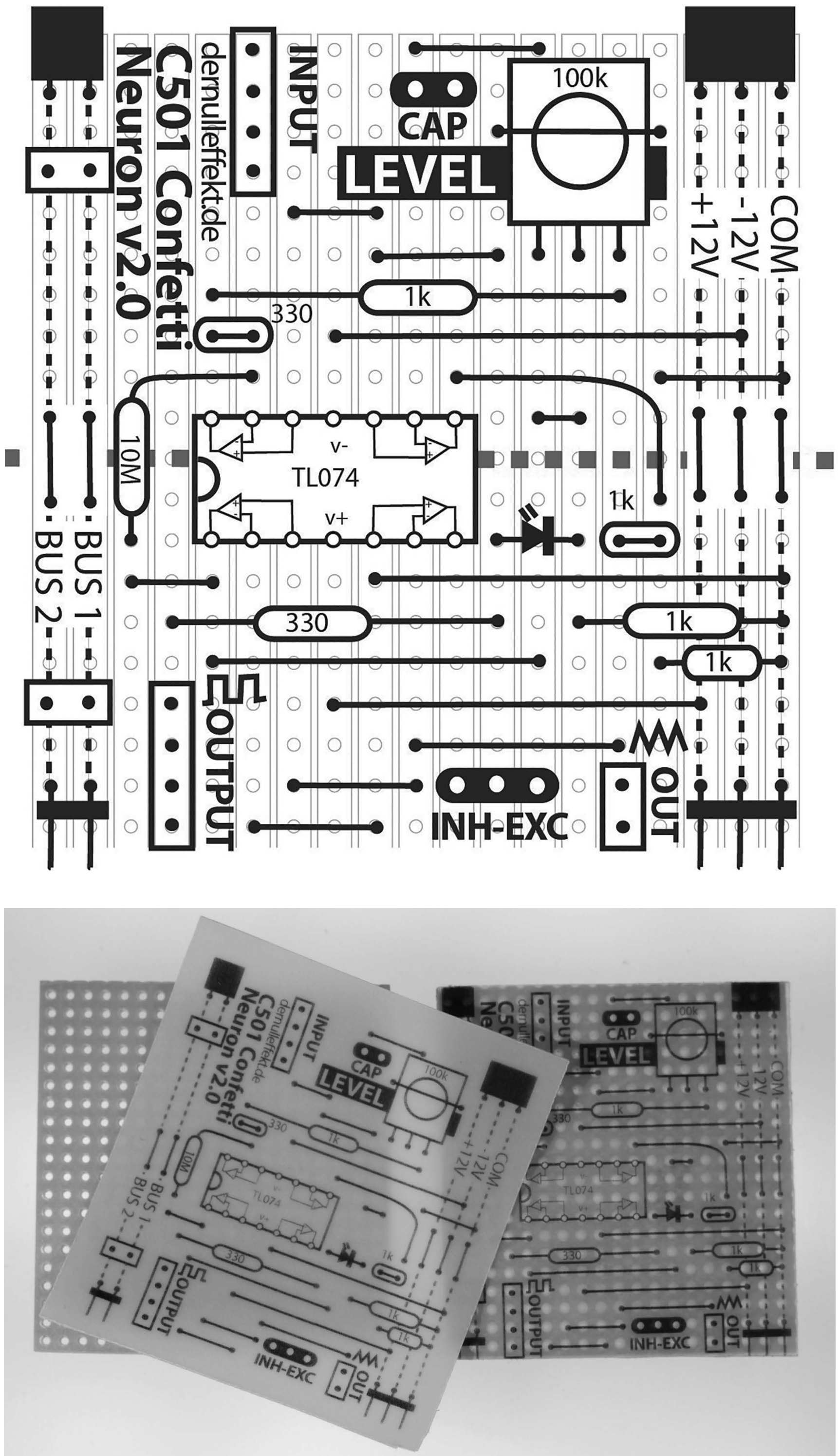


Figure 31.1 Confetti Neuron paper template; Confetti Neuron paper template and strip board.

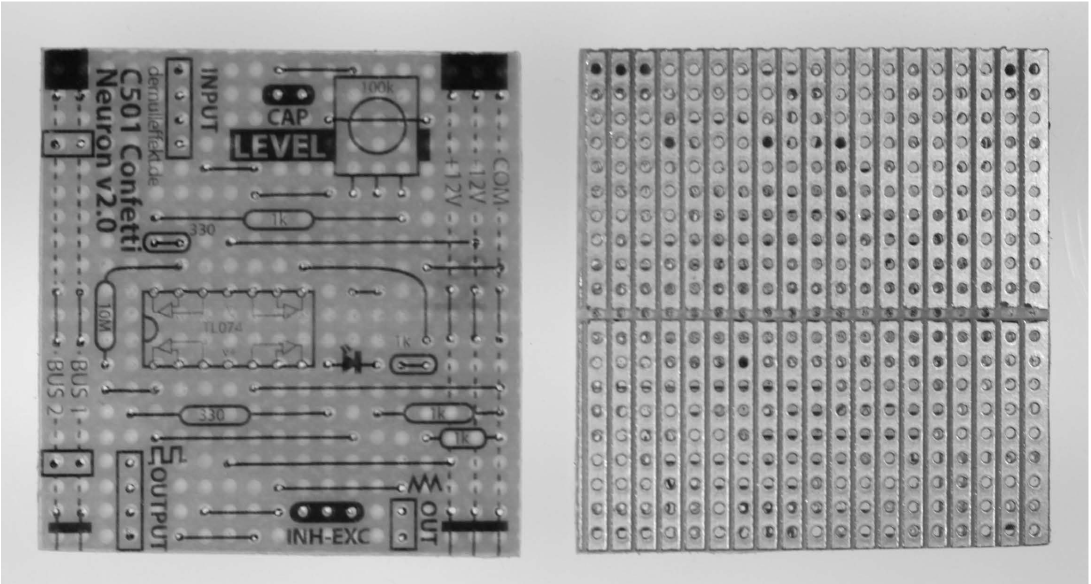


Figure 31.2 Confetti Neuron paper template laminated to strip board with cut traces.

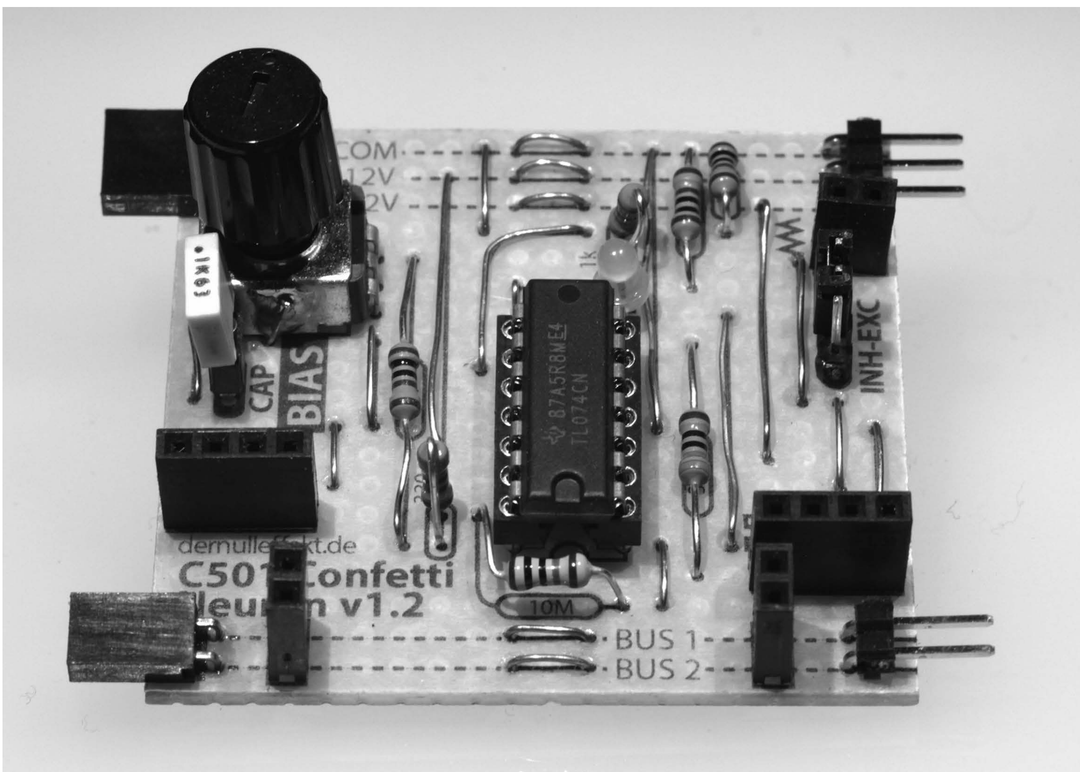


Figure 31.3 Confetti Neuron with components soldered into place.

a *model* of the human brain, and it is worth bearing in mind the sage words of Norbert Wiener: "The best material model of a cat is another, or preferably the same, cat."[3])

The Confetti Neuron is a simple circuit containing two different neurons: an *excitatory* neuron and an *inhibitory* one. The task of the excitatory neuron is to generate a signal, i.e., to fire, while the inhibitory neuron's purpose is to prevent that, i.e., to not fire. On the receiving end of each neuron (the input), output signals from other neurons are accumulated. As soon as a certain level (the so-called action potential of the neuron) is reached, the neuron should fire.

Each Confetti Neuron consists of four different modules: a summing amplifier,[4] an integrator,[5] and a comparator[6]—a non-inverting comparator for the excitatory neuron and an inverting one for the inhibitory (Figure 31.4). When combined, these four models fire or not-fire when a certain input level (action potential) is reached. Both comparators are built into each Confetti Neuron, which can therefore be used as either an excitatory neuron or an inhibitory neuron, depending on the setting of the jumper (EXC or INH). Like all Confetti boards, it comes with three power rails and two buses, plus it provides a number of inputs and outputs for connecting each Confetti Neuron to all other neurons within the network. The inputs also allow for connecting various kinds of signals, devices, and voltage sources that act correctly together as long as the input voltage stays within the power limits. Similar to neuro-sciences, where the strength of the connection is defined as *weight*, in this case the value of the input resistor one may call the weight of the connection.

The output of each neuron generates an on/off signal depending on the state of the neuron. When the state changes fast enough, the result looks like a square wave. The Confetti Neuron provides an additional point where one can receive a triangle wave. As this is internally connected to the output of the integrator, it is possible to see

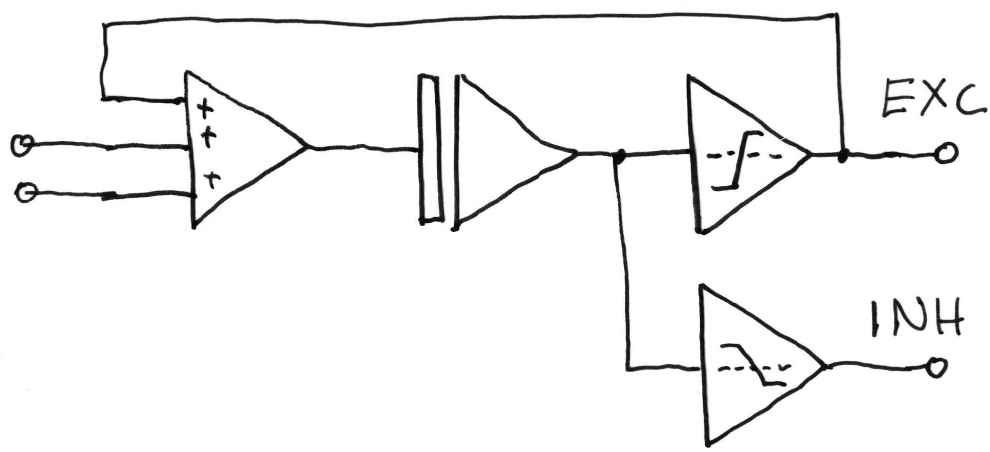


Figure 31.4 Confetti Neuron structure.

or listen to the integration capacitor being charged and discharged. Thus the output provides a sound signal source in the form of a triangle wave that can be easily shaped into a sign wave, and it also delivers a controlling signal that offers an alternative to on/off.

Circles are commonly used to symbolize neurons. The excitatory neuron is represented by an arrow at the output and the inhibitory neuron by a small circle.

POWER UP

Unfortunately, the analog electronics used here need negative power in addition to the usual positive one since inverted signals are often part of the process (refer to Chapter 23 for a discussion of powering op amps). For example, within the inverting integrator the new (inverted) signal will become negative only if negative power is provided. A positive and a negative power rail (for example +12 volts or −12 volts) can be provided by the Confetti Power Board.[7] The easiest way to generate a split supply is with two 9-volt batteries. A connection of one plus pole to one negative pole of the other battery provides a +9 volt and −9 volt power supply—this new third connection one calls COM, like common (the two other connections are +9 volts and −9 volts). You can solder a red wire from one 9-volt battery clip to a black one from another (this becomes COM), then the three wires provide your voltages (red is always plus) (Figure 31.5). Careful handling is key here as connecting reversed polarity will damage the chips.

Figure 31.5 Powering the Confetti Neuron from two 9-volt batteries.

NEURAL OSCILLATOR

A neural oscillator always consists of two Confetti Neurons: an excitatory neuron and an inhibitory neuron. Each output of a Confetti Neuron is connected to the other one's input via an inserted resistor (the square wave output is always used for patching). These two neurons combined form a form of Bicore Oscillator[8]—i.e., an oscillator with two oscillation cores, each based on capacitors. The result will be that each neuron oscillates with the other, both in the same frequency but with one signal shifted in phase with respect to the other (Figure 31.6).

This analog neural oscillator comes with four different outputs (each neuron provides two): two square wave outputs and two triangle wave outputs. All four have the same frequency, but the phases of the two waves of the second neuron are shifted slightly in respect to the first neuron.

Lower frequency can be seen at the built-in LED of the Confetti Neuron, but at higher frequencies these LEDs appear always on. To listen to the oscillation, patch one of the four outputs to the Confetti103 Buffered Output[9] board and patch it to your amp or mixer. You can also patch directly from the neuron output and bypass the buffer. The square wave output sounds like the CMOS oscillators elsewhere in this book (see Chapter 13) or the classic Atari Punk Console, rich in harmonics. The triangle wave comes with softer harmonics and provides a mellower sound (Figure 31.7).

There are two way to change the frequency of the oscillator: by changing the level on the neuron or changing the value of the patching resistor between the two neurons. For the first method, rotate the level knob on the neuron patched to your mixer while the other one is set to zero level. For changing the frequency with the inter-cross-resisters, use a potentiometer or—even better—a stereo potentiometer of

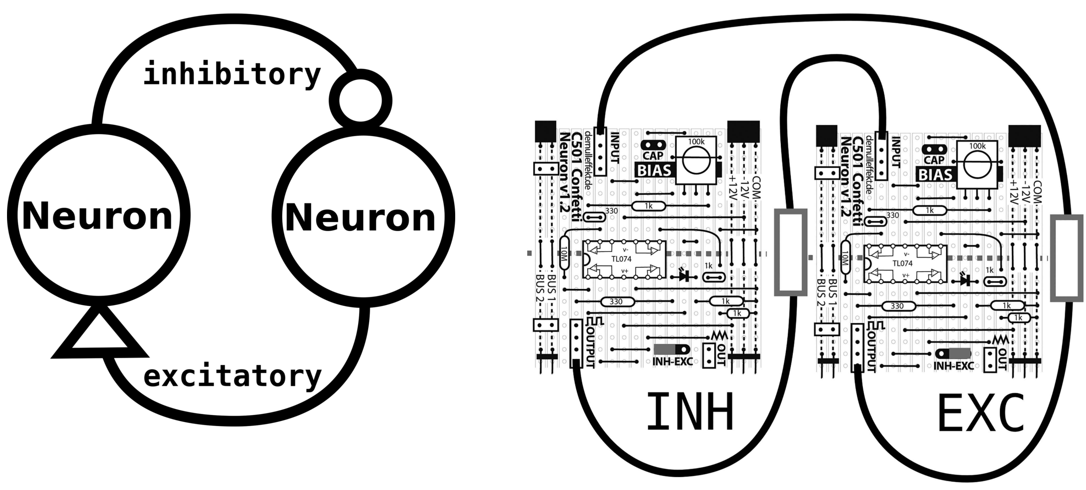


Figure 31.6 Confetti Neuron oscillator.

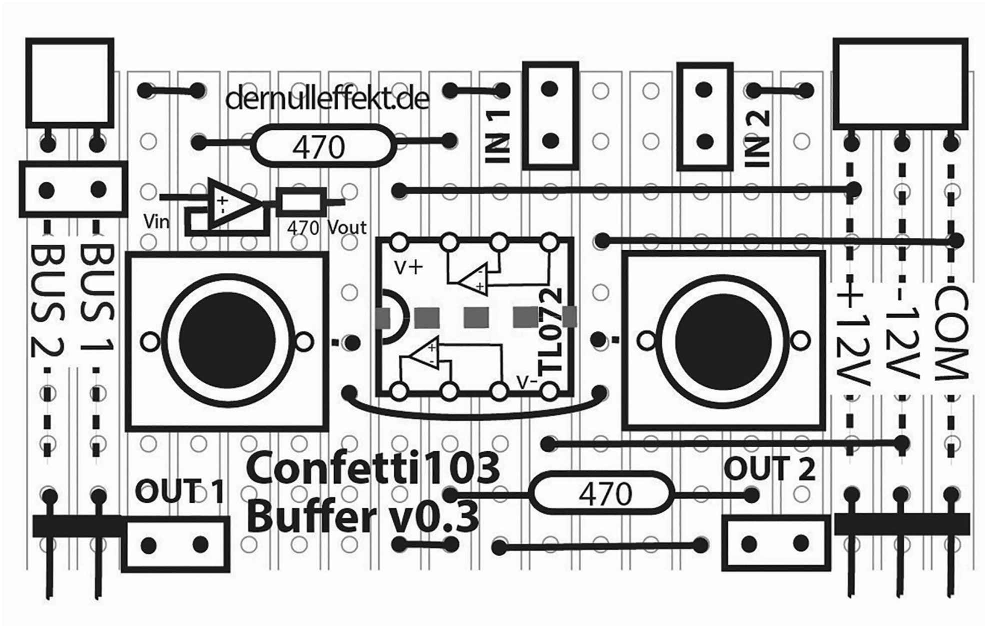


Figure 31.7 Confetti Neuron oscillator buffered output.

1–2 mOhm. The stereo potentiometer keeps both resistor values equal while changing them, resulting in a triangle wave that preserves constant level while the frequency changes. For patching the neurons with a potentiometer, an additional Confetti Bread Board[10] will be helpful (Figure 31.8).

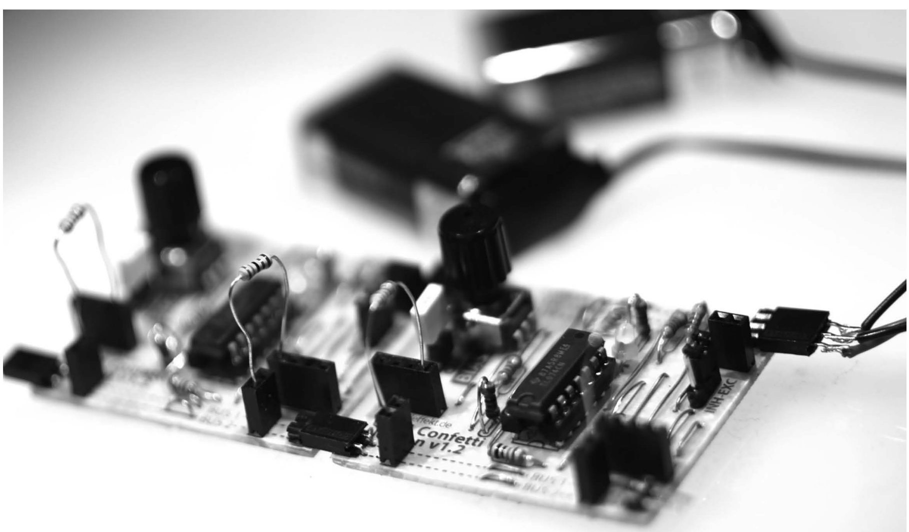


Figure 31.8 Confetti Neuron oscillator and array.

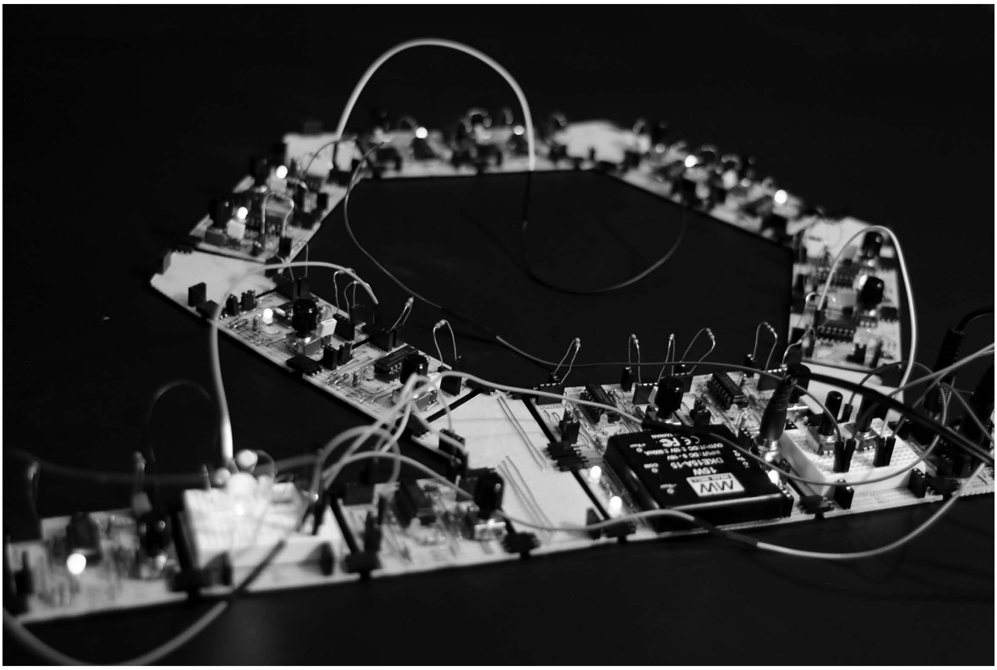


Figure 31.8 (Continued)

For more applications of the Confetti Neuron, go to the Confetti Neuron website: confettineuron.dernulleffekt.de

NOTES

1. https://en.wikipedia.org/wiki/Attractor#Strange_attractor
2. Also refer to http://paperpcb.dernulleffekt.de/doku.php?id=about:manual
3. Arturo Rosenblueth and Norbert Wiener, "The Role of Models in Science." *Philosophy of Science* (October 1945). P. 320.
4. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti704_non_inverting_summer
5. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti705_inverting_integrato
6. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti108_non_inverting_comparator
7. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti003_power_brick
8. http://solarbotics.net/bftgu/starting_nvnet_bicore.html
9. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti103_buffered_output
10. http://paperpcb.dernulleffekt.de/doku.php?id=analog_computer:confetti302_breadboard

PART 4

Computing

CHAPTER 32

Sharing Traces
Designing and Fabricating Your Own Printed Circuit Boards With Fritzing

EDUARDO F. ROSARIO

INTRODUCTION

You've spent hours moving things around on the surface of a breadboard. You've soldered up prototypes using generic circuit boards (Chapter 14), maybe knitted some as well (Chapter 16). But one day you might decide that it's time to consolidate all your work into an easily reproducible object: a printed circuit board (PCB). Assembly time is reduced significantly with a design-specific PCB: a circuit that took hours to hand wire can now be soldered up in 10 minutes. A PCB makes sharing your own design with others much easier. If you have a design that you frequently use, such as an amplifier or a power supply, you can easily have multiple copies available for use on various projects. And they look so cool.

Before sending a design to manufacture, make a prototype yourself. There are few things more frustrating than getting 100 boards in the mail and discovering a mistake in your design. These mistakes sometimes are simple things that you can correct in assembling (such as the size of the mounting holes), but sometimes they force you to change a part, cut traces, or throw the boards away. Ouch!

We'll start by designing a PCB for the 74C14 Hex Schmitt Trigger oscillators from Chapter 13. By now you should have some understanding of how schematics work. If not, don't worry, because by the end of designing your first board, your acquaintance with cryptic representations of electronic circuitry will grow significantly.

SOFTWARE

The first step in transferring your design to a PCB is importing what's on your breadboard into a computer. For this we will use what is called an *electronic design automation* software, or EDA. There are dozens of EDAs available, with different advantages and costs. Popular ones include KiCad, Fritzing, Altium's CircuitMaker, and Autodesk's

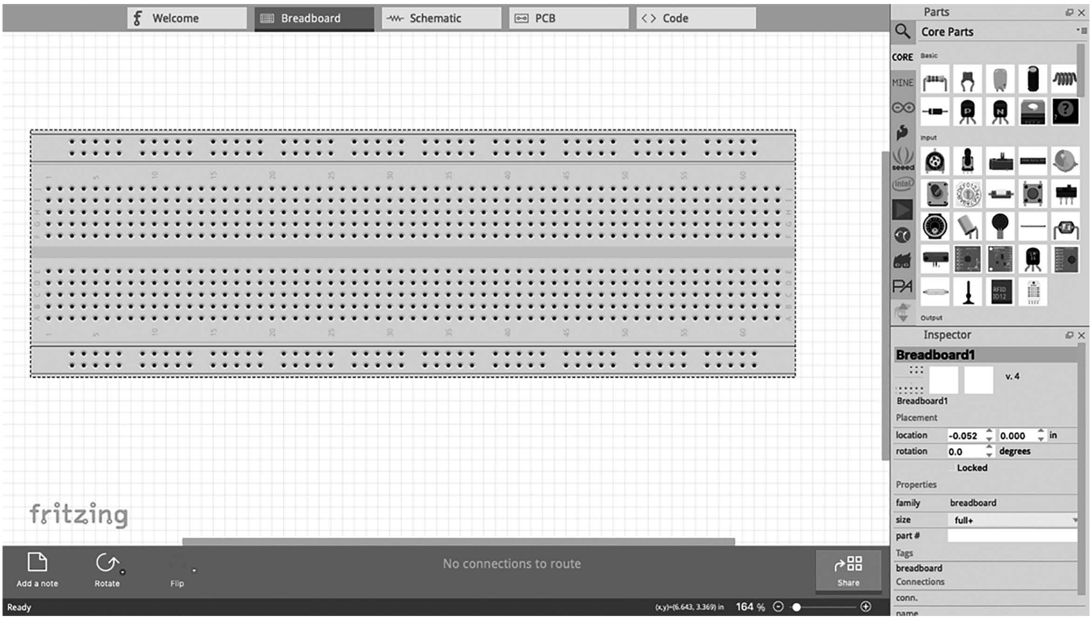


Figure 32.1 Fritzing breadboard view.

Eagle. I encourage you to research which is best suited for your project. For example, Fritzing has a breadboard view (Figure 32.1) in which you can put your PCB design together in the same way that you cobble together the circuit on a breadboard—this makes it more intuitive than other EDAs and, therefore, an attractive option for artists, hobbyists, and hackers (it's also free).

FRITZING

Fritzing is an open-source EDA. Go to fritzing.org/download and select the appropriate version for your computer. Installation is fast and easy, so you shouldn't have any problems. If you are an OS X user, you may have to check your Security & Privacy settings since it is not downloaded from the App Store.

Open Fritzing (Figure 32.2). You will notice five tabs across the top of your screen: Welcome, Breadboard, Schematic, PCB, and Code. This is the View Switcher. It will help you navigate the different aspects of your project. Below this to the left a panel called Recent Sketches lets you create new or open ongoing projects. Click on New Sketch. A new window will pop up on the breadboard view. Before you do anything, go to File and save it. The default location will be the Fritzing folder on your Documents directory with the file extension ".fzz"; this file will include all the data for your project, or "sketch," as Fritzing calls them (Figure 32.3).

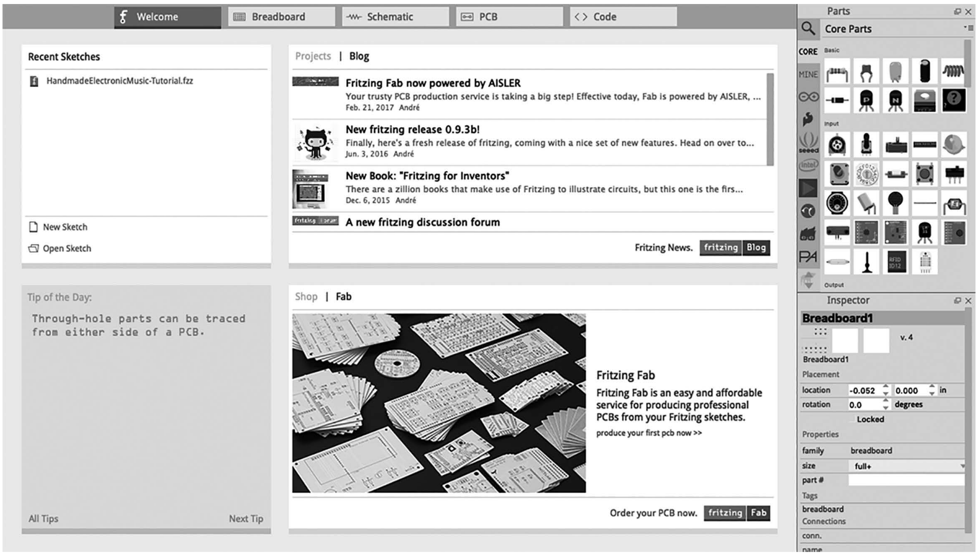


Figure 32.2 Fritzing welcome view.

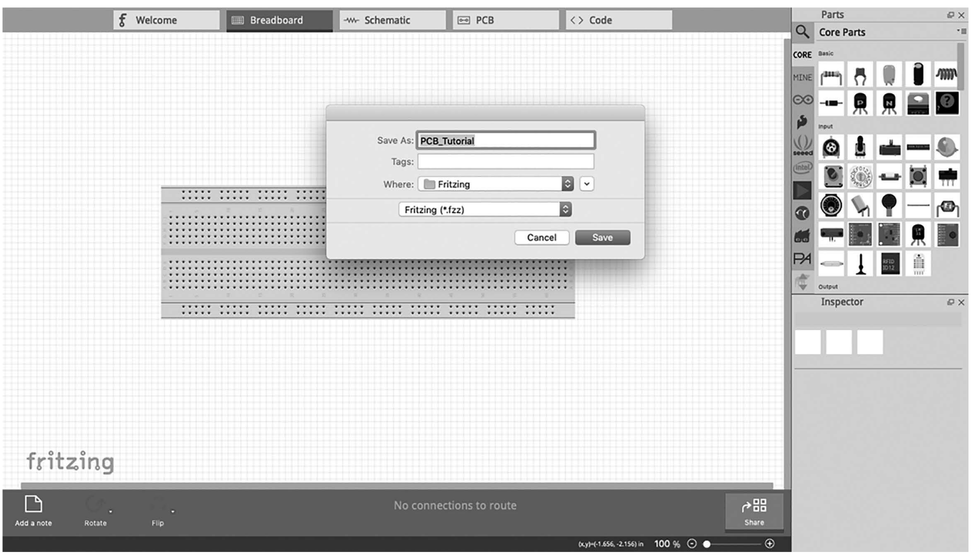


Figure 32.3 Saving sketch.

Let's start with something very simple: linking the + power and ground buses on the breadboard (Figure 32.4). Hover your cursor over any hole in the power rails and you will notice that the hole is highlighted; now simply click and drag and a wire will appear. Connect the + power bus at the top of the breadboard to the one at the bottom, and do the same for the ground buses, so you have + and − available on each side of the board. Every time a wire is selected, or a new one placed, important information will appear on the Inspector menu (Figure 32.5), at the bottom right, below the Parts

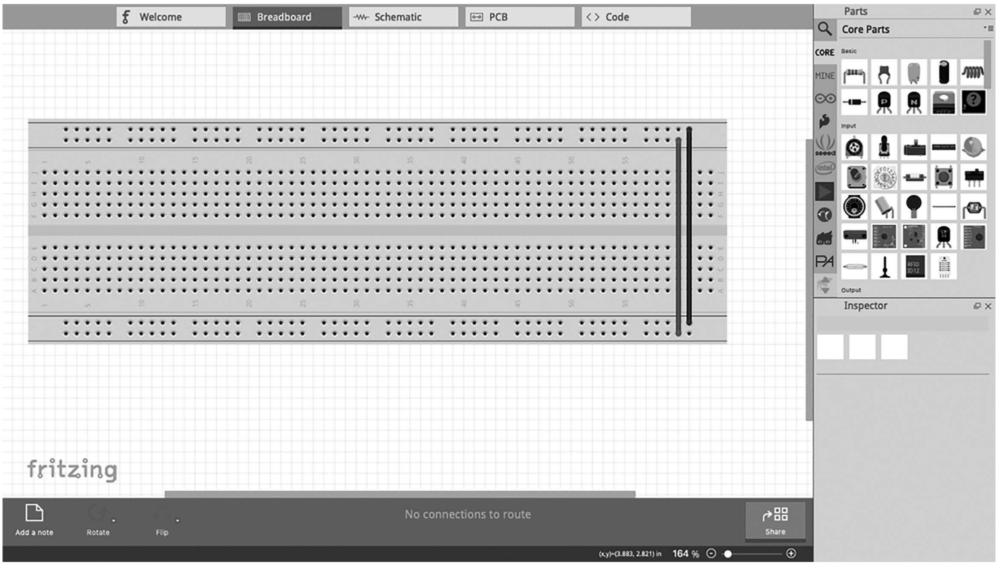


Figure 32.4 Adding power and ground.

Figure 32.5
Inspector window.

menu. The Inspector panel is an essential component of working with Fritzing. Go ahead and change the name of the wires to "power" and "ground" and the colors—in the Properties tab—to red and black respectively (Figure 32.6).

Let's add an integrated circuit. If you scroll down on the Core Parts menu, you will see the ICs (Figure 32.7). There are a few options, but grab the first one simply

Figure 32.6
Naming a wire.

Figure 32.7
Core parts IC.

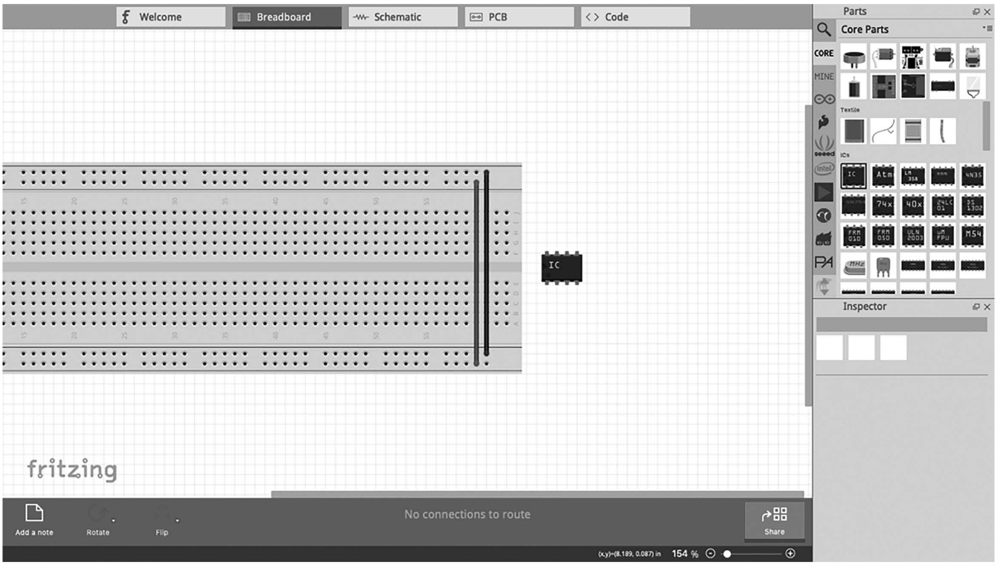


Figure 32.8 IC next to breadboard.

labeled IC and drag it near the breadboard on the workspace. Don't place it *on* the breadboard yet (Figure 32.8). This part defaults to a generic 8-pin dual in-line package IC. But taking a quick look at the inspector reveals how much this part can be modified. For example, on the Package tab (Figure 32.9), it can be changed to other formats,

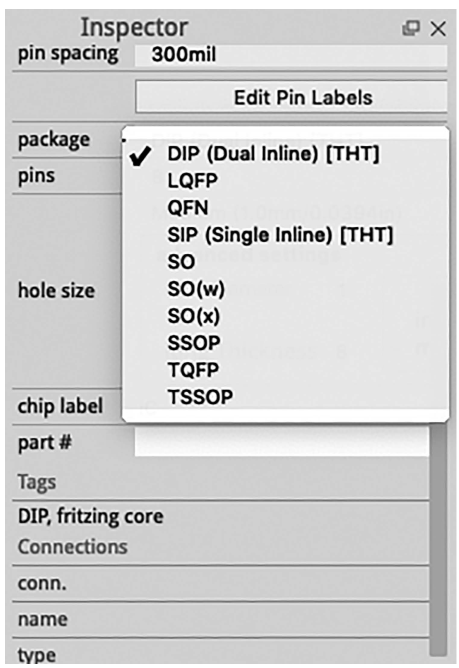


Figure 32.9
Package tab.

such as SO (small outline), with different versions on the variant tab below. On the Pins tab, we can change how many pins this chip has. Let's keep it in its DIP package and change the number of pins to 14. Now, place it on the breadboard as you would while prototyping (Figure 32.10). Columns connected to the IC on each side will be highlighted green. Go back to the Inspector and on chip label type "74C14" since that's the chip we're using to build our oscillators. (You can assign it another name if you're using a variant on the 74C14, such as "40106" or "4584"—or, for that matter, "Schmitty"—it's just a label.) (See Figure 32.11.)

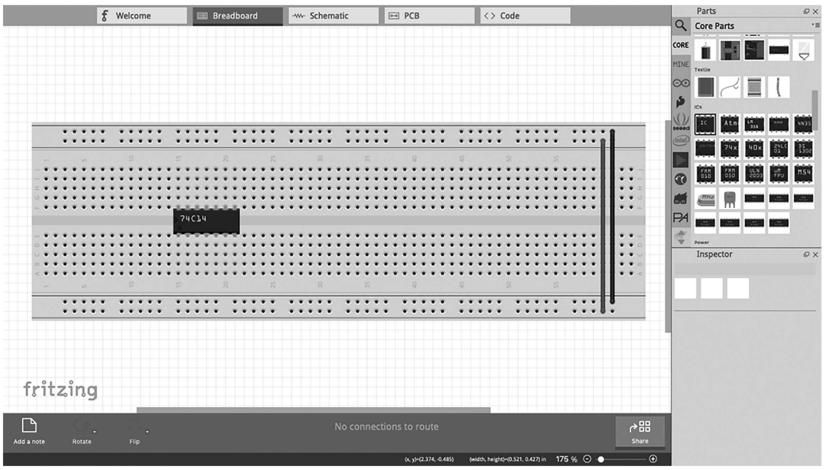


Figure 32.10 IC in breadboard.

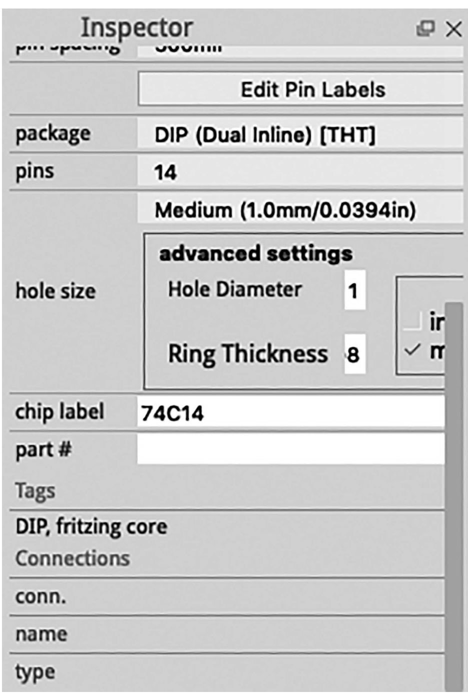


Figure 32.11 Labeling the IC.

Before we go any further, use the View Switcher on top of your screen to look at the Schematic and PCB. All of a sudden those are no longer empty views, but the IC that you placed on the breadboard now also appears as a symbol and as an outline (Figure 32.12). In Fritzing, you can work all modes of your design simultaneously. If you drop a part on the PCB view, it will show on the schematic and the breadboard and so on. But if you connect something on the schematic, for example, it will show as a dashed line on the Breadboard and PCB views (Figure 32.13). That dashed line is

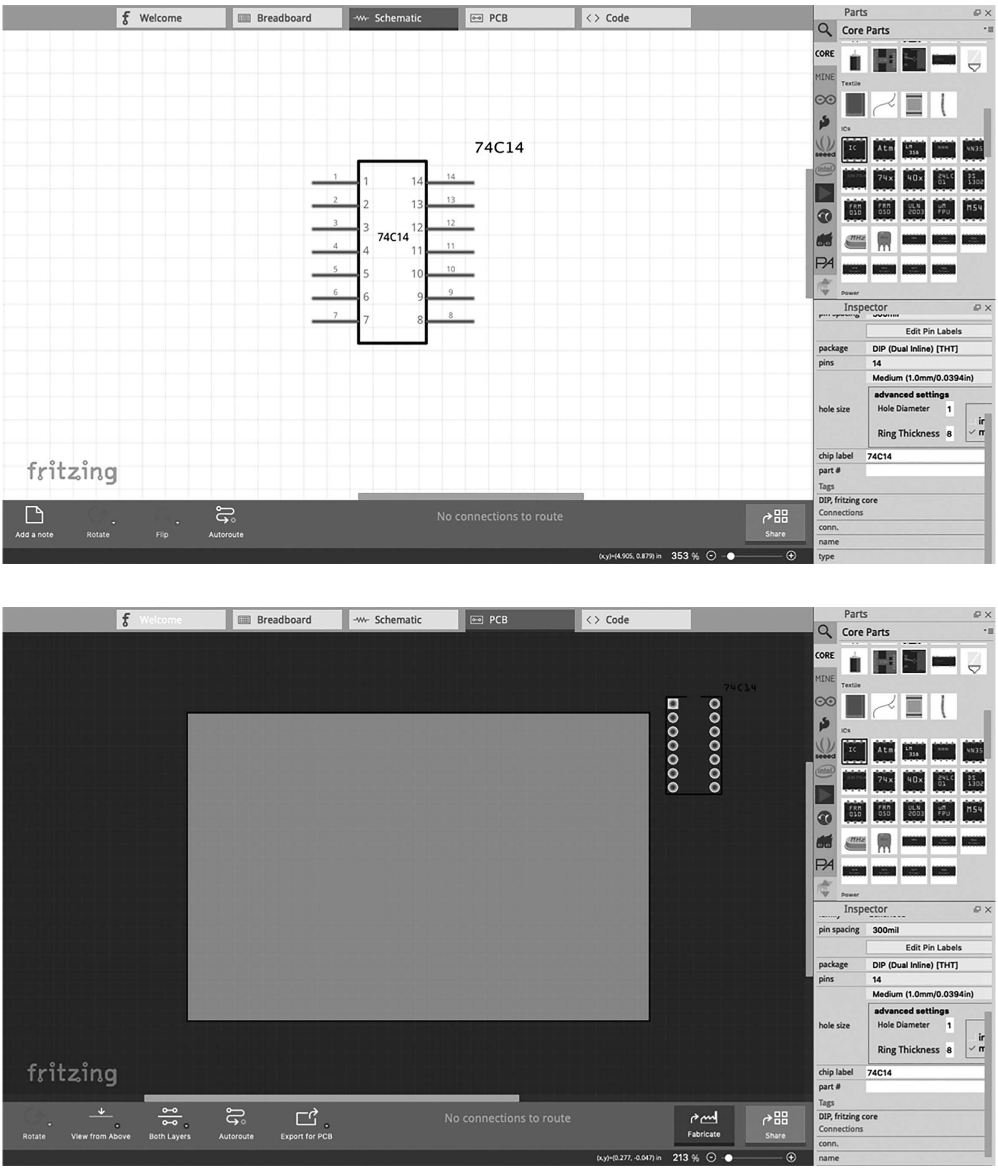


Figure 32.12 Schematic and PCB views.

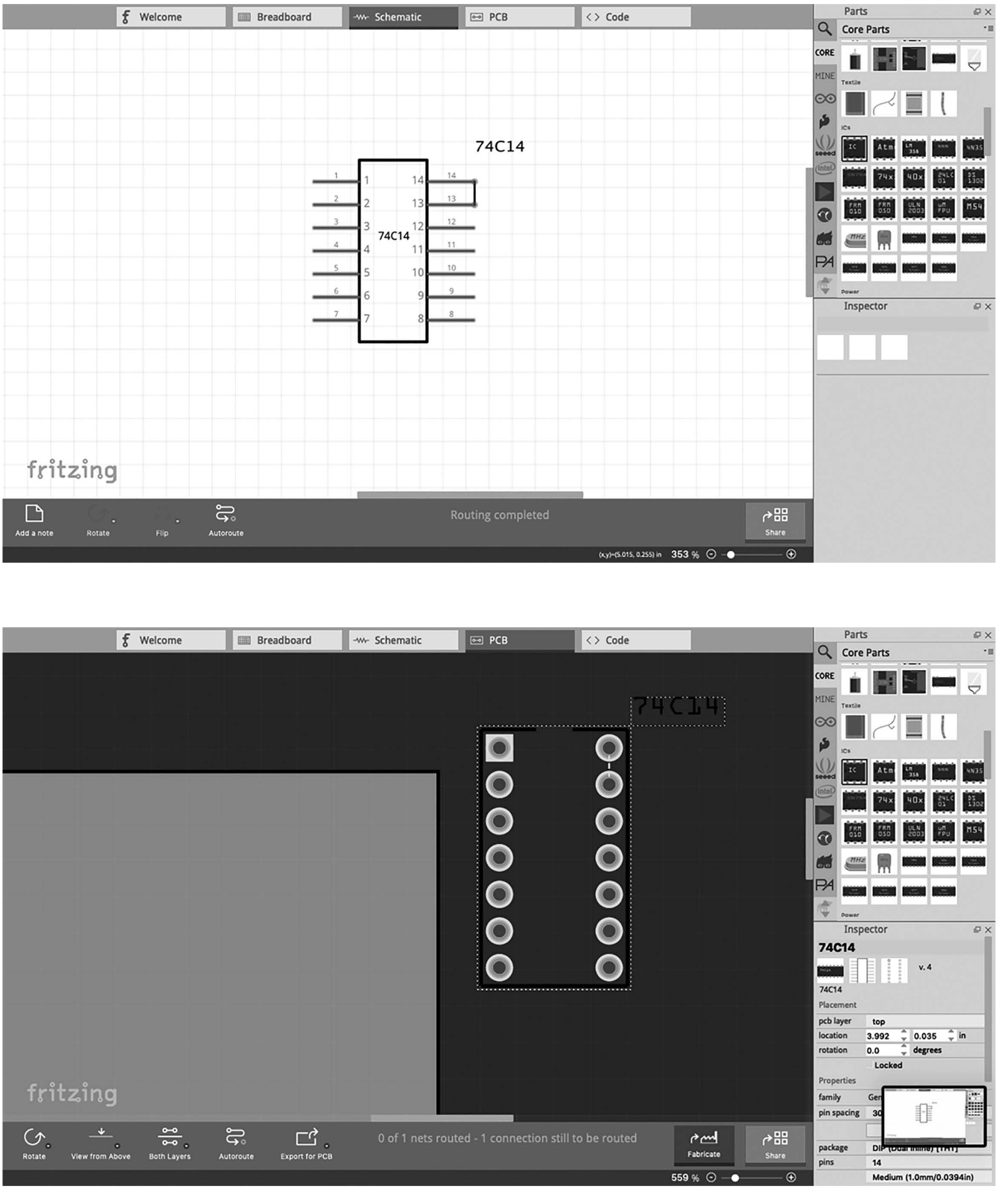


Figure 32.13 Schematic and PCB views with connections.

there to remind you that a connection has been made on a different view but the paths need to be made for the one you are looking at the moment. Once these are made, the dashed line disappears. This is a particularly useful feature of Fritzing.

Go back to the Breadboard view and add the remaining components for our oscillator. On the Parts magnifying glass, type "9v" and grab either the battery with the clip or the one with the SparkFun battery cradle (Figure 32.14). As you can see, it is an actual image of a battery. Connect it to the power rails by placing wires as we did before. Power up the chip: drag a wire from V+ to pin 14 and one from ground/V− to pin 7 (Figure 32.15).

From the Core Parts menu, pick an electrolytic capacitor and place it somewhere near the chip (Figure 32.16). This capacitor sets the range for your first oscillator. Right click the capacitor, on the Rotate menu, and select "Rotate 90° Counter Clockwise." Drag it and place the capacitor's cathode (the negative side with the stripe) on the ground rail. Now, click the anode (the other leg) and drag it to pin 1 of the chip (Figure 32.17). You can bend the legs of many components exactly as you would on your physical breadboard (this does not work the same for every part, though). For the oscillator's feedback resistor, search for a photoresistor on the Parts menu and put it on the breadboard next to the chip. Place and drag wires to connect it between pins 1 and 2 of the chip (Figure 32.18).

Figure 32.14
9v battery in Parts.

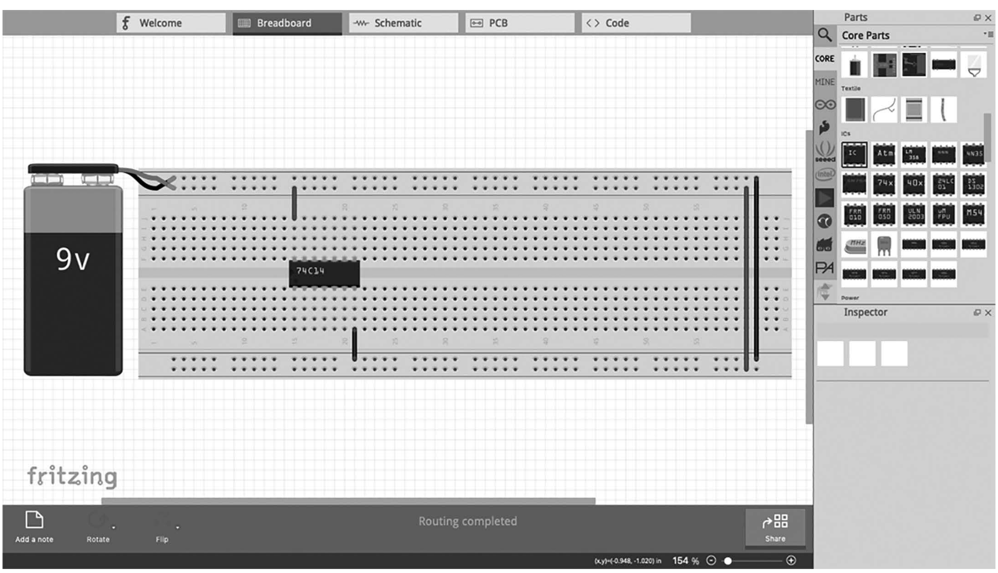


Figure 32.15 Connecting battery to breadboard.

Figure 32.16
Capacitor in Parts.

Now we need an audio jack for the output. Select whichever style suits your needs. I will use one of the 1/8″ (3.5 mm) jacks already mounted on a PCB and place it on the breadboard—you can rotate it to make it fit better. If you place and hold your cursor on top of the PCB's pins, a pop-up message will tell you what it is: the tip, ring, or sleeve (this is a feature that works for almost every part in Fritzing). Connect pin 2 of the IC to the tip and the sleeve to ground (Figure 32.19).

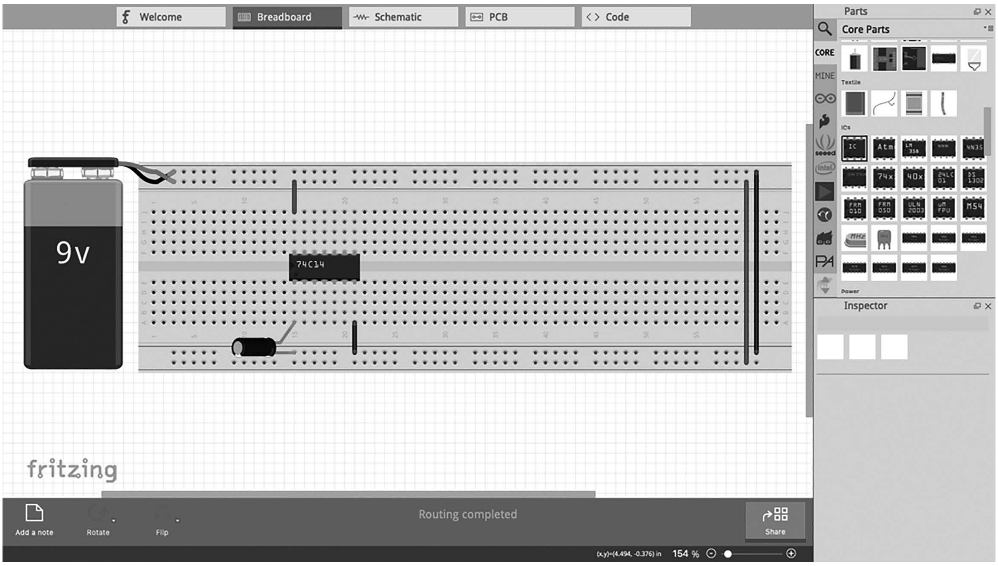


Figure 32.17 Connecting capacitor to breadboard.

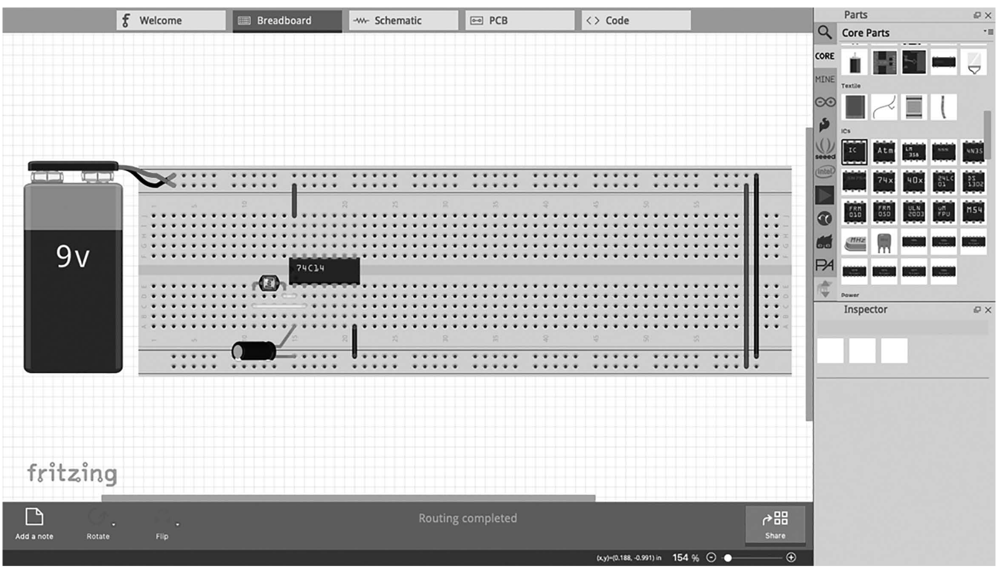


Figure 32.18 Photoresistor added to breadboard.

At this point you have successfully imported a prototype from the breadboard on your workbench to a breadboard on your computer screen. We still need to create the schematic and design the PCB. A caveat: schematics in Fritzing are not as concise as in other EDAs, but if you just want your board made, it will suffice.

In Fritzing you can design single- or double-sided boards (i.e., with copper traces on one or both sides). This can be changed in the Inspector. On the PCB view, the

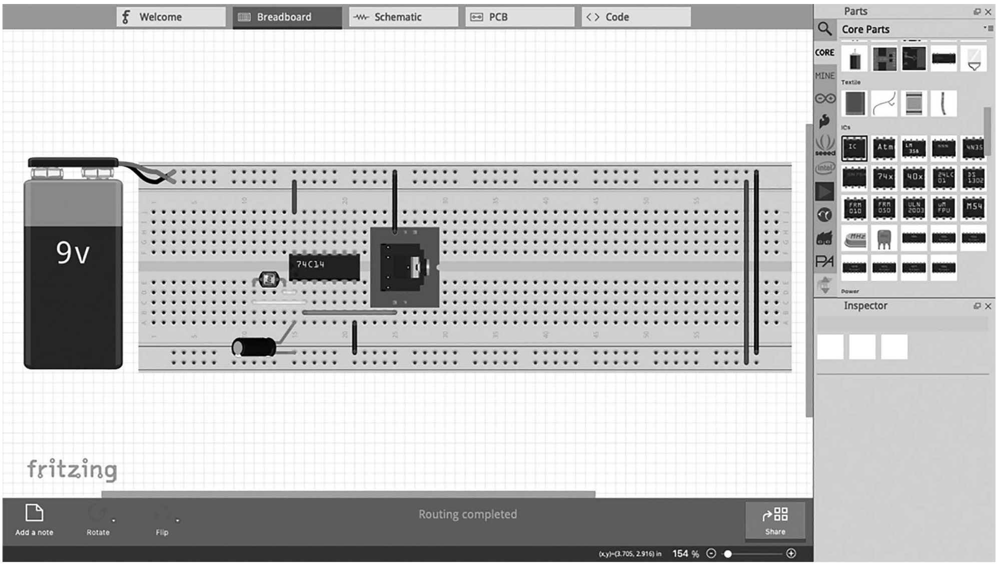


Figure 32.19 Output jack added to breadboard.

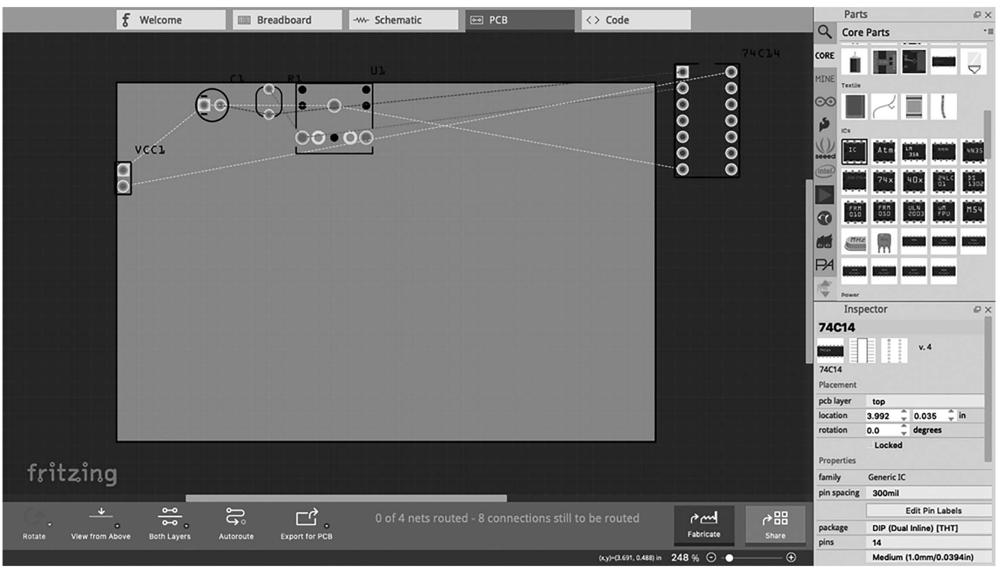


Figure 32.20 Initial state of PCB view, with components scattered.

parts will be interconnected correctly but scattered around the screen (Figure 32.20). You'll want to arrange the parts in an appealing way. There are a few factors that need to be considered, such as centering the IC or part with the most connections, not placing parts too close to one another, mounting audio jacks on the edges, avoiding overlapping traces, etc. But feel free to experiment. Doing things wrong can reveal hidden features of your design. Layout is a combination of puzzle solving and drawing a treasure map. While optimal functionality is usually a desired attribute, it should not be the exclusive goal of PCB design.

All those colored dashed lines display how things are virtually connected on the board and suggests potential routing for the traces. Although this rat's nest is a good reference, it is hardly the final word on how traces should be made.

To make traces and connect the parts on the board, click and drag like when laying out wires on the Breadboard view (Figure 32.21). If you double-click on a dashed line, a trace will be created. You can edit these traces. By clicking and dragging, you add bend points. If sharp angles are not your thing, press Command in OS X or Control in Windows to make curves (Figure 32.22). The default trace width is 24 mm, but you can make them thicker or thinner if you wish (I don't recommend making them much thinner). Our first design is simple, meant to illustrate the basic features of Fritzing, and therefore should not present any problems. But when you are short on time, or have trouble figuring out a specific routing, Autoroute might be an option. Using Autoroute does not mean that all the work will be done automatically, it simply entails a different workflow in which, instead of drawing all the traces yourself, you will be editing what the Autoroute did. Which, in truth, can end up being be a lot of editing.

As you work, acquaint yourself with the different layers that comprise a PCB. On the top menu bar, click the View tab. In the dropdown list, you will see all the different layers, plus the option to hide or show all layers. To familiarize yourself with these options, go ahead and choose Hide All Layers. The workspace will be empty, showing only the background grid. Now select them one at a time: Board, Silkscreen, Copper, etc. Some of them might not show anything right now, but as you work on your design, each of these becomes an essential part of your board.

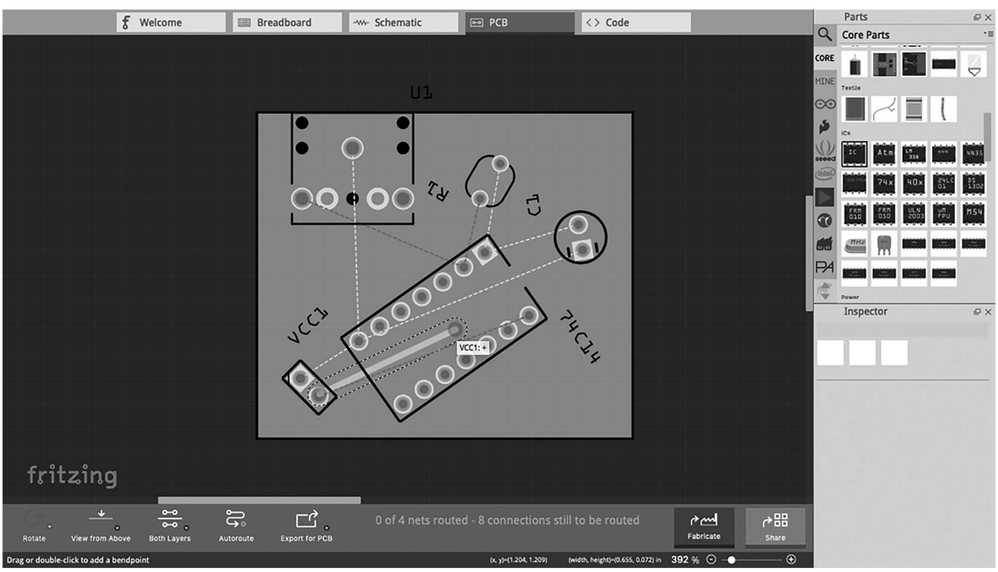


Figure 32.21 Adding traces to PCB.

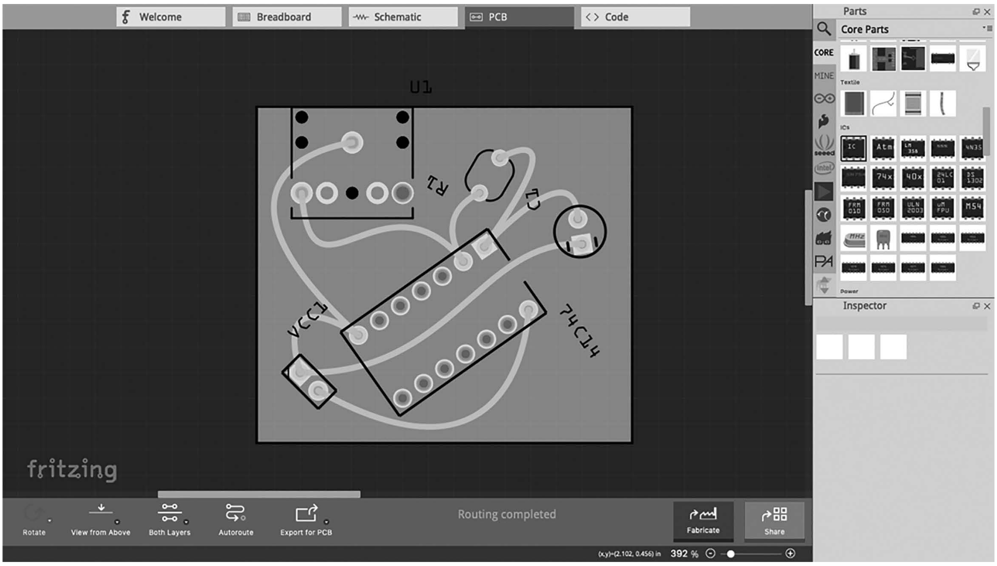


Figure 32.22 Curved traces on PCB.

Figure 32.23 Design Rules Check menu.

Once you are done making traces, make sure that everything's connected. You can look at the bottom bar and it should say "Routing completed". Or go to the menu bar and on the Routing tab select "Show unrouted." This way you can confirm that everything is really connected. If the pop-up window says "There are no unrouted connections," you are good.

After all traces have been made, perform a Design Rules Check (DRC). You will find this also under the Routing tab on the menu bar (Figure 32.23). If everything's all right, a prompt message saying "Your sketch is ready for production" will appear (Figure 32.24).

You can also use third-party custom settings for manufacture. First, select Autoroute/DRC Settings. Three options will be made available: "home-brew,"

Figure 32.24 Successful Design Rules Check.

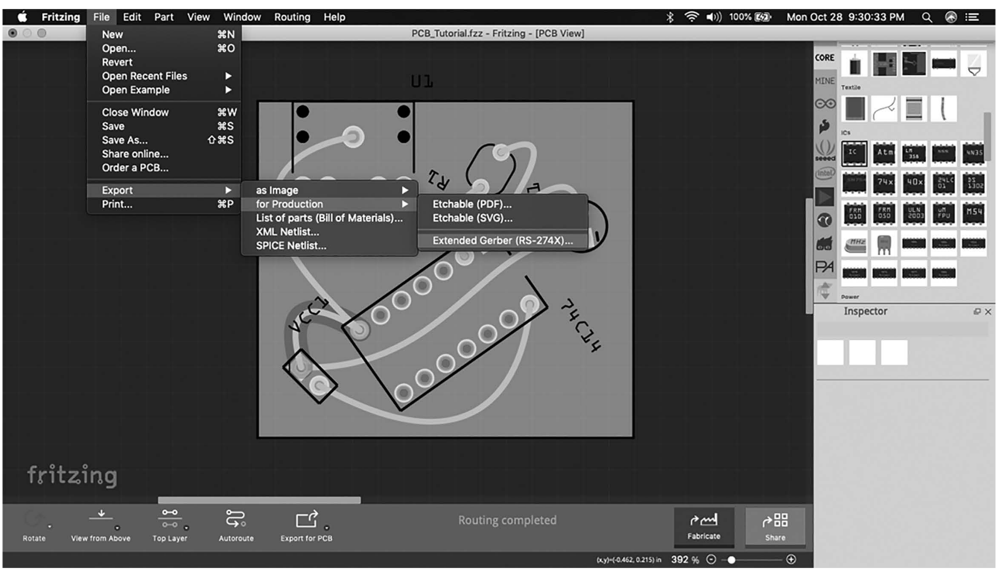


Figure 32.25 Generating and exporting a Gerber file.

"professional," and "custom." OSH Park—a popular board house for hackers—has specific Design Rules on how to prepare your files for manufacture. Go to docs.oshpark.com/design-tools/fritzing/ where you will find the Autorouter/DRC Settings that conform to OSH Park's manufacturing specifications. An important thing to notice is that they do not process .fzz files directly; instead they require a zip file with the Gerber layers and drill file. To generate the Gerber files, go to File > Export > for Production > Extended Gerbers (RS-274X) and save it to a new folder (Figure 32.25). Additionally, on the Export menu, you can also export other things such as the bill of materials for your board. Some other things to check are copper pour and board outline. It is good practice to check the board house's specs on their websites or to email them to make sure your design will be produced as desired. Many PC board production runs are held up not because the design is flawed but because of formatting issues in your files.

Overall, Fritzing is a good tool for beginners, teachers, or hobbyists, with its strong feature being the Breadboard view. But for some other aspects of designing PCBs, other EDAs are preferable. See "Sharing Traces: Designing and Fabricating Your Own Printed Circuit Boards With Eagle" on the website.

MAKING BOARDS

It's good practice to print your board layout on paper once it is done, glue it on a foam block, and do a mock-up version of the circuit to make sure that the parts are correct and that the spacings are adequate (thank you, Robert Drinkwater!). It is no fun to find yourself fixing every hole on the board with a pin vise because they were too small or learning that the legs of your switch do not match the part that was used as a template for the design. If everything fits on paper, it's time to fabricate a prototype.

There are three option when it comes to making your own boards: etching, milling, and manufacture. Etching will not be covered on this chapter—despite its old-school cred (think Robert Moog's 1960s Theremin kits)—because improperly disposed ferric oxide will not only ruin your sink and plumbing but is also a serious environmental hazard and is illegal to throw out. The other two approaches are safer.

Milling PCBs

Milling boards is a fast and reliable way of prototyping your designs before manufacturing them by the hundreds through a commercial house. It can be sufficient to produce a handful of boards for yourself and your friends. For this example, I will focus on Bantam Tools Desktop PCB Milling Machine, a CNC machine. Although at the time of writing still a little pricey for the amateur hacker, they are popping up in art schools and makerspaces. Ask around, but if, for any reason, milling is not an option, jump ahead to the next section on board houses.

The first thing to do is download Bantam's milling machine software to communicate with the mill and set up your project. You can find it at bantamtools.com/software-download. Install the application and launch it. The software is simple and fairly intuitive. In the center of your computer screen, you should see the bed of the machine and a blue block, which is your generic material (Figure 32.26). Zooming out enough will reveal an opaque image of the mill itself. Above, different views are available: front, top, and 3D, as well as preview and toolpath options. On the right are the configuration settings for your project. Connect the mill to your computer. Locate the bed.

Printed circuit boards are made of different materials. FR-4, made of fiberglass and epoxy resin, is probably the most common, but because of the fiberglass, machining FR-4 is dangerous for your health. Instead we are going to use FR-1, made with phenolic resin, which is safer (but nonetheless avoid ingesting, inhaling, or getting it in your eyes). FR-1 can be acquired from many sources, including Bantam, but also SparkFun and Digi-Key. On Bantam's Material dropdown menu, select Single-Sided FR-1. On the Material dropdown menu, select Single-Sided FR-1. Immediately the generic blue block turns into a copper board. If the board you have has different measurements, adjust the width and the height on the Size menu. I recommend leaving the default value thickness. Using thin double-sided tape, place your board on the mill's bed and against the guides. The tape is just so that the board is stable while the machines works. Do not use anything thicker than Scotch tape.

Import your design from your EDA into Bantam by clicking on the open button on the Plans section. Bantam supports many different file formats, but we will use ".brd." The design layout will appear on the board and can be moved within the

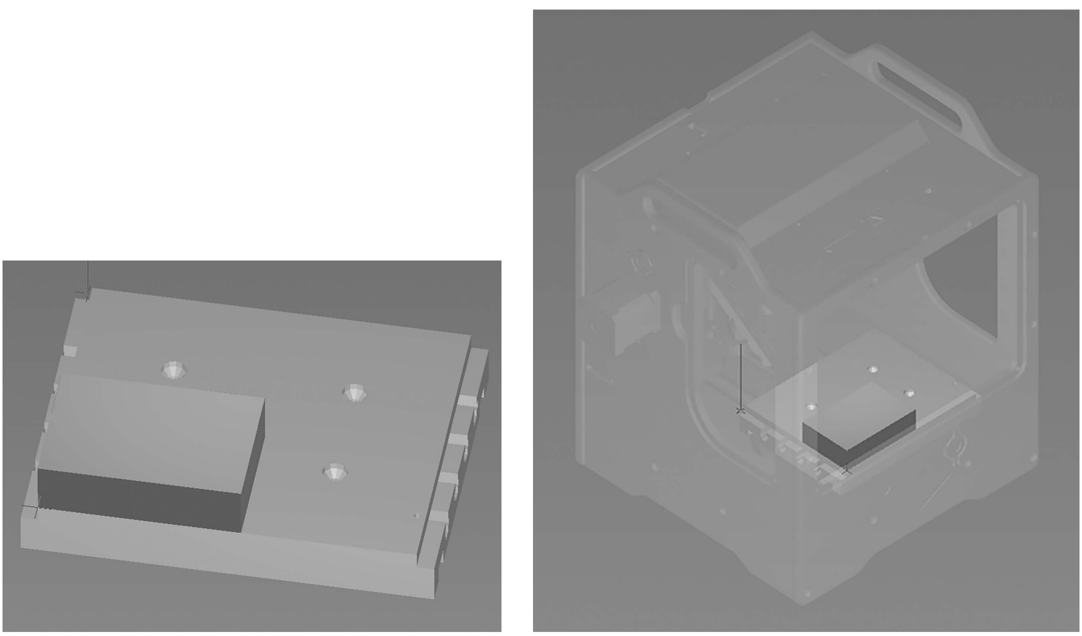


Figure 32.26 Mill software image.

boundaries. Use the placement menu to indicate the location and orientation. Leave small distance between the design and the guides for the safety of the drill bit.

In Milling Tools select the sizes of the bits you will be using. If two traces are too close for the bit diameter, red marks will show on the design. You can try different milling tools, or the traces can be adjusted in Eagle and refreshed on Bantam so that the changes are reflected. Whenever possible use just one tool for the whole job—it's easier. Below the Start Milling button is an estimate of how much time will it take to mill the board. Under Advanced, both the depth and clearance of the traces can be adjusted. Remember that milling too deep may result in a fragile board and the bit will become dull faster. Too much clearance will increase the turnaround time considerably and will make the bit dull faster.

Once your setup is ready, click Mill All Visible. The machine will ask you to insert the new tool. After the tool is in place, the mill "locates" the tool by touching the bed. Then milling starts.

When the board is ready, carefully—you don't want to snap it—lift it off the bed. Before soldering the components in place, take the time to check and clean the board. Make sure that there is no copper residue shorting the circuit.

Assemble the board carefully. FR-1 boards will allow for maybe one soldering mistake before the copper burns or peels off.

Manufacturing

Getting a board design manufactured requires that the files submitted comply with the board house's specifications. There are hundreds of manufacturers, and although the process is fairly standardized, requirements often differ from one to another. Failing to adhere to the appropriate spec may result in mistakes or the board not being produced at all. You need to generate the necessary manufacturing files, called Gerber (.gbr) files, which include copper, soldermask, and silkscreen, among other layers; and Excellon (.xln) for your drill files.

In Fritzing you can export the necessary files from File > Export > for Production > Extended Gerber (RS-264X). There is no CAM Processor for Fritzing. If you want to use a CAM template or do a custom job, you have to use an additional piece of software specifically for that. This is, in itself, knowledge worth acquiring, and I encourage you to look further into that. Nonetheless, if you go to File > Order a PCB, same as with the Fabricate option at the bottom right, you'll be directed to Aisler's website, where it is very easy to upload your board as a. zip file and view each one of your board's layers. Either using Aisler or some other board house, make sure that your DRC settings match the manufacturer's specifications (as I explained prior). A good way to explore is to take a look at the example files in Fritzing. You will find these on File > Open Example. Select one that might be appealing to you and study how they are made.

You might work with various board houses, so knowing how to produce the necessary files is important—you should import their Design Rule files to check your design. There is no need to produce 100 boards of a design that has not been tested.

An order can be placed with OSH Park—for example—for as little as three boards. This way economic risk—and e-waste—is reduced significantly when a project is still in its development phase.

EXPANDING NETWORKS

Now that you have a basic guide to making your own circuit boards, go ahead and practice these newly acquired skills on some of the other projects in this book. Make a nice matrix mixer, a sequencer, or a ring mod. I also encourage you to check Altium Circuitmaker, KiCad, or the cloud-based Upverter, among others. Get in touch with other designers of DIY electronics and trade, mix, modify, and learn. For some examples of how creative artists can get with designing their own boards, see my essay "Circuit Board as Design" on the website.

CHAPTER 33

Microcontroller Sound

JOSEPH KRAMER

You will need:

- An Arduino Uno.
- A computer running Windows, macOS, or Linux.
- A USB cable that can connect from your computer to the Uno's USB type-B connector.
- A breadboard.
- Some solid hookup wire or premade wire jumpers.
- Assorted resistors and potentiometers (1 kOhm–100 kOhm is a good range).
- A pushbutton or toggle switch.
- Some photoresistors.
- Some LEDs.
- An audio amplifier.
- Assorted jacks and plugs, to match your amplifier.
- Hand tools.

In this chapter we'll take a big step and configure electronic sound using *software*, rather than by connecting different electronic components together on a breadboard. The sounds are surprisingly similar, but the working environment is quite different. We'll start the tutorial here in the printed book; the website contains several additional projects, increasing in difficulty but also sonic reward. The website also includes code examples for all the projects, so if you don't fancy typing in all those commands by hand, you can copy and paste from your browser. If you haven't visited www.HandmadeElectronicMusic.com yet, now's the time.

Microcontrollers are essentially integrated circuits that are user configurable. These small ICs look much like the chips used elsewhere in this book but are unique in that they can be programed and reprogramed many times over to respond to inputs like knob turns, button presses, complex digital sensors, or even tweets and then generate outputs to blink and fade lights, create sound, control motors, drive displays, and more. Microcontrollers can also communicate with MIDI synthesizers or software running on personal computers. They are tiny computers in their own right but are distinct from laptops, desktop towers, and even Raspberry Pi–type computers (see Chapter 34) in that they do not include a full operating system to manage multiple applications. Instead, they run only one program at a time, written on an external computer and uploaded into the chip, where it stays until replaced by new code.

Arduino and its ilk are more than just the chips themselves but comprise an open-source computing *platform*. This includes a software programming application as well as a family of development boards that are designed to make building and programming microcontroller systems surprisingly easy for beginners while still offering significant power for the experienced user. The two core components of the Arduino platform are the software, which is an integrated development environment (called the Arduino IDE) that is used to program all the compatible boards, and the ecosystem of the hardware boards themselves, with all their associated shields[1] and other accessories.

There are many varieties of boards available from the Arduino company as well as from third-party developers. The one we will use for the examples in this chapter is the Arduino/Genuino *Uno*.[2] (Arduino is the name in the US. Genuino is the name outside the US.) This board is affordable (about US$22), rugged, and extensively supported on the internet with example circuits, code, and other learning resources. It is also extremely powerful, considering how beginner friendly it is. You can buy the hardware from a large number of retailers—refer to Appendix A for suggestions.

To get started using the Uno, direct the browser on your personal computer to Arduino.cc. There you can download the latest version of the IDE that works on your operating system. Windows, macOS, and Linux are supported. Bookmark the Arduino site, where you will find all sorts of resources for getting started, troubleshooting, learning new skills, etc. I encourage the reader to go back to the Arduino website regularly while reading this chapter since many questions that may arise are likely addressed there.

Once you have downloaded the software, launch it and you will see a window for a new file, called a "sketch" in Arduino parlance. Now you are ready to plug in your Uno and make some magic.

HELLO WORLD OR TESTING YOUR TOOLCHAIN

We'll start by confirming your Arduino board is working and communicating with your computer. Connect a cable between a USB port on your computer and the USB type-B plug on your Uno; this cable powers your Uno from the 5-volt supply of your computer and serves as a serial communication link between the two machines.

1. Open the blink example sketch from the Arduino IDE menu bar at **File > Examples > 01.Basics > Blink**.
2. Select your board, "Arduino/Genuino Uno," from **Tools > Board**.
3. Select the correct port from **Tools > Port**. This selects which of your serial ports is assigned to communicate with your Uno. The port names may be confusing. You can ignore almost everything. Just look for one that says "Arduino" somewhere in the name, e.g., "/dev/cu.usbmodem14202 (Arduino/Genuino Uno)." If you have trouble finding the correct port, see troubleshooting tips on arduino.cc.

4. Upload the Blink sketch to the Uno by clicking the **Upload** button (right arrow near the top left corner of sketch window) or in the menu at **Sketch > Upload**. The program should start up automatically after upload is finished.
5. Observe the built-in LED, attached through a resistor to pin 13 on the Uno. It should light up and then stay lit for one second, then turn off and stay off for one second, then light up again, and so on, until you disconnect the Uno from your computer, which removes its source of power.
6. Change the blink speed by altering the code. The number in the parenthesis in the instruction "delay(1000);" represents the amount of time in milliseconds that the program pauses before the next instruction. Change the number from 1000 to 100 in the parentheses after *both* delays. The two lines of code that previously read "delay(1000);" should now read "delay(100);."
7. Re-upload by clicking the **Upload** button again.
8. Observe the built-in LED. It should be blinking much faster now.

You have just programed a microcontroller!

If your Arduino ever behaves strangely, or you find yourself at a troubleshooting roadblock, it can help to run through this simple process again to confirm communication and basic functionality of your setup.

ELEMENTS OF A SKETCH

An Arduino sketch is just a list of instructions written in plain text. The text makes use of special code words, punctuation, capitalization, symbols, and formatting to tell a microcontroller what to do and how to do it. Sketches could be written in any text editor, but one of the things that makes the Arduino IDE convenient is that it helps to identify the code words that have a special meaning in the language, and it provides assistance with formatting. The following are a few concepts to help orient readers to what they are seeing when looking at a sketch:

- Comments. Often the first text in a sketch, and found in abundance throughout, comments provide information and clarification for human readers. Comments do not get complied or sent to the microcontroller. They appear gray in the IDE window. There are two ways to create a comment:
  - For a block of comments (multiple lines), type a slash followed by an asterisk: /*. Everything that follows, regardless of how many lines, is considered part of the comment block. To indicate the end of the comment block, type the symbols again in reverse order: */. Everything between /* . . . and . . . */ will be considered a comment.
  - For comments on a single line, type two slashes //. All text *to the right of the two slashes on the same line* is considered a comment.
- Functions. Functions are keywords that tell the microcontroller to *do something*. They can set internal modes, toggle or read pins, do math, or do some combination of these things and more. Arduino's built-in functions will turn orange in the

IDE. Function names are always followed by parenthesis. Some functions allow you to set parameters (called "arguments") inside the parenthesis. An example of a function in the Blink sketch is delay(). The delay() function causes the microcontroller to pause for a certain amount of time before going on to the next instruction. The length of the pause (in milliseconds) is specified by the user by typing a number into the parenthesis. For information on any built-in function, simply highlight the function name and click **Help > Find in Reference**.
- void setup() and void loop()—These are special functions in Arduino. Every sketch must have one setup() function and one loop() function. Neither may be missing, nor may either appear more than once.
  - setup()—Runs once each time the microcontroller is powered on. Inside the block of this function, after the parenthesis between the opening and closing curly braces, can be any number of other functions that will each run in order once.
  - loop()—Runs after the setup() function. Each instruction is executed in order until the end of the loop. Then it starts over at the beginning. The loop repeats continuously as long as there is power. Every sketch must have one and only one loop function. Inside this function, between the opening and closing curly braces, any number of other functions can be called.

A DIGITAL OSCILLATOR

Now that we have a better idea of what we are looking at, we'll start by modifying the basic blink sketch to make noise. This project requires some additional circuitry to be connected to the Uno.

Most Arduino boards have connectors called "headers" used for making connections to additional components. Each hole on the header is labeled with a shorthand reference to its function on the microcontroller. The most important connections you will use on the Uno are:

- 14 *digital* pins labeled 0–13, used for digital input or output, and referenced in the code by their numbers. They read *or* write either 0 volts or 5 volts, but no values in between.
- 6 *analog* input pins, labeled A0–A5, referenced in the code by those labels. They read continuous changes in voltage between 0 volts and 5 volts.
- The *power* pins:
  - "5V" provides 5 volts for things like sensors or additional circuits.
  - "GND" provides a ground reference, important in many input and output connections. There are three completely interchangeable pins for ground.
  - "Vin" is for connection from a non-USB power supply between 7–12 volts. The Uno can run your program from a 9-volt battery or a USB battery pack.

You can link components such as potentiometers or jacks to the Uno by soldering one end of a piece of solid wire to a connection on the pot or jack and then

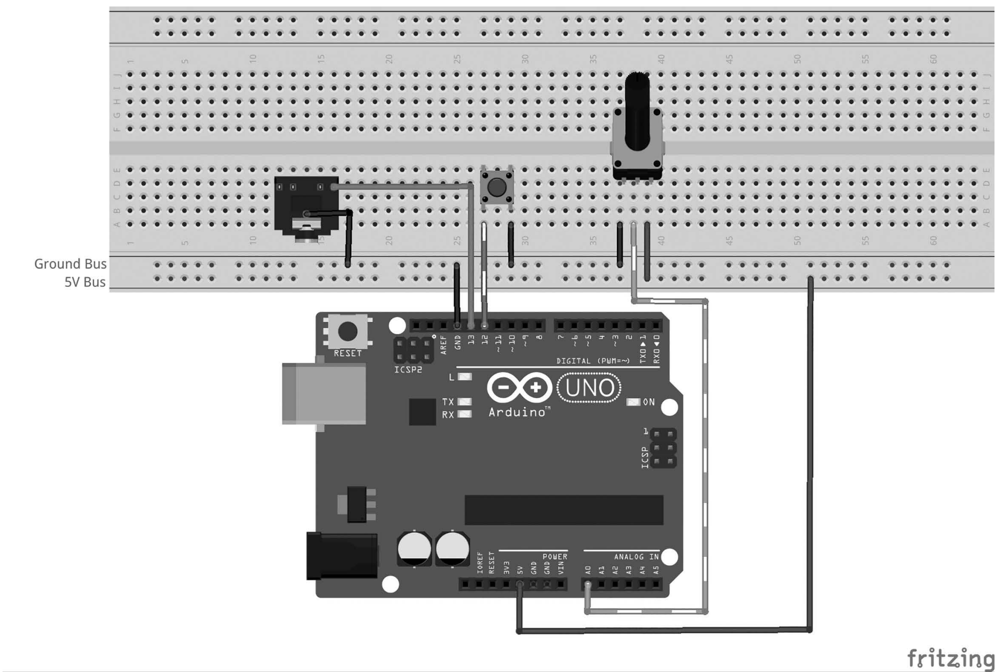


Figure 33.1 Fritzing image of digital oscillator circuit on breadboard.

inserting the other end of the wire (with insulation stripped back 1 cm or so) into the appropriate point on a header. (You can buy ready-made jumpers for interconnections as well.)

Connect the components shown in Figure 33.1 and load the first sketch for this project:

1. UNPLUG the USB cord from your Uno so it is not powered while you make your connections.
2. Hook up an audio jack to pin 13, a button to pin 12, and a potentiometer to pin A0.
   - The signal/tip connection of the jack is connected to pin 13 and the ground is connected to a ground bus of the breadboard. (Remember Rule #10!)
   - The button is connected to pin 12 on one side and the ground bus on the other.
   - The potentiometer is connected to A0 from its center tab, the ground bus on the left, and the +5v bus on the right.
   - The ground bus on the breadboard is connected to the "GND" pin on the Uno.
   - The +5v bus on the breadboard is connected to "5v" on the Uno.
3. Open a new sketch in the IDE from the menu with **File > New**.
4. Delete everything out of the sketch so you start with an empty window (select all, delete).

5. Type into the window exactly what you see in the code following. (You can also copy/paste the code from the companion website.)
6. RECONNECT the USB cable and double-check that you have the correct port selected in **Tools > Port**.
7. Upload the code by clicking the **Upload** button near the top left of the sketch window.
8. Connect the output to your amplifier and listen to the results.

```
//CODE HEM_DigitalOSC ******
int ledPin = 13;     //variable to represent LED Pin
int periodKnob = A0; //variable for knob pin (A0 = analog
                       in pin 0)
int delayTime; //variable for the delay time

void setup() {
 pinMode(ledPin, OUTPUT); //configure pin 13 as a digital
    output
}

void loop() {
 //set delay time equal to the current value read on
    analog pin 0
 delayTime = analogRead(periodKnob);

 //map the analog read range from 0-1023 to 10000-1
 //delayTime = map(delayTime, 0, 1023, 10000, 1);
 digitalWrite(ledPin, HIGH);     //set pin 13 to 5 volts
 delayMicroseconds(delayTime);   //pause program
 digitalWrite(ledPin, LOW);      //set pin 13 to 0 volts
 delayMicroseconds(delayTime);   //pause program
}
//**********END OF CODE
```

TROUBLESHOOTING ERROR MESSAGES

If your code uploads correctly, you should see that encouraging message "done uploading" near the bottom of your Arduino IDE window. What you should *hear* is a square wave that changes pitch as you turn the attached knob. No joy? It is very common to get error messages when you upload code. While I recommend going to the Arduino website and checking out the troubleshooting resources there, I want to offer a few quick tips here:

- The Arduino IDE highlights the line of code *near* the error. Often the error is in the line preceding the highlighted one.
- Code errors are often just typos or missing punctuation. Arduino is case-sensitive. Check for capitalization. Check colons, semicolons, commas, and periods. Lines of instruction end with semicolons. Make sure every *opening* curly brace, bracket, or parenthesis has a corresponding *closing* curly brace, bracket, or parenthesis.

- Be patient, errors happen all the time. Just solve the problems one at a time until your code uploads (don't worry, the irritation you experience reading this sentence will decline the longer you program).

If all seems lost, try going through the steps of Testing Your Toolchain (prior). You won't need to change anything about your hardware setup to test the Blink sketch (which is an example you know is error free). This will verify that the Uno is communicating with your computer and let you feel the satisfaction of blinking your light. Then come back to troubleshoot your own code.

WHAT'S HAPPENING?

The sketch provides a list of instructions to the microcontroller detailing what it should do internally and how it should interact with the connected hardware. The first line of our sketch is a comment to let the reader know the name of the code. The rest of the sketch is a combination of variable declarations, functions, and explanatory comments.

VARIABLE DECLARATIONS

The first three lines of code after the initial comment each declare *variables*. Variables are code words chosen by the user to represent specific values within the code. The data they represent get stored in memory so they can be used throughout your program. When creating, or *declaring*, a variable, you must start by indicting what type of data you want the variable to store. This is necessary because the program needs to know how much memory to allocate (among other things). In this example we are using the data type *int*. This is short for integer (whole numbers between −32,768 and 32,767), and it is a good default data type for beginners (other examples include *float*, which is used for storing numbers with decimals, or *long* for storing very large numbers).

This sketch uses variables for two purposes: to indicate which pins external hardware is connected to and to store a number that we will use to change the delay time while the program is running. The first variable declaration happens in this line of code: "int ledPin = 13;." This statement tells the program that every time it sees the word "ledPin" (capitalization is important), it should replace it with the number 13. A similar thing is happening with the declaration of periodKnob: every time we see that variable in the code, the microcontroller will replace it with "A0" (to the Uno this means the first pin on the analog input header). If we wanted to hook the potentiometer to the *last* analog input on the header, instead of the first, we would simply change the number in this variable declaration to A5. This would make all other references to periodKnob in the code reflect that change. What a time saver!

The third variable declaration is a little bit different from the first two. It is still declared as an int, but the variable delayTime is not set equal to anything. We do this because the value of delayTime will be set and changed by functions in the code as it is running. We will learn more about this when we discuss the function that uses this variable.

DIGITAL CONNECTIONS

This circuit uses 2 of the 14 digital pins available on the Arduino Uno, although this first example sketch only utilizes 1 of these pins for now (the second connection will be programed to read a button as we build on this code in the next example). Recall that each digital pin can be configured as either an input or an output, and all speak the two-word language of 0 or 5 volts. Pin 13 is configured as an *output* in the setup() function. This pin will now function similarly to the outputs of the CMOS oscillators seen elsewhere in this book. The mode is set using the following line of instruction: "pinMode(ledPin, OUTPUT);". This instruction calls the function pinMode(), which takes two arguments in its parenthesis: which pin and what mode. We use our variable ledPin to tell it that we want to set the mode of pin 13. The word OUTPUT, written in all capital letters, tells the IDE to set up the microcontroller pin for being set to a "HIGH" state of 5 volts or a "LOW" state of 0 volts (also known as ground). This can illuminate an LED (if you add a resistor in series—see Rule #22). This same pin can be hooked to an audio jack and used as our sound source. If the pin is not set up as an output in the setup() function, the LED will look very dim or not light at all. If you run into trouble with a digital output, double-check that its mode has been properly defined in the setup() function.

We toggle the pin's value inside of the loop() function. When a pin is set up as an output, we use the function digitalWrite() to tell the microcontroller to either set the output to 5 volts or to 0. The lines that do this are digitalWrite(ledPin, HIGH); and digitalWrite(ledPin, LOW);. A blinking light or rectangular audio waveform can be produced if we toggle the pin high, wait a short amount of time using the delay() function, then toggle the pin low, delay(), and repeat.

The Uno can only directly produce a rectangular wave from its output pins, similar to the waveform of our trusty CMOS chips throughout this book. There are no analog output pins, so no voltage in between 0 and 5 volts can be directly generated by the Uno.[3]

ANALOG INPUT

This example also features a potentiometer wired up to the Uno through the analog input header. These six pins are dedicated analog inputs. They cannot be configured as analog outputs, so it is not necessary to set the pin mode as we did for the digital pins.

The Uno can read the voltage on its analog input pins using the function analogRead(). An argument in the parenthesis tells the Uno which analog input pin to read. In our case, this is pin A0, represented by the variable we named "periodKnob." The analogRead() function *returns* an number in the range of 0–1023 (this is the range of the 10-bit analog-to-digital converter in the Uno). When a function is said to return something, it can be useful to imagine that the whole function is replaced by whatever data results from the operation of the function.

In a single line of instruction, "delayTime = analogRead(periodKnob);" we set our variable delayTime equal to the result of the function analogRead(periodKnob).

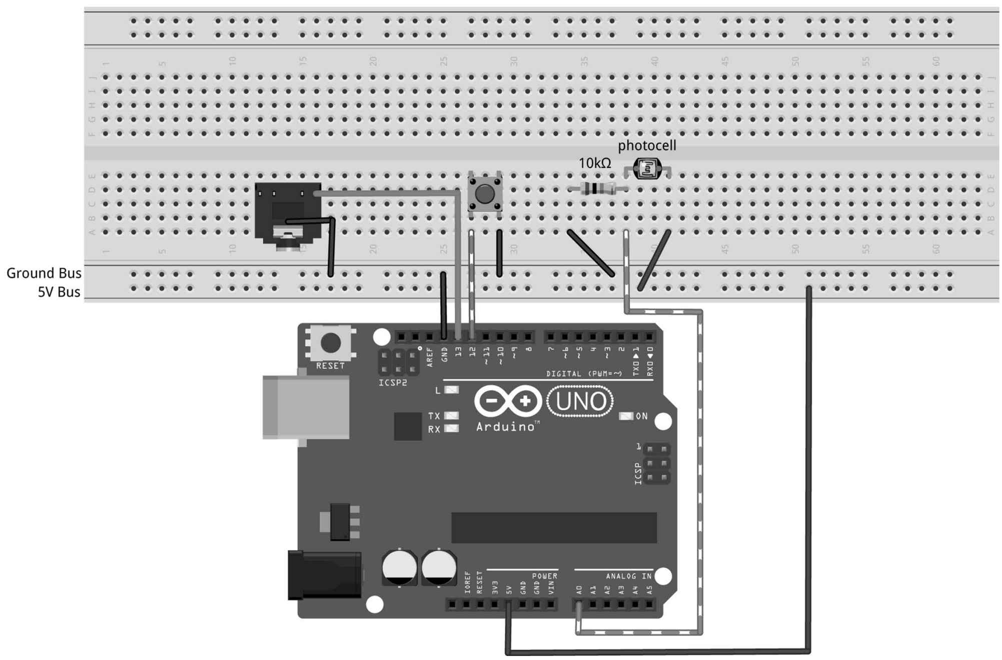


Figure 33.2 Using a photocell for light-controlled frequency.

The next line of instruction, "delayMicroseconds(delayTime);" will set the length of the pause relative to the position of the potentiometer. Since the length of the delays is directly responsible for the frequency of the oscillator, this is how the pot is able to control the frequency.

The potentiometer is set up as a voltage divider. Other sensors could be used here as well (Figure 33.2). See Chapter 13 for ideas of how to hook up photocells, pressure sensors, etc. as alternative voltage dividers.[4] Additionally, there are many novel, off-the-shelf sensors that have analog outputs that can be substituted into any sketch that uses analogRead().[5]

"COMMENTING OUT"

Curious readers may have wondered about this line: "//delayTime = map(delayTime, 0, 1023, 10000, 1);." It is clearly a comment because it has two slashes in front and appears gray in the IDE, but it doesn't look like it is there to clarify nearby lines of code for a human reader. It is, in fact, a line of instruction with its variable names, equals sign, and semicolon at the end. What is happening here is known as "commenting out" code. You can think of this as a way of temporarily muting lines of instruction. It is a useful technique for gaining an understanding of what each line of code does and for trying new ideas. This is an effective alternative to deleting and retyping lines of code when experimenting so you can quickly restore them if needed.

In this case, with the line commented out, you will notice that turning the knob changes the pitch, but the range is fairly small and the direction is the opposite of what might seem most intuitive. You can change the operating direction by swapping the +5v and ground wire connections from the potentiometer. But we are going to leave the wiring unchanged and both reverse the direction and at the same time scale the range of numbers the knob provides to the code. If you "uncomment" the line of code with the "map()" function by deleting the "//," the line should no longer appear gray in the IDE. Reupload the code and you will see that the knob now has increased range and is operating in the conventional direction. You can experiment with changing the last two values in the map function to experiment with different range and direction settings.

LET'S USE THAT BUTTON

Now it's time to use that button that we hooked up in the beginning of this project (note that even though the button has been connected all along, it has no effect until we include it in the programming). One side of the button is connected directly to ground, and the other side is attached to a digital pin. We will need to set the digital pin's mode in the setup() to allow the microcontroller to read the voltage on the pin. For this arrangement of hardware, we will use the INPUT_PULLUP mode. The pin will always read HIGH unless the button is pressed, thereby connecting the pin to ground and causing the button to read LOW (this may seem backwards, but it is a common and efficient way to make the connection).

To do something useful with the button, we will need to set up a *control structure* to check and see if the button is being pressed. Control structures can effectively create alternative paths and mini loops inside the main loop of a program. If the button is pressed (the pin reads LOW), we want to play a tone. In our sketch, we will use the "if/ else" control structure as illustrated in the pseudo code here:

```
if (digitalRead(buttonPin) == LOW) {
//play my tone here
}
else {
//do not play the tone
}
```

The "if statement" in this code fragment uses a *conditional statement* to decide whether or not to execute the instructions that follow in the block between its opening and closing curly braces. A conditional statement compares two things. In our case, the word "if" is followed by parenthesis enclosing a condition "(digitalRead(buttonPin) == LOW)" that must be met in order for the subsection of code nested between the pair of curly braces to be executed.

If the microcontroller reads the button pin and sees that it is equal to "LOW," it will execute the code between the braces. If the condition is not met, the microcontroller will skip the instructions in the if statement and simply move on to execute whatever code follows after the closing brace. Note that this equality is specified with *two* equals signs. Nearly all comparison operators use two symbols.

Always make sure that every open curly brace "{" has its corresponding closing curly brace "}" in the correct spot. Finding companion punctuation is made simple by a feature of the Arduino IDE: if you place your curser next to any parenthesis, curly brace, or bracket, the IDE will highlight its companion. If the corresponding brace is off screen, the IDE will create a small pop-up indicating the location of the matching brace.

Without changing the circuit on the breadboard, open a new window, clear the contents, and type the following code (or copy and paste from the website):

```
//CODE HEM_ButtonOSC ******
int ledPin = 13; //variable to represent LED Pin
int buttonPin = 12; //variable for the button pin
int periodKnob = A0; //variable for knob pin (A = analog in)
int delayTime; //variable for the delay time
//unsigned int means the integer doesn't go negative, so
   it can be twice as big
unsigned int delayMax; //variable for max delay time

void setup() {
 pinMode(ledPin, OUTPUT); //configure pin 13 as an output
 //INPUT_PULLUP sets the pin high. It gets pulled "low" by
 //connecting it to ground through a button
 pinMode(buttonPin, INPUT_PULLUP);
}

void loop() {
 //check the button. It will be low if pressed
 //If the button is pressed, execute the oscillator code:
 if (digitalRead(buttonPin) == LOW) {
  //set delay time equal to the current value read on
     analog pin 0
  delayTime = analogRead(periodKnob);

  //map the analog read range from 0-1023 to 10000 to 1
  delayTime = map(delayTime, 0, 1023, 10000, 1);
  digitalWrite(ledPin, HIGH);
  delayMicroseconds(delayTime); //shorter delay
  digitalWrite(ledPin, LOW);
  delayMicroseconds(delayTime);
 }

 else {
  //if the button is not pressed, execute the noise
     oscillator
  delayMax = analogRead(periodKnob); //set delayMax to
     current knob reading
```

```
    //map delayMax from 0-1023 range to 0-30000 range
    delayMax = map(delayMax, 0, 1023, 20, 30000);
    digitalWrite(ledPin, HIGH); //toggle pin 13 to +5V
    //Pause the sketch for a random number of
    //microseconds between 0 and delayMax
    delayMicroseconds(random(delayMax));
    digitalWrite(ledPin, LOW);
    delayMicroseconds(random(delayMax));
   }

}
//**********END OF CODE
Upload the code.
```

If the button is not pressed and the pin reads HIGH, the code does what is in the *else* block (this is the area between the open curly brace that follows the word "else" and the corresponding closing curly brace). This example code has a variation of the blink sketch in the *else* statement that produces a noisy random rectangular wave that is influenced by the knob whenever the button is not being pressed. Pressing the button will cause the pin to read LOW and engage the oscillator code from before. In an if/else structure, only one of the statements gets activated each time through the loop. To disable the noise and leave only the button-operated oscillator and silence, comment out or simply delete the else statement and its curly braces and reupload the code.

Swapping the pushbutton for a toggle switch will change it to a latching mode selector. Replacing it with a tilt switch could cause your oscillator to break into noise when it is not held level. In place of buttons and switches, it is also possible to use the outputs of other circuits as the inputs to microcontroller and vice versa. Just make sure that no voltages go over 5 volts and the grounds of all circuits are connected.

GOING FURTHER

This chapter introduced the Arduino platform through the less commonly discussed built-in sonic capabilities of the Arduino Uno. There are many resources available for learning more about Arduino. Most of these focus on connecting sensors and actuators, and interfacing with basic programming, but there is still significantly more *sound* experimentation that can be done with microcontrollers: live audio processing, arbitrary waveform generation, drum sound synthesis, and even sampling are possible. Go to the "Microcontroller Sound" section of the website. There we have uploaded sketches and accompanying instructions for a seven more sound projects.

NOTES

1. "Shields" are hardware expansions that plug into specific microcontroller boards to add functionality.

2. It is possible to use a different board for these code examples, but some of the details of the electronic hardware will need to be adjusted as pin numbers and voltage levels may be different.
3. Analog-like output effects can be achieved using Pulse Width Modulation (PWM), and true analog outputs are possible using additional circuitry.
4. Many voltage divider sensors using fixed resistors will not be able to provide voltages that cover the entire range from 0 volts to 5 volts. It can be useful to see the actual values that are making it into your code. To do this, you must open a line of communication between the microcontroller and the computer. For an example of how to do this, start with the built-in example in File > Examples > 01.Basics > AnalogReadSerial. Once you know the actual range being sent, you can use the map() function to quickly scale to your desired target range.
5. Examples of sensors that can work with analogRead() include joysticks; distance, temperature, and sound level sensors; air pressure sensors for breath control; etc. The key is to find sensors that provide an analog output voltage, otherwise your code may need to be modified significantly to use the device.

CHAPTER 34

Small Sound
Pure Data on Raspberry Pi[1]

ROBB DRINKWATER

You will need:

- A Raspberry Pi, with power supply.
- A computer running Windows, macOS, or Linux.
- A USB cable that can connect from your computer to the Pi's USB type-B connector.
- An HDMI computer monitor, with cable to connect to Pi.
- A computer keyboard and mouse, with cable to connect to micro-USB connector on the Pi.
- Some form of amplifier and speaker, with cables to connect to the 1/8-inch/3.5 mm stereo female jack on the Pi.
- Hand tools.

This chapter introduces the Raspberry Pi microcomputer and the software Pure Data, which combined create an inexpensive yet powerful computing platform for digital audio creation.

Although barely the size of a credit card, the Raspberry Pi is a full-blown computer with the capabilities of a desktop or laptop machine. Unlike the Arduino and other *microcontrollers* (see Chapter 33), the Pi can run a high-level operating system (Linux), and after hooking up a keyboard and monitor (a TV with HDMI will do), you have access to a full range of programming tools. But once your program is finished, the tiny Pi can be disconnected from its support hardware and embedded in a sculpture or musical instrument, bringing full computing power into almost anything.

PureData (also known as Pd) is a free, open-source software for creating audio and visual art. Originally written by Miller Puckett and now supported by a community of programmers, it is a tool that allows artists to create their own audio and visual applications through a visual programming interface, similar to Max/MSP (also originally written by Miller). For many people, this is a less intimidating way to start programming than the text-based language used with things like the Arduino.

GETTING SET UP

Hardware

You can purchase a Raspberry Pi from a wide range of online sources—see Appendix A. The Pi arrives as a classic DIY project: to get it working, you'll have to add a number of accessories that would typically come bundled with a "normal" personal computer: a power supply, a keyboard, a mouse, and a display. To support these peripherals, the Pi circuit board is equipped with several ports: USB ports for a mouse and keyboard, a micro-USB port for power, and an HDMI connector for attaching a screen. A keyboard, mouse, and monitor can be scrounged from a retired personal computer (if you don't already have a keyboard, look for one that has its own USB ports; this way you can attach the mouse without using up one of the USB ports on the Pi itself), and any HDMI-equipped TV can be used as a monitor. A small wall wart power supply can be purchased from the same source as the Pi—look for a 5-volt supply that provides a minimum of 1,500 mA (1.5 amps). (A nice touch in the design is that the "power connection" is the same as many phones and electronic devices, i.e., micro-USB, and so it's easy to get, and you might be able to use one you already have.)

Since we're focusing on sound in this book, you'll want to connect the Pi to speakers or headphones via the 1/8-inch/3.5 mm female jack.

You'll also need an ordinary personal computer to download and install the software required to get the Pi up and running—Macintosh or PC, it doesn't matter. This computer should have an SD card reader built in; if not you can use an external one.

Operating System

Before you can power up your Pi and get to work, you need to install an operating system, which every computer needs in order to function. There are a number of Raspberry Pi "starter kits" out there, some of which contain not only the power supply but also an SD card with the OS already installed so that you can insert it into the slot, apply the power, and get started (this is referred to as "NOOBS," which stands for New Out Of Box Software). With NOOBS installed on your SD card, you can simply slip it into your Pi, apply power, and your Pi will boot. If so, you can skip ahead to "Installing Software." But if your Pi came without the OS installed, read on. . . .

You'll need a blank SD card or one recycled from another device, like a camera or audio recorder, and reformatted to erase existing data. The OS generally fills 2 to 3 GB, and the balance of the available memory is used for your application software and the code you write. For this chapter, with our focus on PureData, 8 GB is usually sufficient, but if you need to store lots of audio files or make long recordings (more about that later), you might want something larger.

As of this writing, your best starting place to obtain your OS is the Raspberry Pi Foundation at www.raspberrypi.org/. Go there with the browser on your *other*, "normal" computer. In the download section of their website, you will find a number of format options, including the aforementioned NOOBS. For those who are new to the arcana of operating systems, NOOBS is a great start; those with more experience might want to download a more minimal OS and add applications as you need them. Whatever your preference, you'll download the OS as a "disk image" file.

Copy the uncompressed image file onto your blank SD card, eject it from your personal computer, and install it in the card slot on the Pi. Connect your keyboard, mouse, and monitor if you haven't done so already and connect the power supply to the Pi and the wall socket.

When first powered up, your Pi will run some setup operations and eventually get you to a desktop image. Once there you will probably be prompted with a series of questions about things like what language you speak, what country you are in, and (very importantly) your network. Set up internet access at this time since it is needed to install the rest of your software. Follow all the on-screen instructions carefully and take notes if you change your password (you should). You might be asked to restart your Pi once or twice, but when you finish installing the OS, you can move on to installing the software you will use for writing your programs.

Installing Software

In the Raspbian OS (the variant of the Linux OS Debian that runs on the Pi), installing software is done through the Terminal application by typing short commands directly. Open the Terminal app by clicking its icon. You should get a black screen with some text at the bottom that looks something like this:

pi@rasbpberry:~$

This is the "command prompt."

To get a sense of how the Terminal works, type **ls** (that's a lowercase *l*, not an uppercase *i*) and press Return on your keyboard. The command **ls** stands for "list," and what you should see is a listing of all the files in the current directory. And what directory are you in? Type **pwd** for "print working directory" and you should see something like:

pi@rasbpberry:~$/home/pi/

There are many more useful commands you can type, and the internet is a great place to learn them, but for now you will stick to the ones used to install software.

To install additional programs on your Pi, use the command **apt-get**, which "gets" software from the internet. Because most of the software/applications you can get are stored in "repositories" on various servers, you *must* be online with the Pi (not your main computer) to install and update software—if you did not configure network access when you initially set up your Pi, do so now.

Before you start installing software on your Pi, let's make sure your OS is up to date. To do so type:

sudo apt-get update

You will see a list of various files to be updated. When it finishes and you return to the command prompt, type:

sudo apt-get upgrade

This will apply the updates (note: this can take many minutes if you haven't done it in a while). Once this finishes you are ready to start installing software.

Type the command **apt-get** followed by **install** and then the name of the software you want to install.[2] The full command for downloading and installing PureData is:

sudo apt-get install puredata

(The word **sudo** before the command stands for "super user do." Without going into detail about what "super user" is, just know that it's needed, and if your Pi ever asks you if you are "super user" or says you need to be, just type **sudo** before the appropriate command.)

Once you have typed this command and entered your user password, you will see lots of text while your Pi goes about busily installing PureData and all the little support files it needs. This might take a few minutes, but don't step out in case you are asked any more questions. When the process finishes, the Pi will return to the **$** command prompt, and if you didn't get any errors, PureData should be installed.

Go to the Raspberry Pi icon in the upper-left corner of your desktop and click; in the dropdown menu under Sound & Video, you should see PureData listed; if not, retrace your steps and try installing it again.

Assuming that you have PureData successfully installed, let's move on to using it to make our first patch. (If you do run into problems, the internet is your best source for answers. Fortunately for us the open-source community is a very large, very talkative one, and there are many forums where you can ask questions.)

MAKING YOUR FIRST PD PATCH

Launch PureData and a window should open that says "Pd" at the top. Go to the File menu, choose New, and a new "Untitled" window will open. This is called a "patcher window," and it is the main workspace where you will construct your "patches." Pd is known as a "flow control program," which is a collection of visual elements, called "objects," that get connected together with "patchcords" much the same way modules on an analog synthesizer (or the integrated circuits elsewhere in this book) get wired.

Let's create some "objects" and "patch" them together...

Go to the menu Put and choose Object. In the patch window, wherever your mouse is, you will see a dashed outline box. When you click, the box with a dashed outline will be there with a flashing cursor inside prompting you to type. Type *osc~* and then click anywhere in the window and the dashed lines should become solid and you should the object [osc~]. If not you will get a red error message in the Pd window helping you figure out the problem (did you spell it right?).

(A note about typographical conventions in this chapter: in Pd the graphical objects you work with on screen are “objects,” “messages,” and “number boxes.” In trying to describe those things in text (this book), I use the convention of opposing square brackets to denote an object by its name, like this [an_object], e.g., [osc~], and for messages with two left brackets, like this [a message[, e.g. [open[, which hopefully looks something like how these appear on the screen graphically.)

The object [osc~] produces a periodic waveform, like (surprise!) an oscillator, and is the fundamental building block of a synthesizer.

Go to the Put menu again, choose Number, click above the [osc~] object, and you should get a slightly different looking box, called a *number box*. As the name suggests, this object outputs a numeric value. Go to the Edit menu, choose Edit, and your cursor should change from a hand icon to a pointer. Click on the number box, move your mouse, and you should see the number change. Go back to Edit, choose Edit again (pro tip: use Control-E to go in and out of edit mode) and you should be able to click on your two objects and move them around inside the patch. Hover your mouse pointer over the lower-left corner of the number box until you see a circle, then click and you should see a “wire” start to grow out of the bottom of the number box. Hold down the mouse button and drag your wire until it reaches the upper-left corner of the [osc~] object; when you see the circle grow, let go; you should now have a “wire” (patchcord) connecting a number value to an oscillator (if you didn’t get it the first time, try until you get a feel for how to “patch” together objects). When you’ve done that, your patch should look like Figure 34.1.

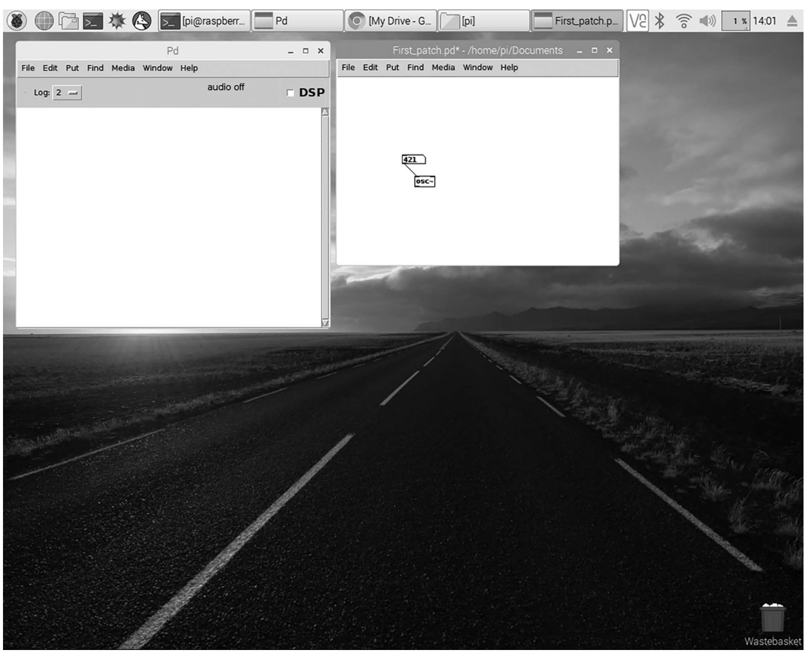


Figure 34.1 Number box connected to [osc~].

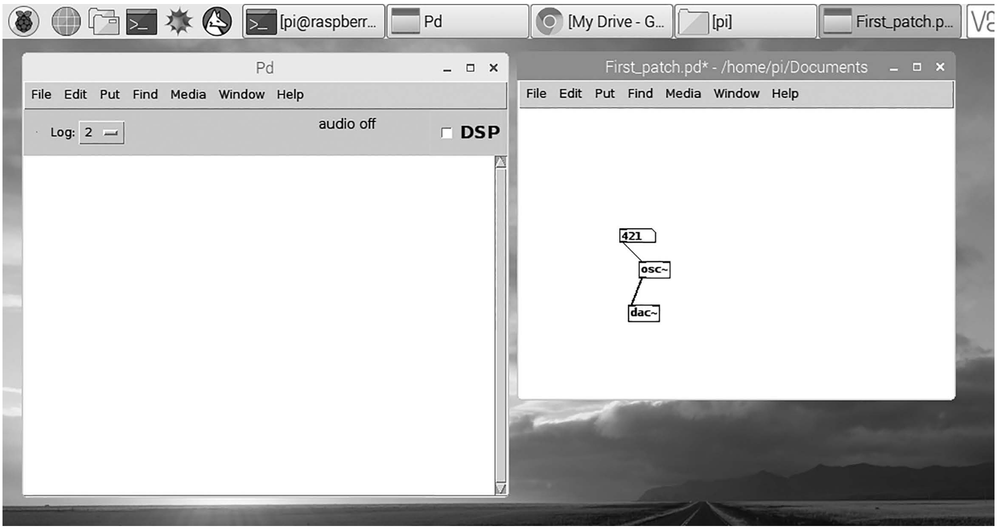


Figure 34.2 Adding a [dac~].

Now go back to Put, choose Object (pro tip: use Control-1 to make a new object without going to the menu), place it somewhere below [osc~], and type *dac~*, which is the object that represents the digital signal processor that the Pi uses to transform internal data into audible sound. Connect a patchcord from the lower-left outlet of [osc~] to the upper-right inlet of [dac~], like Figure 34.2.

Now mouse over to the main Pd window and you'll see the letters DSP with a checkbox next to it; clicking that box turns on the digital signal processor. If you haven't already, you should plug a pair of headphones or a set of speakers to the audio jack on your Pi. Now go back to your patch window, click on the number box, and scroll vertically until the numbers are in the low 100s and you should hear a tone—My First [Digital] Oscillator! If you don't hear anything, go to the speaker icon in the upper right of the desktop, click on it with the right mouse button, and choose Analog (if HDMI was initially enabled, the audio was trying to go through HDMI to your monitor, which might not have speakers). Hopefully by now you should have sound. You've made your first Pd patch, and you're on your way to working with audio on the Pi. (You might want to save this patch now so that in the future you can look back upon it fondly).

Now let's add a new object.

Under the Put menu, choose Message. As before you are going to click in the patch window and get an object, and there will be a blinking cursor prompting you to type: type "440" and click outside the box. Note that this new object, a message box, looks slightly different from the number box. Connect a patchcord from the outlet of the message box to the inlet (top) of the number box (Figure 34.3).

Lock your patch (menu Edit > Edit mode, or Control-E). If you click on the message box that says 440, you should see the number box jump to that value.

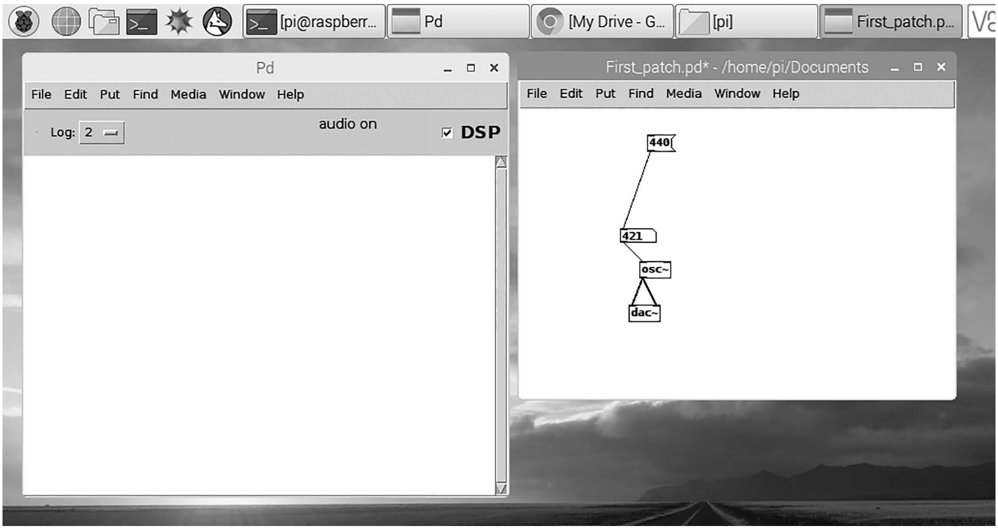


Figure 34.3 Adding a message box.

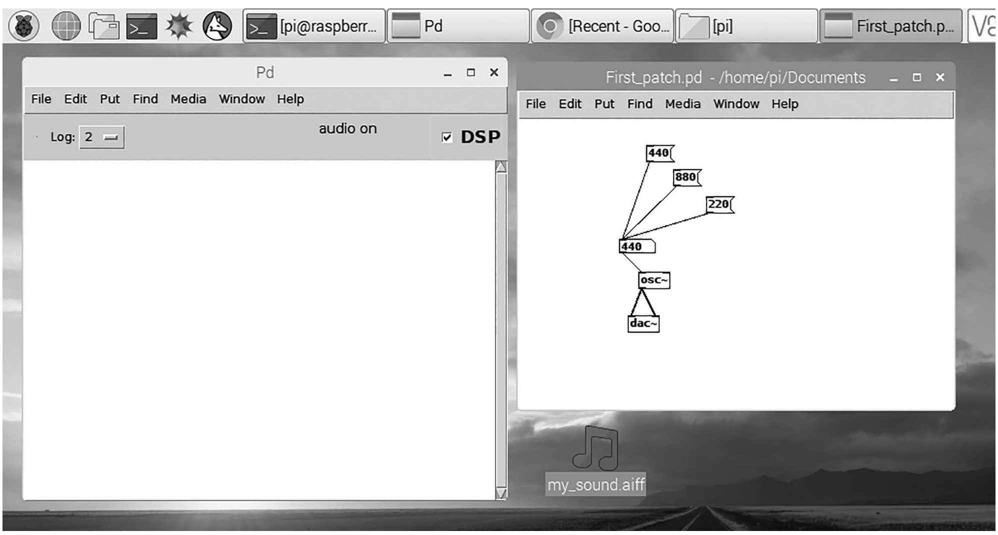


Figure 34.4 More message boxes.

Now unlock your patch and again go to Put > Message (or use the keyboard shortcut Control-E) and make another message box. In this one type 880 and connect its outlet to the inlet of the number box. Make a third message box with 220 in it and attach it to the number box as well, then lock your patch (Figure 34.4).

By clicking on each message box, the number box changes to that value and the oscillator plays that frequency (which in this case are musical octaves).

Congratulations, you have just made a primitive synthesizer! Experiment with adding more message boxes and typing different values. On the internet you can find charts showing what frequencies are which notes, e.g., A, C#, etc., and if you take the time to type those values into message boxes, you can make yourself an awkward "keyboard."

Even though this patch only makes simple tones, it's a glimpse of the sonic possibilities of PureData. Not only can you change the frequency of the oscillator, but you can also change its level, turn it on and off, alter its tone, and more. Likewise, you can copy, paste, and duplicate the objects in the patch to expand it.

YOUR SECOND PATCH

From simple tones let's move on to something more complicated: playing recordings.

Your Raspberry Pi, along with PureData, can work as a handy and inexpensive audio media player, and in this patch you'll learn the basics of playing back sound files from the SD card.

Start by opening a new window and creating a new object [readsf~].

Below that add the object [dac~] like we did in the previous example. Connect the outputs of [readsf~] to both inlets of the [dac~] (left and right speakers).

Above [readsf~] create a message box, type "open $1" in it, and attach its output to the inlet of [readsf~] (Figure 34.5).

Above the [open $1[message add a new object [openpanel] and connect it. Above the [openpanel] object, add a "bang" button. Bang is PureData's basic "pushbutton" to make things happen, and it is an object we haven't seen yet, so go to menu Put and choose Bang. Connect the bang button to [openpanel].

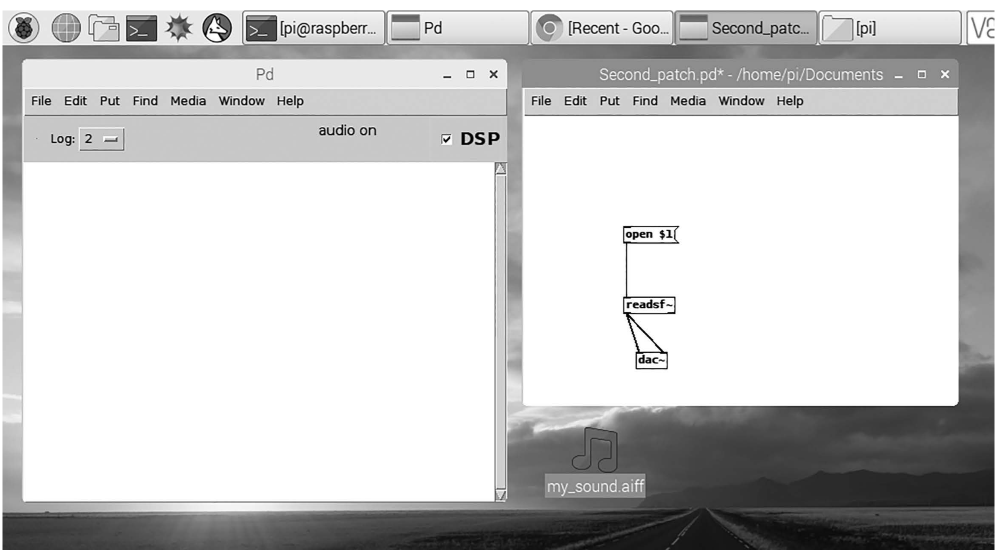


Figure 34.5 File player.

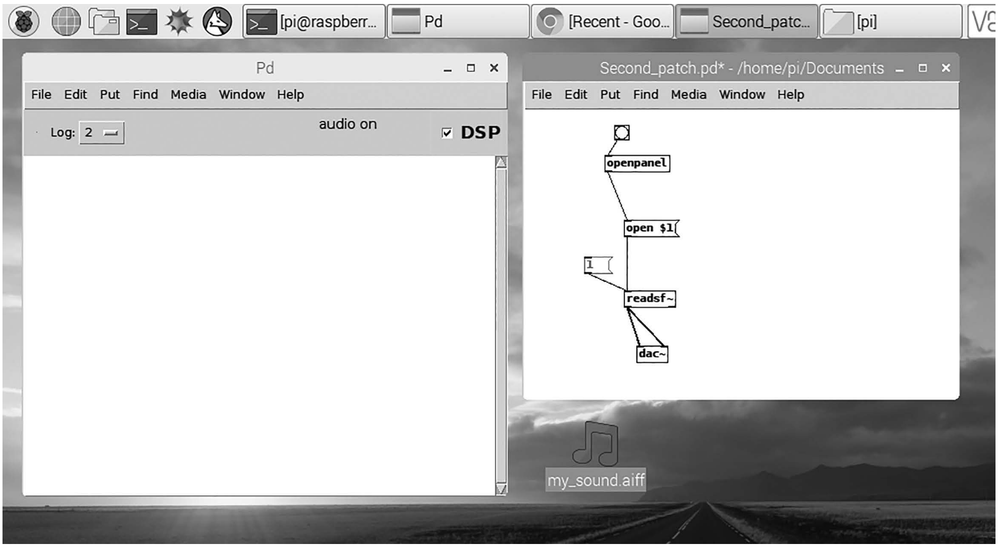


Figure 34.6 Adding openpanel to file player.

Now create a new message box with a 1 in it and connect it to [readsf~] (Figure 34.6).

Lock your patch (Control-E), click on the bang button, and a dialogue box will open so that you can navigate to a file you want to play; click OK to load. Note that the file will not play right away; rather, you have just told Pd *which* file you want to play. To start playing your file, click the [1[message box. You will have to wait for the file to play to the end, or you can unlock the patch and add another message [0[, which can be used to stop it (1 for play, 0 for stop—very binary).

A note about getting audio files onto your Pi: depending on what you are trying to do, you can, of course, just download sounds from the internet using the Pi's included browser (Chromium). However, if you want to work with files you already have, you can transfer them with a USB stick or use a cloud drive to upload/download them.

Unlock the patch and add another object [print] and connect it to the output of [openpanel]. Lock the patch, click the bang button, and, like before, navigate to a sound file. You should see the name of your file in the Pd window. In my case it's called "my_file.aiff," and it's on the Desktop, so I see "print: symbol /home/pi/Desktop/my_file.aiff."

Highlight the bang button and the [openpanel] object and delete them. In their place add a new object [loadbang] and connect it to the [open $1[message.

Look at your Pd window where you printed out the path to your file and copy everything after "symbol" and put it where the $1 was. In my case my "open" message becomes [open/home/pi/Desktop/my_file.aiff[. Yours, of course, will be different depending on the name of your file and where you put it.

Make two more changes to your patch: connect a patchcord from the output of your open message to the input of the [1[message. Likewise, take the *right* outlet of [readsf~] and connect it to the input of the [1[message.

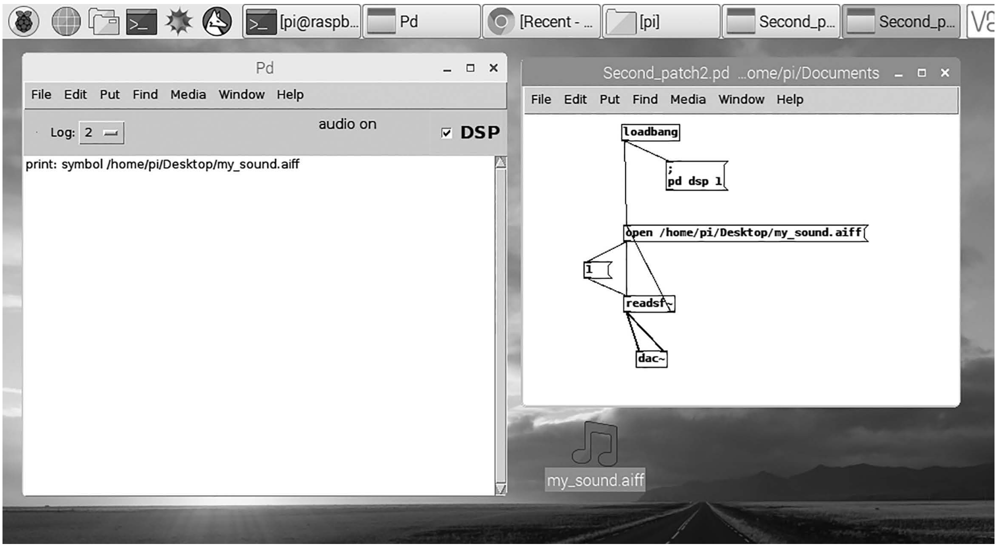


Figure 34.7 Adding [loadbang], dsp data and filepath to file player.

Finally make a new message box like this [; pd dsp 1[and connect the outlet of [loadbang] to it.

Your patch should look something like Figure 34.7 (your filepath and name might be different).

So what did we just make?

As might be apparent from the name, [loadbang] is an object that sends a bang when the patch opens. In turn that bangs on the "dsp 1" message to start the DSP audio system, just like clicking the DSP checkbox in the Pd window. It also bangs on the "open" message and, because we've given it the name of a file, that file will be opened. Likewise, it bangs the [1[message so the file will play from the beginning. Once the file finishes playing, the *right* outlet of [readsf~] bangs on the [1[message to play it again in an infinite loop (or leave this patchcord off if you don't want to have a looping file player).

To test your patch, lock it and click on the [open[message and make sure your file plays and repeats (if it doesn't check the Pd window for errors; most likely you either don't have the path right and/or the file name is wrong).

If the patch works, close the patch window, click the DSP checkbox in the Pd window so it's off, and then reopen the patch. If all is good, the DSP should come on, your file should play automatically, and it should play in a loop.

Now let's wrap it up . . .

The only thing left to do now is to start PureData and your patch automatically when you power up your Pi. To set this up, open the Terminal and type the following command:

sudo nano /etc/rc.local

Your terminal should change into the text editor "nano," and you should see the contents of the file rc.local. Use your down-arrow keys to move the cursor to the very last line and type the following:

pd -nogui /home/pi/Documents/media_player.pd &

The name of your patch and the path to it may be different. If you are not sure, go back to Pd, open your patch, and look at the title bar of the window: it will show you the name of the file and the path, then type that into rc.local (enter this path in the Terminal exactly as it appears). Save your nano text file with Control-O and exit with Control-X.

Now type:

sudo reboot -t now

If you did everything right, your Pi should shut down and restart, Pd should launch automatically, and your sound file should start playing. Congratulations, you have just turned your Raspberry Pi into an audio media player that will play and loop the sound file you choose whenever it's plugged in!

If it didn't work, open a terminal and type the full **pd -nogui** command prior, press return, and look for error messages from PureData.

WHERE TO GO FROM HERE?

In this short chapter, we have only touched the tip of the friendly iceberg that is PureData. Even from the simple examples here, it should be obvious there are many different directions you can go in the future. For example: with a few changes, our media player patch could have two different files playing out of two different speakers by duplicating the [readsf~] objects and connecting one each to the left and right sides of [dac~]. Likewise, you could play multiple files in order, one after another, or have files play in a random sequence (shuffle), using objects like [random] and [select] to do so. The internet has lots of examples for you, so start poking around. Variations on the start-up script can run any patch you write—very useful for gallery installations or digital instruments . . . just plug in and go!

AUDIO OUT (AND MAYBE IN)

As you have already found out, your Raspberry Pi has a 1/8-inch (3.5 mm) stereo headphone jack, into which you can plug a pair of headphones or computer speakers. But as you have probably also noticed, your Pi does not have an audio input. To get sound into your Pi, you are going to need some form of audio interface. A variety of companies make add-on boards for the Raspberry Pi (called "hats"), some of which

Figure 34.8 Audio settings panel.

provide audio input and output. These can take a number of forms, ranging from expensive studio-quality units for professional recording to cheaper ones for simple audio tasks. If you'd be satisfied with a basic audio interface, online you can find many that are simple USB "dongles" with one mono input (microphone) and one stereo output (headphones). These typically only cost a few dollars and offer a quick and easy way to get audio into your Pi and Pd.

Assuming you have found yourself an inexpensive USB interface, plug it into your Pi, open Pd, and go to the menu Media > Audio Settings. In the window that opens, make sure Input device 1 is checked, then click on the dropdown menu and select your interface (the one that *doesn't* say bcm2835) and set Channels: to 1 (Figure 34.8).

OTHER THINGS IN AND OUT (MIDI)

You might also want to connect some sort of keyboard, or other performance controller, to your Pi and Pd. The easiest way is with a MIDI (Musical Instrument Digital Interface) device, which come in two styles: older controllers that use special connectors and cables and require an interface and newer ones that work directly over USB. Look for a "modern" USB controller if possible; if you have an older device (and interface), you will have to investigate if it will work with Linux (Raspbian).

Plug in your controller and open PureData, go to the Media menu, and check that "OSS-MIDI" is selected. Go to the menu item MIDI Settings . . . and in the window that opens check to see if your controller shows up as an Input device. (You might have to click the button. If you plugged in your controller after opening Pd, try quitting Pd and reopening it.)

In a new patcher window, add the object [notein] and attach number boxes to the outlets; when you play notes on your keyboard, you should see their values displayed.

If you then open the Help patcher for [notein], you will see other useful objects for MIDI such as [ctlin] for controller values, [bendin] for the pitch-bender, etc. as well as their corresponding output objects. Also worth noting is the very helpful object [mtof], which stands for MIDI-to-frequency and allows you to convert MIDI note numbers (each note in MIDI is numbered 0–12, 13–24, all the way up to 127 for every octave in the western musical scale) into its frequency in Hertz, suitable for connecting to [osc~] and making a keyboard-controlled Pd synthesizer.

Something worth noting for these projects: Pd comes with a patch for testing both audio and MIDI in and out. If you go under the Media menu and choose the item Test Audio and MIDI, a patch will open that allows you to test your settings.

AND THEN THE REST OF THE WORLD. . . .

Not only is the Pi a cheap yet powerful computer, it's also born to be hacked.

As you may have noticed, your Pi has a set of 40 pins (called a header) where you can make physical attachments. This is how "hats" get attached, but it can also be a way to hook up things like buttons and sensors to individual pins, which means that your Pi can not only be used as a computer but also as other "physical computing" devices such as Arduino. There are plenty of instructions online about how to hook up a push-button to one of the Pi's pins and then read when it is pressed.

IN CONCLUSION

The Raspberry Pi microcomputer is an amazing platform for running many applications, especially PureData for sound creation. The small size and low price make it the perfect platform to run all kinds of sound projects, from sound installations, to stand-alone synthesizers, to MIDI controllers, and more. Now that you know the basics of Pd and have written few patches, you are ready to move on. A sea of possibilities beckons, as the briefest of web searches will reveal. Videos, forums, and even books abound. So cast off and get your Pi working. . . .

NOTES

1. Some of the material in this chapter is based on the course "Small Sound: Pure Data on Raspberry Pi" offered at the School of the Art Institute of Chicago. The author is grateful to be allowed to republish it in part here.
2. When using apt-get install to install software, you might not always know the name of the software you need to install; for example, a useful audio program to have along with Pure-Data is Audacity. Another useful command is **apt-cache search**, which is then followed by a keyword search. To look for Audacity, you would type **apt-cache search Audacity** and you will see it listed, and then you would type **sudo apt-get install audacity** (but take careful note of the upper- and lowercase letters: Linux is case-sensitive!). Likewise, if you were looking for photo editing software, you could type **apt-cache search photo**. . . . But beware: like any search engine, you can get hundreds of results!

CHAPTER 35

Data Hacking
The Foundations of Glitch Art

NICK BRIZ

The First Rule of Hacking (Chapter 2) states that "anything worth doing is worth doing wrong." In this chapter you'll apply that spirit of misuse to digital files, rather than electronic components. We'll be replacing circuit bending with databending and swapping our solder for code. That said, you will not be coding per se—this chapter is not about learning to program or how to properly use programs. This chapter is about corrupting code and misusing programs in the service of glitch art. Glitch artists employ various techniques (including hardware hacking) to instigate digital errors. Their tools are often esoteric, like the computerized Jacquard loom used by Melissa Barron to produce textiles of Apple II "crack screens."[1] Approaches to glitch art are often unpredictable and messy, as in the collaborative work of Kyougn Kmi and Daniel Rourke. In their cacophonous Glti.ch Karaoke project, the duo linked a hodgepodge of free apps—YouTube, Skype, TinyChat—to conduct disastrous duets, intentionally exploiting bad connections and high latency.[2] The techniques can get extremely complex, like designing entirely new file formats for the sheer purpose of breaking them, as Kim Asendorf did in his project ExtraFile.[3] But they can also be beautifully low tech, like UCNV's piece *New Vulnerability*, where the artist repurposes LCD screens as floor tiles that get progressively more glitched and cracked as visitors walk all over them.[4] While glitch artists deploy a myriad of tools, tricks, and techniques, they all share the same hacker ethic: to take a familiar piece of technology and do something unfamiliar with it.[5] For most glitch artists, however, it all started with a simple text editor.

DATABENDING

Aside from a computer, you will not need any specific tools or materials to undertake the experiments in this chapter. Although I will describe precise ways of misusing different types of software and files, don't get distracted by specifics; focus instead on the general approach. We will begin by opening a file with a program not intended for opening those kinds of files: we'll open a JPEG image file with a text editor, the first step in *databending*.[6] Like circuit bending, from which it gets its name, this is a process of trial and error, and it is a rite of passage for most glitch artists (like circuit bending is for many sound artists). You can use any text editor, such as TextEdit on a Mac, Notepad++ on Windows, or Nano on Linux. The trick is to avoid "document" editors that do fancy formatting, like changing text color or font (i.e., Word)—you need an editor that keeps things in "plain text."

Once you have chosen an editor, make a copy of a JPEG image you wish to bend and then open it with the text editor. As with circuit bending, you will aim to nudge your JPEG into a sweet spot between its correct appearance and a completely corrupted image. (It's easy to bend a file beyond readability, which is why you should start with a copy and save successive versions of the file as you experiment.)

Upon opening the JPEG file, the text editor will present you with lines of seemingly random text characters (what databenders like to call "ASCII barf").[7] Scroll down to somewhere near the middle of the file and type in some text—you could write a poem, tag your name, or type your password. Whatever you choose, it will be something that is not supposed to be embedded in this JPEG file, and this "wrongness" is precisely our goal. Once you have entered your text, save the file. Make sure that the filename maintains its. jpg or. jpeg file extension (some text editors try to change that); you might also add a suffix to the basic file name to make it easier to save successive modifications (i.e., "cat.jpg," "cat2.jpg," "cat3.jpg," etc.). Now open your bent JPEG in whatever app you use for viewing images. You should see something similar to Figure 35.1.

If you do not see any visible change, or are otherwise unimpressed with the result, repeat the process: open the file with your editor, insert some text, save the file, open with an image viewer. You could also select and copy a chunk of text from one location in the image file and paste it in several other locations or delete random sections

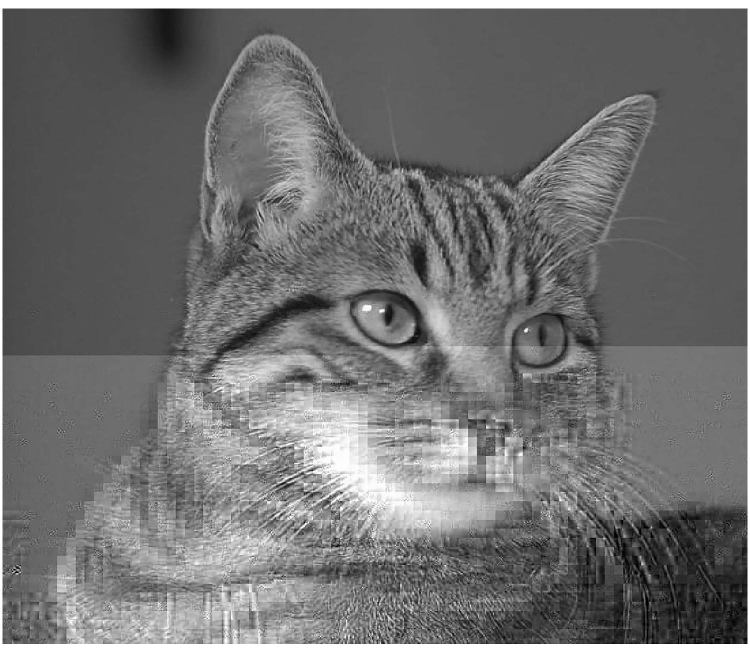


Figure 35.1 Public domain cat photo databent by Nick Briz.[8]

of text. You can use the editor's "find and replace" functionality to swap any character with another (for example, replacing every instance of the letter *a* with the letter *z*). Whatever you do, make your changes incrementally, saving each version and checking to see what the file looks like in an image viewer. You will notice that once you start making big changes to the file, your colorful glitches get replaced with a less aesthetically pleasing error message—"warning, this file appears to be corrupted."[9] The computer is on to us: in circuit bending, sometimes you go too far, adding a bend that causes the circuit to smoke and stop making sound; in databending, the worse-case scenario is an error message—less dangerous, but also less interesting.

IT'S ALL JUST A SERIES OF NUMBERS

So what exactly is going on here? What you see in Figure 35.1 are "digital artifacts" that are typically caused by bad file compression or bits of data gone missing over a slow internet connection. When we say something is digital, we mean that it is numerically represented. Every digital file on a computer—be it an image (.jpg,. gif,. bmp), video (.mp4,. mov,. ogv), audio (.mp3,. wav,. ogg), or text file (.txt,. md,. html)—is just a series of numbers. This collection of numbers, the raw data, is meaningless until it is interpreted by some program. (A program is itself also just numbers: numerically represented instructions that tell the computer what to do with the data.) What we experience as a digital image is the interaction between the file itself (the raw data) and the program designed to translate that data into a grid of colored pixels. Open your databent file in other programs that can display images and you may notice that your glitch manifests itself differently in each (Figure 35.2).

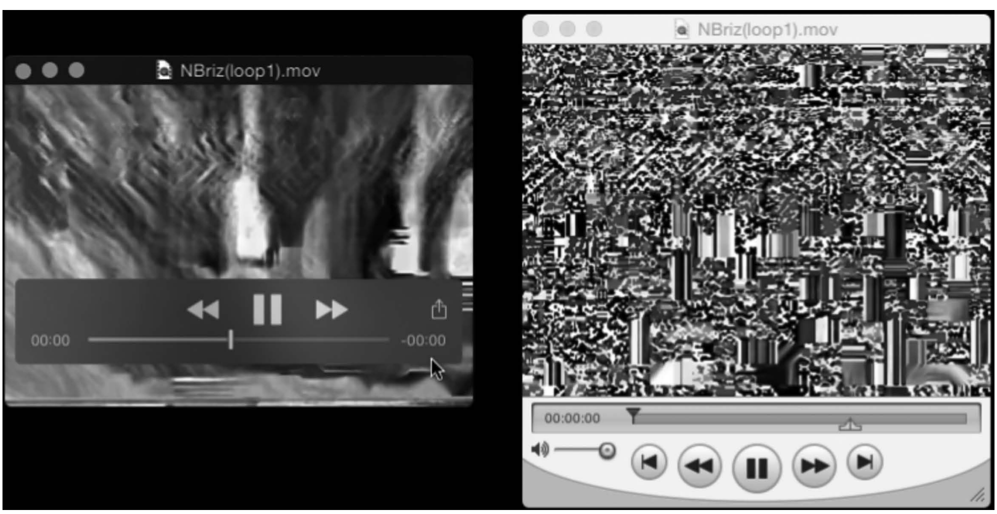


Figure 35.2 Screenshot documentation from the piece "an-*uh*-mit data," 2008, by the author, depicting a databent video file being played by two different video players, QuickTime X (left) and QuickTime 7 (right). Even though both players are playing the exact same file, *NBriz(loop1).mov*, they render entirely different images because of the way this file was databent.

Consider Figure 35.3, which shows a tiny image enlarged to show the individual pixels, with numbers superimposed to indicate each pixel's numeric value. Because this is a black and white image, each pixel is represented by a grayscale value, 0 being black, 255 being white, with 254 possible variations of gray in between.

Each image file is a collection of numbers stored in the computer's memory. Those numbers only take on the form of a picture when that file is interpreted by image viewing software, but if that same file is opened in a text app, each number is then interpreted as a letter, number, or symbol according to the ASCII character code. To a computer the value 78 could represent a dark gray pixel, but it could also represent the capital letter *N*, depending on the function of the software reading the file, just like the character sequence "sin embargo" will be interpreted as meaning one thing in English and something different in Spanish. If we open the file used to render Figure 35.3 with a text editor instead of an image viewer, we would see the words "This File Is Just Numbers."

Typically, computer files contain more than just the representational data (the values that correspond to text characters or pixel colors), they also contain *metadata*.

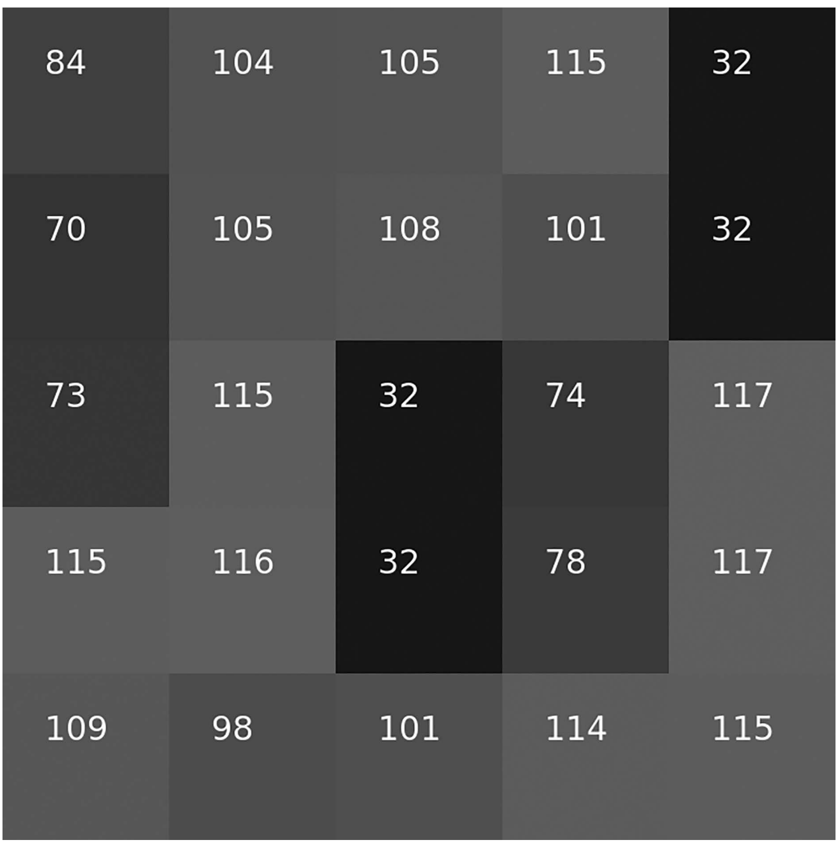


Figure 35.3 Tiny 5 px by 5 px image zoomed in with each pixel's corresponding byte value superimposed.

Metadata is data about the data: a text document might contain information about the colors of fonts to use while an image file could include its dimensions (width and height). Metadata contains information the program needs to decode the file properly. This metadata is contained in the part of the file known as the "header"; as the name implies, the header is usually located at the start of the file. This is why I told you to scroll to the middle of the file before databending earlier. If you add text too close to the start of the file, you risk manipulating the header. Headers are usually very sensitive, consisting of precisely organized values of fixed length, which is why "random" changes to this area of a file usually result in an error message when we try to open it in an image app after bending. That said, once you understand the structure of a particular file's header, it can be one of the more exciting spots to databend.

HUFFMAN HACKING

There comes a point in circuit bending where you go from licking your fingers and poking at random spots on a circuit board to soldering in new connections in precise locations. Some say this is the point where "bending" becomes "hacking." To take our data misuse to the next level—the land of dry fingers and wet solder, so to speak—we will need to download a *hex editor*. A hex editor is a tool that allows you to view and edit the raw data of any file. They're typically used by programmers for debugging, often when a file has been unintentionally corrupted. Naturally, we'll be doing the exact opposite. There are loads of free hex editors out there to download. Popular ones include 010 Editor on Windows, HexFiend on Mac, and GHex on Linux.

Once you have downloaded and installed a hex editor, use it to open up a fresh copy of a JPEG. You will see a series of numbers, but you will also see letters. You may have heard that computers only understand "binary code." This means that at the lowest level of the computer's digital circuits, it encodes values as a series of high and low voltages, which we call 1's and 0's in binary arithmetic. So while we typically represent the number seventy-eight as 78 using the decimal system, in the computer's binary system of high and low voltages, it would be 1001110. Because the binary system only has two symbols, it takes seven numerals to spell out a value that only takes two numerals in decimal (with a system of 10 symbols to choose from).

Hexadecimal is yet another system for representing numbers. It uses 16 symbols—the numerals 0–9 and the letters A–F—and so we can represent a larger quantity with a shorter string of digits than binary's 0 and 1 and even less than decimal's 0–9. And because programmers value efficiency over most other things, hex editors by default represent the raw data, the numbers inside any file, in hexadecimal, where 0 (decimal) becomes 00 (hex), 255 (decimal) becomes FF (hex), and 78 (decimal) becomes 4E (hex) (Figure 35.4).

Depending on your specific hex editor, you might have the option (somewhere in the preferences) to switch from hex to decimal or binary. Most hex editors will usually display a split window, with the hex code on the left and the ASCII on the right (text interpretation, similar to what you saw with your plain text editor in our first example). If you compare the raw hex to the ASCII text, you might notice that many

Binary (base 2)	*Decimal (base 10)*	*Hexadecimal (base 16)*
0	0	0
1	1	1
10	2	2
11	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
. . .	. . .	. . .
11111110	254	FE
11111111	255	FF

Figure 35.4 Machine code values in different numeral systems.

values do not have any corresponding text character and just appear as blank spaces or dots, depending on your editor. This brings up a subtle but important point: when you opened your JPEG in a text editor, you were seeing a text (ASCII) representation of the data, not the raw data itself. As noted earlier, a text editor simply decodes the data into text, the same way your image viewer decodes it into a grid of colored pixels; if a number has no ASCII character associated with it, the baffled editor inserts an arbitrary placeholder (the computer equivalent of your drawing out a "hmmmmm . . ." in response to a difficult question). This is precisely why hex editors are such useful hacking tools: you're not limited to editing only the values the text editor can decode as text; instead, you have access to every raw byte of data in that file. This opens the door to some really advanced hacks.[10]

Hex editors are ideal for databending sound or video files. If you try to databend an MP3 file with a text editor, you often just end up with an unplayable file. MP3 files organize data into a series of "frames" each starting with their own header, which means the headers are distributed throughout the file and thus the chances of corrupting it beyond interpretation are much higher than with a JPEG. But with a hex editor, you can identify the characters in the MP3 frame headers, allowing you to avoid manipulating those. Then instead of getting an unplayable file, you'll get a glitched MP3, which usually sounds something like a drowning dial-up modem or fax machine

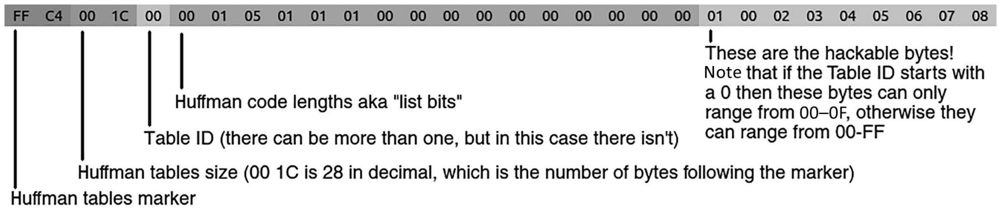


Figure 35.5 Huffman table.

(sounds you've probably never heard if you were born after 1999). See the Hacking MP3s section that follows for more details on this.

Unlike our first exercise with a JPEG file, this time we don't want to avoid the header, we want to hack it. A JPEG is a form of lossy image compression, which means it is the result of algorithmically processing the numbers in an image such that we remove pixels we are not likely to notice are missing in order to store a readable image in as few bits as possible. Part of how JPEGs achieve this is by creating a special section in the header of the file with information on how to translate its compressed set of bits back into the correct bytes of data to represent the colors. This section of special codes in a JPEG file is called the Huffman tables, and hacking these can lead to drastic glitches (Figure 35.5). To identify the Huffman tables, we use CMD+F (or CNTRL+F) to Find their marker: FF C4. These two bytes might appear multiple times throughout the entire image file, but their first instance is likely the Huffman tables. The two bytes that follow FF C4 denote the size of the Huffman table (i.e., D4 would mean that the table is 212 bytes long); the byte following that is the first table's identifier (it's worth noting that there can be more than one Huffman table in an image file); and the sixteen bytes following that contain what are known as the code lengths or list bits. You want to avoid editing any of these bytes, from the FF C4 header through the code length/list bytes (i.e., the first 21 bytes starting at FF C4), but the bytes that follow those are ripe for the bending.

Be careful not to add any new bytes of data while editing; instead, you want to simply replace bytes, for example, swapping where it might say 03 with 09. It is worth noting that there are two kinds of Huffman tables (AC and DC); the table identifier byte (the fifth byte from the start) will start with a 0 when it's DC and a 1 when it's AC. This is relevant because DC tables can only have bytes ranging from 00 to 0F whereas AC can use the full byte range from 00 to FF, so you'll want to keep that in mind when replacing the original bytes with your own. Last, a small change goes a long way in the Huffman tables; changing just a single byte is enough to glitch the entire image because any change here redefines how all the data in the file should be interpreted (Figure 35.6).

Keep in mind everything we have discussed pertains specifically to JPEG headers. Each image file type, PNG, GIF, etc., has its own form for compressing and storing image data. This means that the type of information contained in the header is different, so this particular Huffman Hack won't work on anything but a JPEG file. But it also means that when you discover header hacks appropriate to other image file formats, the digital artifacts they create might be very different.[11]

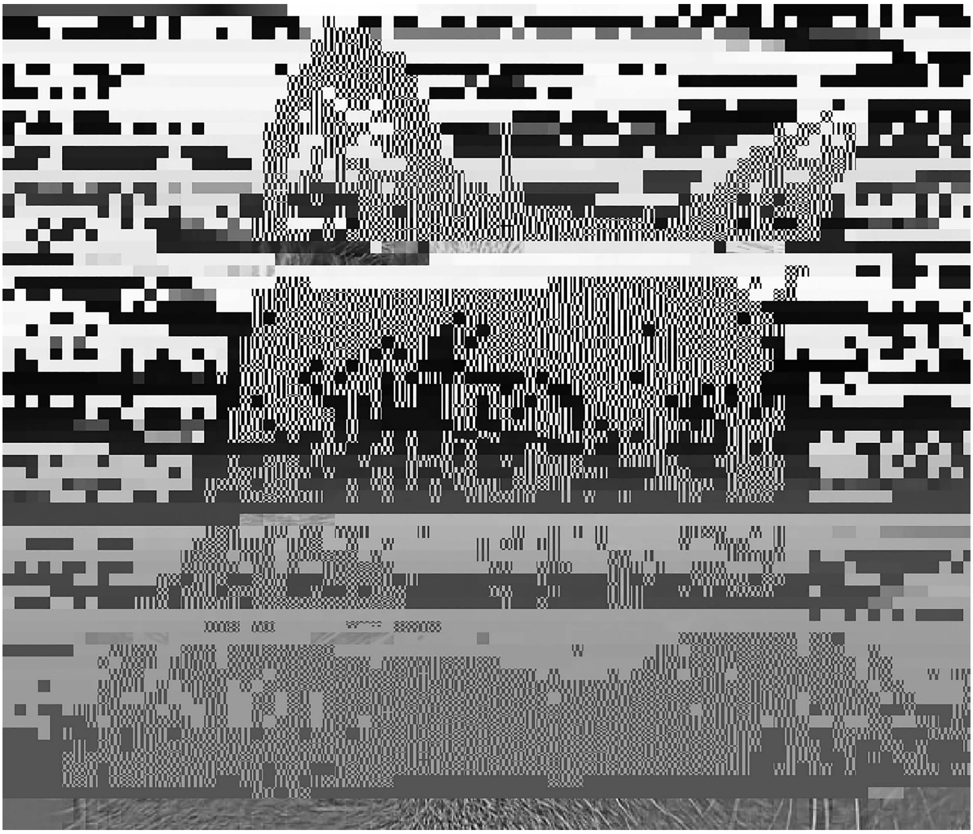


Figure 35.6 The same cat from Figure 35.1, this time with only a single byte changed in the Huffman table (from 03 to 09).

HACKING MP3S

Many of the same techniques we use on JPEGs can be applied to other file formats, including MP3 audio files. The general idea is the same: open the file in an editor, remix, save a copy, and listen to the result. That said, your first attempt is likely to result in an error message: "this file appears to be corrupted." As with any other file type, there are a few considerations specific to MP3 files to keep in mind in order to hit that glitchy sweet spot between uncorrupted and too corrupted to play. First, when it comes to audio files, it's best to avoid databending with a *text* editor and instead use a *hex* editor (the same is true for video files). You typically (with some exceptions—see the section on Huffman Hacking) want to avoid manipulating the header of the file, where the metadata resides, when you're hacking at the raw data in a hex editor. With image files this is easy because the header is always contained at the top of the file: skipping past the first few lines before hacking any data nearly always guarantees an uncorrupted header and thus a glitchy image instead of an unreadable one. With MP3s, however, there are multiple "frame" headers that are scattered throughout the file. In order to avoid unintentionally editing one of these frame headers, you'll need to know how to find them.

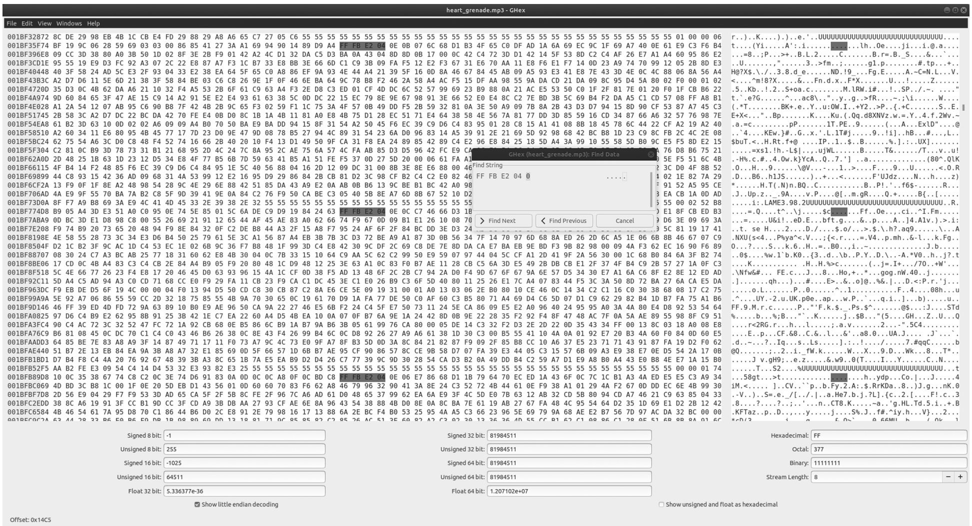


Figure 35.7 An MP3 file opened with GHex (Linux), with a search for every instance of its frame headers FF FB E2 04.

While frame headers vary between different MP3 files, the frame headers within a given MP3 file are all identical. Once you've figured out a particular MP3 file's header pattern, you can quickly find them all using your hex editor's search functionality. These headers will each be four bytes long and always begin with the byte FF. For example, a header might be: FF FB A0 40. Once you know your MP3 file's frame header pattern, you can search for every instance of it and simply avoid editing any of those bytes, opting instead for editing any of the bytes between the appearances of the frame headers. Although it's not necessary to understand what the bytes in the frame header mean, if you're curious to know what the metadata in these bytes tells us about the encoding of the MP3 (and thus how a program should decode the file), see Figure 35.9 at the end of this chapter.

The last consideration to keep in mind while hacking MP3 files is to keep the same number of bytes between the frame headers. When you edit the file, you should replace the bytes of data that are already there, rather than adding or removing data. So it's best to avoid manually copying and pasting large chunks of data. Instead, you should use your editor's find and replace functionality: search for a byte (which does not appear in your header pattern) and replace every instance of it with a different byte. Save a copy of the hacked file, listen to it, repeat. For example, in the MP3 file in Figure 35.7, we could do a find for "4F" (which does not appear in its header of FF FB E2 04) and replace every instance of it with "F4." We then save a copy of the newly edited version of the file and listen to it in a media player.

To recap these considerations:

- When hacking MP3 files, it's best to use a hex editor (over a text editor) so that we can find our MP3 frame headers to avoid editing them.
- Edit only the bytes between the frame headers.
- Keep the number of bytes between frame headers the same as the original file.

That's all well and good . . . but only if you know the pattern of your MP3 file's frame header.

IDENTIFYING YOUR MP3 FILE'S FRAME HEADER PATTERN

Most MP3 file headers start with FF FB; this is the first half of a header for an MP3 file using MPEG-1 Layer-3 containing no copy protection. This tends to describe the vast majority of MP3 files. So searching for "FFFB" is the best place to start. Once you find your first instance of that pattern, include the subsequent two bytes (for a total of four bytes) in your next search. For example, when I searched for "FFFB" in the MP3 file in Figure 35.7, the two bytes the followed the first instance of "FF FB" were "E2 04," so I then searched for "FFFBE204." If you notice that this pattern repeats often with a consistent number of bytes between the appearances (no need to count, you can usually eyeball it), then you've found the header! However, if this is not the case, try the next instance of "FFFB" and see if its subsequent two bytes are different from the last one. If it is, repeat your search with this new pattern and see if it appears consistently throughout the file. It usually doesn't take more than a couple of tries to find it.

There is a chance your MP3 file is non-standard and doesn't begin with FF FB, and while it will always start with the hex byte FF (or 1111 1111 in binary), it's not very effective to simply search for "FF" in our hex editor because this byte will likely show up way too many times on its own. However, if we search for the first byte and a half, FF F (in the example header prior), we're more likely to spot it. That said, not every MP3 has frame headers starting in FF F (or 1111 1111 1111)—see Figure 35.8.

Hex	*Binary*	*meaning*
F	1111	All four bits are part of the MP3 sync code (used to find the header).
F	1111	All four bits are part of the MP3 sync code.
F	1111	The first three bits are part of the MP3 sync code. The last bit, in combination with the next bit below (i.e., 11), tells us which MPEG version this was encoded with. In this case 11 translates to MPEG version 1.
B	1011	The first is used to determine the MPEG version (see prior), the second and third bit tell us the layer (i.e., 01, which is Layer 3), and the last bit tells us if there is copy protection (i.e., 1). In this case there is no protection; if there was the last bit would be a 0.
A	1010	This byte (all four bits) tell us the bitrate; in this case 1010 is a bitrate of 160 kbps.
0	0000	This byte tells us the sample rate, in this case 0000 is a sample rate of 44,100 Hz. Had it been 0100, this would be a sample rate of 48,000 Hz, or 1000 would be a sample rate of 32,000 Hz.
4	0100	The first two bits contain channel information; in this case 01 means Join Stereo. When set to Joint Stereo (like this example), the latter two bits tell us the mode of joint stereo.
0	0000	The first bit tells us if the MP3 file has a copyright (0 means it does not), the next bit tells us if it's a copy of the original file or not (0 means it is). The last two bits tell us if there are emphasized frequencies (00 means there are not).

Figure 35.8 An example of an MP3 frame header, FF FB A0 40, and its meaning.

Only those MP3s using MPEG version 1 or 2 will begin with FF F; if the MP3 is using a non-standard MPEG version like 2.5, the first byte and a half will be FF E (or 1111 1111 1110).

If you are working with a non-standard MP3 file, it may start with FFF or FFE, so you might have a little more to search for before you find the pattern of four bytes that constantly repeats throughout the file. Again, although it's highly likely that your headers start with "FFFB," it's helpful to remember that on some occasions they will not. For this reason we've included a chart (Figure 35.9) that includes every possible start (first two bytes) of an MP3 file's frame header as well as what it means for reference.

Hex	Binary	Meaning
FFE1	1111 1111 1110 0001	MPEG-2.5; Layer (reserved); no protection
FFE9	1111 1111 1110 1001	MPEG (reserved); Layer (reserved); no protection
FFF1	1111 1111 1111 0001	MPEG-2; Layer (reserved); no protection
FFF9	1111 1111 1111 1001	MPEG-1; Layer (reserved); no protection
FFE3	1111 1111 1110 0011	MPEG-2.5; Layer-3; no protection
FFEB	1111 1111 1110 1011	MPEG (reserved); Layer-3; no protection
FFF3	1111 1111 1111 0011	MPEG-2; Layer-3; no protection
FFFB	1111 1111 1111 1011	MPEG-1; Layer-3; no protection
FFE5	1111 1111 1110 0101	MPEG-2.5; Layer-2; no protection
FFED	1111 1111 1110 1101	MPEG (reserved); Layer-2; no protection
FFF5	1111 1111 1111 0101	MPEG-2; Layer-2; no protection
FFFD	1111 1111 1111 1101	MPEG-1; Layer-2; no protection
FFE7	1111 1111 1110 0111	MPEG-2.5; Layer-1; no protection
FFEF	1111 1111 1110 1111	MPEG (reserved); Layer-1; no protection
FFF7	1111 1111 1111 0111	MPEG-2; Layer-1; no protection
FFFF	1111 1111 1111 1111	MPEG-1; Layer-1; no protection
FFE0	1111 1111 1110 0000	MPEG-2.5; Layer (reserved); CRC protection
FFE8	1111 1111 1110 1000	MPEG (reserved); Layer (reserved); CRC protection
FFF0	1111 1111 1111 0000	MPEG-2; Layer (reserved); CRC protection
FFF8	1111 1111 1111 1000	MPEG-1; Layer (reserved); CRC protection
FFE2	1111 1111 1110 0010	MPEG-2.5; Layer-3; CRC protection
FFEA	1111 1111 1110 1010	MPEG (reserved); Layer-3; CRC protection
FFF2	1111 1111 1111 0010	MPEG-2; Layer-3; CRC protection
FFFA	1111 1111 1111 1010	MPEG-1; Layer-3; CRC protection
FFE4	1111 1111 1110 0100	MPEG-2.5; Layer-2; CRC protection
FFEC	1111 1111 1110 1100	MPEG (reserved); Layer-2; CRC protection
FFF4	1111 1111 1111 0100	MPEG-2; Layer-2; CRC protection
FFFC	1111 1111 1111 1100	MPEG-1; Layer-2; CRC protection
FFE6	1111 1111 1110 0110	MPEG-2.5; Layer-1; CRC protection
FFEE	1111 1111 1110 1110	MPEG (reserved); Layer-1; CRC protection
FFF6	1111 1111 1111 0110	MPEG-2; Layer-1; CRC protection
FFFE	1111 1111 1111 1110	MPEG-1; Layer-1; CRC protection

Figure 35.9 Every possible MP3 file frame header (first two bytes only); the most common pattern is FF FB.

THE WRONG TOOL FOR THE JOB

Once you realize that all computer files are just numbers waiting to be interpreted in a particular way, you will understand that a JPEG isn't the only thing you can open in a text editor and a text editor isn't the only software you can use to open a JPEG. As enticing as (I hope) this sounds, you will soon discover that when you try to open files with programs not intended for those files, you will usually be met with error messages. That said, over the years glitch artists have discovered various ways of fooling programs into decoding any file. For example, make a copy of any file in your computer and replace its file extension with. raw—i.e., rename song.mp3 to song.raw, and Photoshop will be convinced the data should be interpreted as pixels. Now, because there is no image header in this music file, Photoshop won't know what the "image" dimensions are or whether to interpret the numbers as grayscale values or color values. Fortunately, in place of an error message, Photoshop will present you with a window asking you these questions; answer these and, behold, your song.mp3 becomes a noisy mosaic. We can now apply image filters (which are, again, just numbers designed to manipulate other numbers in a particular way) to our music file. Since we're now inside a powerful image editor, maybe you'd like to hear what an image blur would sound like? Apply the effect to the picture, save the file, rename it back to. mp3, and listen to find out.

We can also go the other way, interpreting images as sounds by opening image files in audio editors. What might our JPEG sound like? In turn, what would it look like if we apply audio effects, like reverb or distortion, to image files?[12]

Glitch artist Jon Satrom taught a glitch art class at the School of the Art Institute of Chicago for a number of years. The first assignment he would give was called "the wrong tool for the job." The assignment was simple: take a piece of software and use it for some purpose other than its intended use. That could mean opening up files with the "wrong" software, but the assignment also extends to using Microsoft Excel as a real-time performance tool, for example.[13] While the techniques I cover in this chapter open a host of possibilities, it's important to remember that every program on your computer has the potential to be misused. More important than any specific databending technique discussed here is realizing that on computers we are always just interacting with numbers. There are myriad ways for the numbers stored in files (data) to be misinterpreted and the numbers stored in programs (instructions) to be misused. To quote the iconic glitchy net art duo jodi.org, "the mistake is nothing wrong, the computer keeps working. Something wrong still works, there's nothing wrong with something wrong."[14]

NOTES

1. http://melissabarron.net/2010screencaptures/index.html
2. I produced a short video doc about the project called "there's a huge noise in the middle of this: the ha[ng]ppenings of Glti.ch Karaoke," available here, https://vimeo.com/58901196, their official website is http://glti.ch
3. The ExtraFile project website is no longer online, but you can access a copy on the Internet Archive, https://web.archive.org/web/20170713185601/http://extrafile.org/about/

4. *New Vulnerability* was originally exhibited in Tokyo, the artist's hometown, in 2012. It's worth mentioning that toward the end of the opening night, one gallery visitor got so excited by the piece he started jumping on the screens, which triggered a surge, causing a power outage in the building—serendipitously the last glitch of the night. Documentation for both iterations can be found on the artist's site, https://ucnv.org/newvulnerability/
5. I co-organized an international conference called GLI.TC/H from 2010–2012 that brought together a diverse group of interdisciplinary glitch artists. This was the one sentiment we could all agree on.
6. One of the earliest artists to document and share this process within the glitch community was stAllio! (aka Benjamin Berg), who popularized a particular variation on this technique he called "the wordpad effect"; the original post from 2008 can still be found on his blog, http://blog.animalswithinanimals.com/2008/08/databending-and-glitch-art-primer-part.html
7. ASCII is essentially just plain text, a standard for mapping binary codes to text characters, https://en.wikipedia.org/wiki/ASCII. Those of us involved in the GLI.TC/H community would generally refer to any large dump of misformatted or misinterpreted data displayed on a screen as "barf" (another example is "RAM barf," when the various kinds of data stored in RAM are played through a media player, as is the case in Cory Arcangel's work Data Diaries, https://anthology.rhizome.org/data-diaries-2003), which is why the festival's logo was always some variation on a file icon "barfing" some kind of data, http://gli.tc/h/imgs21k12/post2112_logo.gif
8. https://commons.wikimedia.org/wiki/Felis_silvestris_catus#/media/File:European_shorthair_procumbent_Quincy.jpg
9. What is or isn't aesthetically pleasing is obviously subjective. Glitch artist Jon Satrom incorporates error messages into much of his work, especially in his live "prepared desktop" performances. His entire Twitter account is devoted to archiving error messages, https://twitter.com/jonsatrom
10. For examples of more advanced hex editor hacks, check out Party Time! Hexcellent! (aka Rachel Weil)'s ROM glitches on her site, www.partytimehexcellent.com/ and www.nobadmemories.com/
11. Glitch artist Rosa Menkman has documented different image file type artifacts in her pdf, "A Vernacular of File Formats," www.slideshare.net/r00s/lofi-rosa-menkman-a-vernacular-of-file-formats. Similarly, Evan Meany's Ceibas Cycle project contains case studies for bending various video formats, www.evanmeaney.com/ceibas/ (you will need Flash player enabled on your browser).
12. For further suggestions in audio manipulations of image files, refer to glitch artist Hellocatfood (aka Antonio Roberts)'s Audacity Databending tutorial on his website, www.hellocatfood.com/databending-using-audacity/
13. This is actually something Satrom has done in various performances, including his piece Windows Rainbows & Dinos, documented on his website, http://jonsatrom.com/---/windows-rainbows-dinos/index.html
14. From a video interview with joid.org conducted by Vice's Motherboard, see www.vice.com/en_us/article/gvvndq/jodi-something-wrong-is-nothing-wrong

PART 5

Connecting

CHAPTER 36

Handmade Sound Communities

LISA KORI AND DAVID NOVAK

FROM DO-IT-YOURSELF TO DO-IT-TOGETHER

In the decade since *Handmade Electronic Music* began to worm its way around the globe, clusters of handmade audio enthusiasts and collectively organized projects have formed in a broad geographical swath of sound art and electronic music scenes. Some were directly inspired by Collins's approach to hardware hacking, taking up new experiments enabled by the proliferation of handmade audio schematics online; others assembled projects from scrap electronics and readily available components, inspired by hands-on learning from like-minded creative souls. Participating in handmade audio practices has altered the lives of many artists and musicians and laid the groundwork for new community and collective praxes in electronic sound arts.

Home electronics have a long history in "do-it-yourself" (DIY) hobbyist practice, through which individuals take control of the invention and construction of technologies, creating alternatives to corporate production and the creative limits of consumer design. However empowering to the individual, the DIY ethos can reinforce the misleading ideal of singular innovations created by isolated individual geniuses in a linear progression of new objects and styles. Although we know better, new art and technology can seem to emerge sui generis in the contributions of solitary heroic creators, who break the rules all by themselves. It is a short leap from imagining the invention of counterpoint in Bach's mind to lionizing Steve Jobs cooking up Apple in his parents' garage or ascribing the invention of analog synthesis to Bob Moog and Don Buchla.

This chapter reconsiders the history of handmade electronic music not as the product of discrete independent actors but as a global network of collective activity, based in interactive workshops that serve many different social and aesthetic goals. One way to understand the art of hardware hacking is as a social and sonic practice of community building, in which people "do it *together*" (DIT) in groups that bring different abilities, needs, skills, and worldviews together to generate new forms of experimentation. We are especially concerned with how DIT electronic sound communities have developed among groups that are under- or unrepresented in existing histories of electronic music and experimental sound arts and whose access to electronic infrastructures has taken varied forms, both within and outside the West. We draw on our ethnographic fieldwork as well as interviews with international collectives, including:

- Sound-making communities in the Global South, particularly Brazil and Indonesia.
- Organizations oriented toward a diverse cultural field of participants, especially practitioners of sound technology who identify as female or non-binary in gender.
- Collectives that align technological learning with the social politics of community outreach, education, and accessibility.
- Spaces that build community through musical experiences, and an attempt to resist financial pressure to commodify cultural practices.

In addition, we include parallel innovations in popular culture that are often excluded from institutional narratives of electronic music and sound art: guitar feedback, pedal boards; DJing and turntablism, pause tapes, and beatmaking; experimental sound in film and video; performance art, social criticism, and independent media projects.

As Brazilian sound artist, experimental musician, and arts organizer Vanessa De Michelis explains, circuit bending has had an impact in creating new community approaches to art, technology, and cultural politics (Figure 36.1):

> [By] developing electronic circuits, they found a way of bending other things. Firstly, they bent the precarity of their conditions of access to electronic parts and equipment and the black box of technology. But secondly, and more importantly, they found a way of bending their isolation in the hegemony of Northern hemisphere voices that dictate aesthetics and ways of being, and created a circuit of long lasting friendships that bent the notions of what a noise community and circuit should do, and look, and sound like.[1]

Figure 36.1 Azucrinoise Workshop at SESC Belenzinho 2011, led by Vanessa De Michelis and Manuel Andrade.

Photo credit: Manuel Andrade.

The function of building and bending circuits is twofold: circuits form the basis of sound objects that open up and subvert the expected use of a technology, but they also build social connections between people who are willing to explore collective techniques of creating, living, and being together. Handmade sound communities reframe the exchange values of artistic activity, turning away from the production of commodified objects toward shared contexts of learning that highlight new possibilities for technological creativity in a global society.

WORKSHOPPING COMMUNITY: HACKING THE OBJECTIFICATION OF ELECTRONIC SOUND

In many handmade electronic audio communities, the teaching/learning workshop plays a central role. Hacking workshops use sound technologies to generate new social relationships and new ways of thinking creatively in everyday contexts. Since building, modifying, and bending circuits is a tactile process, one of the best ways to share and disseminate knowledge is in workshops that present basic skills, such as soldering and circuit building, or that focus on the creation of a particular sound tool or electronic object. Workshops can sometimes provide artists a source of income tied to their music or sound art practice and offer a flexible and open-ended model for educators to connect art and technology for practitioners less experienced with electronics. More often, however, these are collectively organized tuition-free or sliding-scale events that do not emphasize a particular pedagogy or instructor and move away from lecture-based instruction to generate critique in more open experimental models. Many workshops foreground ethical questions of living with technology while also creating opportunities for members of the public to acquire technological literacy in their artmaking by literally opening up everyday electronics otherwise "black-boxed" in consumer audio products. Because circuit-bent instruments "have often been built from junk, and/or mess with the innards of mass-produced consumer products," Trevor Pinch argues, they "mount a challenge to the mass-consumer society of modern capitalism and its deleterious environmental effects."[2] Nozu Kanami of the Osaka-based group Destroyed Robot describes his bent noise machines as manifestations of *hanzoku waza* (rule-breaking techniques) that resist the flattening effects of mass-market technologies: "Tamiya [an electronics hobbyist company] has this huge contest, but they have this rule that you can't enter unless you use genuine parts manufactured by Tamiya. That's what I mean. . . . I think many Japanese toys hinder creativity. And I think it's wrong for people to be satisfied with such toys."[3] In learning how to take apart such objects, hardware hackers unlock new ways of thinking about what, and who, makes musical sound.

For many groups, hardware hacking shifts the focus of music technology from inventing new products to exploring processes of improvisation and emergence with existing materials. This emphasis on transitory events rather than fixed objects suggests an alternative politics of social aesthetics that challenges the commodity form of art. The larger goal is not so much the specific artworks produced as the ongoing social process of sharing knowledge and skills. Waft Lab, based in Surabaya, Indonesia, sponsors circuit-bending events such as a "Fun with Knobs" workshop, intended to reach students and musicians who might not ordinarily approach consumer technologies with a spirit of openness and experimentation ("Art is fun, art is easy . . . how great it felt when we were kids who played with the diverse sound of the universe!") (see Figure 36.2).[4] Similarly, the pioneering BENT festival provided many musicians in early 2000s New York City with their first taste of cracking open a children's toy and unleashing a torrent of uncontrollable sounds. These sounds arose from assemblages of broken junk not easily categorized as either artworks or musical instruments. The experimental and hands-on nature of handmade audio workshopping stresses the outcomes of creative labor as ongoing and adaptive processes that, in turn, feed back into community building.

Beyond functioning as sites of learning and exchange, workshops and collectives can perform important social functions, bridging gaps in regional infrastructures and fostering new forms of local culture. House of Natural Fiber (HONF), an arts

Figure 36.2 Waft Lab Tadarus Workshop poster. Caption: "While you're waiting for dawn [to break your fast], let's share and spread the spirit of Ramadan with Do It Yourself! Let's reflect on the importance of this process, of making electronic devices and continuing to refine them."

collective based in Yogyakarta, Indonesia, developed a project called *Intelligent Bacteria: Saccharomyces Cerevisiae* (2011) in which the group made wine out of Indonesian fruits and built circuits to amplify the tiny noises of the CO_2 bubbles created during fermentation. While *Intelligent Bacteria* was a successful art installation, winning the prestigious Transmediale Award the same year, it began as a research project to explore the social ramifications of a recent alcohol tax that had led to the making and selling of deadly methanol-containing homebrews (*anggur panganguran*, or "unemployment wine"). The installation came out of winemaking workshops in which the collective taught safe methods of home fermentation and methods of making affordable drinks with local fruit and generated discussions about the culture of alcohol in Muslim-majority Indonesia.

INTEGRATING REPAIR: KLUDGING, *NGOPREK*, *GAMBIARRA*

Workshops can naturally facilitate conversations about technology and its effects on society. For Daniel Llermaly, a Chilean musician and audio engineer who has been giving workshops under the moniker Oficina de Sonido since 2007, opening the black box reveals both the good and bad of technological systems:

> Everyone uses technology a lot, but nobody knows where it comes from and how the design process works. Now we have everybody wanting to learn to program and build electronics, which is good. But it's much more important to learn other things, like how technological systems are organized and the power structure [around them]. . . . Telephones don't just appear in your hand by a work of magic. On the other hand, we are all very angry when we learn about the environmental contamination of mining [for the materials to make phones]. How can we understand the relationship between our phones and that contamination?[5]

Organizers often stress the need to rethink the endless cycle of development and acquisition of new technologies and to develop more sustainable approaches to electronics, emphasizing reuse and repair. How can we use the materials we have at hand and e-waste that might be considered useless scrap? How do we shift the ethos of "makerspaces" from "make more" to "make do"? The skills of "making do" and "making fit" are not simply pragmatic; they can be seen more broadly as modeling a form of creativity that does not insist on unique, exceptionally new, never-before-existing creations. Art and technology are grounded in the basic affordances and techniques of daily living as people respond to technological displacement with the improvisational tinkering, kludging, and jerry-rigging basic to socioeconomic survival.

The very idea of "hacking" can represent totally different modes of agency and access in the "emergent markets" of the Global South. Media scholar Lilly Nguyen's 2016 essay on the circulation of iPhones in Vietnam, for instance, shows that while the idea of hacking in the North is often seen as a transgressive strategy for breaking out of corporate control, in emerging Asian economies it is seen more as a mode for gaining technical fluency in a global technoculture through intimate hands-on redesigns of its basic materials.[6] Given that repair is the basic starting point of innovation in circuit bending, a regional lack of access to the latest technologies can be counterbalanced by the ready availability of junk components. This approach to technology reflects how handmade electronics can embrace what Steven J. Jackson calls "broken world thinking" that reveals "what happens when we take erosion, breakdown, and decay, rather than novelty, growth, and progress, as our starting points" in thinking through technology.[7]

Peruvian artist Gabriel Castillo Agüero, for example, who grew up around markets full of reused and repaired electronics, describes his perspective on technology as less reliant on expertise and control than on an exploratory fascination with electronic sound and light.[8] In collectives such as Aloardi (Castillo Agüero, Christian Galarreta, and Janneke van der Putten), he uses inductors to induce noise and distortion into signals generated by modified musical greeting cards, builds his own electromagnetic microphones, and uses a discarded slide projector for interactive audiovisual performances (with Gisella López in Proyecto IRI).

Many handmade electronic instrument builders embrace the local lack of resources as aspects of content in their broader experiments. Javanese instrument builder Lintang Radittya created his Acak Baur (Chaos Box) to integrate the fluctuations of his rural power grid in Sewon Bantul, on the edge of Yogyakarta (Figure 36.3). The noisiness

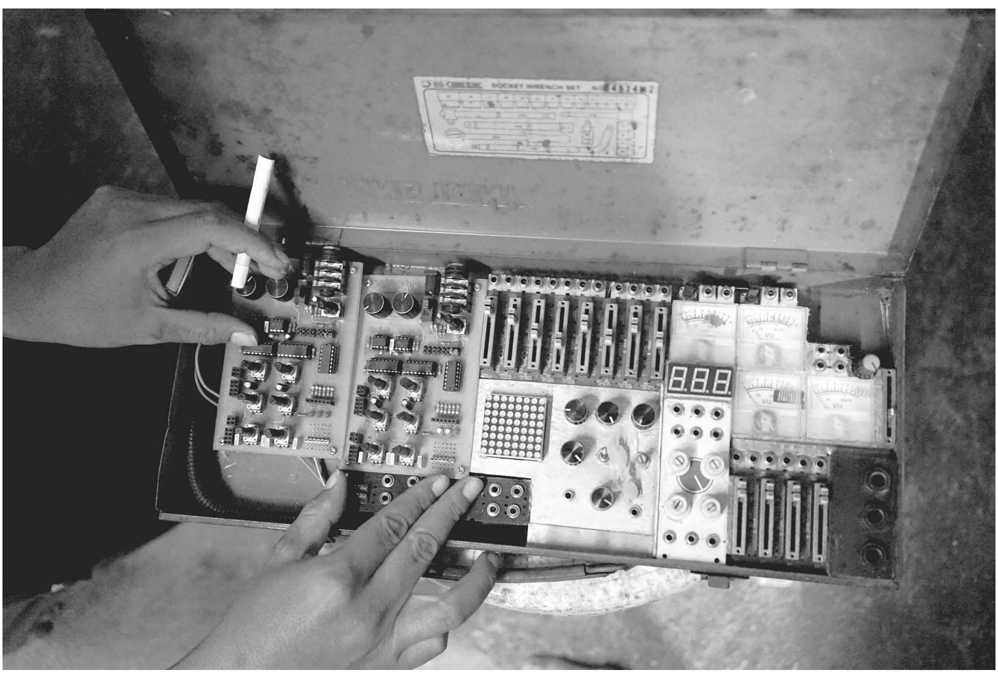


Figure 36.3 Lintang Radittya shows the chaotic functions of his Acak Baur module.

and unpredictability of the instrument work within an ecology of technological change that, he notes, is

> part of our lives in Indonesia. We live with unstable electricity. If you're working with your instrument in the midday or at night, sometimes it's very different. I've tested my hardware before in the midday and at night, and it was very different. When I use it at night, we get a very smooth humming, because here at night the voltage drops, because everyone is using electricity at the same time.[9]

Although this instability creates enormous problems, Radittya integrated its infrastructural flux into the design of his circuit; in Australia, however, he found he had to relearn how to use the instrument because the energy supply was too smooth. These improvisations reflect the Javanese sensibility of *ngoprek*, which Radittya describes as a logic of reusing whatever is available and just "dealing with what's around us." Even as the Internet has radically opened up the availability of information and cheap materials for electronic instrument building, the ethos of *ngoprek* continues to guide Indonesian hardware hackers. "These days," Radittya says, "it's easier to get parts in Indonesia to make whatever we want. We don't *have* to cannibalize, but we *like* to dig around in old technology and renew things."[10]

Argentinian artist and musician Jorge Crowe also chooses to work with unstable and unreliable media. For his installation *Modo Submarino*, Crowe records electromagnetic signals on a prepared cassette that has been unspooled, cut, eroded, and covered in dust. He conceives of the work as a dialogue between the artist and an unpredictable environment of electromagnetic sound:

> I like to think of it as machine/human communication. I ask, the machine answers and creates new questions. . . . There [wasn't] an intention to degrade [the cassette] in the first place. But I do like to create a field where unexpected things may happen. [*Modo Submarino*] had a coil that could make "visible" the electromagnetic field of our cellphones. When the emissions reached a certain threshold, the tape started recording, running and playing that sound. But the coil also picked the EM field of the artwork and it sounded like a strange sort of breathing. I love when these unexpected things happen.[11]

Similar concepts of improvised and experimental remakings of technologies are expressed by other groups, such as the Brazilian collective Gambiologia, founded in 2008 by Fred Paulino, Lucas Mafra, and Paulo Henrique "Ganso" Pessoa. Gambiologia can be seen as a reinvention of the Brazilian folk tradition of kludging (*gambiarra*). The name loosely translates as "the science of *gambiarra*," a concept in the spirit of *jetinho brasileiro*, the Brazilian way of finding creative solutions with limited resources and figuring out "life hacks" with a sense of mischief (potentially reclaiming negative connotations to the term, such as having little regard for the law). In pieces such as the playful *Gambiociclo* (Gambiocycle, a "multimedia vehicle equipped for the realization of electronic graffiti and other audiovisual interventions into public space"), the collective uses clever and chaotic assemblages of discarded objects to question the presumption of technological obsolescence (Figure 36.4).[12]

Figure 36.4 Gambiologia, *Gambiociclo*.

BENDING THE CIRCUITRY OF PARTICIPATION

New communities may be generated by those who don't fit in elsewhere, scrappy leftovers who cobble themselves together into new assemblages. Sometimes groups are formed in response to exclusionary configurations of music and tech scenes that produce uneven contexts of access (consider, for example, how the very names of online sound technology communities such as Gearslutz and MuffWiggler inhibit full participation and lead directly to harassment of female-identified members). Whether unconsciously or not, latent male homosociality can essentially create a social black box, concealing technological skills and knowledge of electronic sound-making from others. Some collectives address this with workshops specifically earmarked for groups who may not otherwise feel encouraged to experiment with electronics.

S1 Synth Library was founded by Felisha Ledesma and Alissa DeRubeis in Portland, Oregon, as a learning/work space in S1, a non-profit, artist-run gallery that includes performance, sound installations, visual galleries, and sound production classes. Since 2016, S1 has run volunteer-led workshops for learning about electronic music, as well as providing open shared access to gear, knowledge, and skills across the Portland sound community. The Synth Library was built with donated equipment solicited by DeRubeis from over 400 synth manufacturers, including 4MS, where she has worked as the communications and outreach director since 2014:

> I was going to NAMM and Moogfest and Superbooth and I talked to a lot of synth manufacturers and was telling them that I wanted to do something to promote

> more female participation, because frankly, I feel constantly surrounded by men and I don't like that. And everyone said, "We don't know why it's this way! We don't want it to be this way! We would love to give you synthesizers!" I wonder a lot about how it has ended up this way, but I'm not exactly sure myself, except that synthesizers exist in the world like everything else, and so they are also subject to the privilege of white men being able to access technology . . . [but] it gets worse with modular synths because they are expensive and shrouded in mystery.[13]

S1 began doing electronic music workshops in 2014, partly centered around the DJ collective Women's Beat League, founded by Ledesma, Daniela Serna, and Alyssa Beers as a space for female and non-binary people to learn DJing and sound production. DeRubeis added the Synth Library in 2016, offering classes in synthesis, patching, soldering, and kit building (Figure 36.5). The Synth Library holds open hours daily with facilitators available to answer questions. Addressing socioeconomic and physical obstacles to access, the organization offers a sliding scale for membership fees and accommodations for vision- and motor-impaired participants. In addition to teaching workshops for women at Moogfest and Superbooth, DeRubeis also collaborated with Czech artists Mary C and the Pink Noise Collective to open a sister library in Prague in 2018.[14]

Creating accessible spaces and recognizing women's contributions to sound technology has been a resonant theme in the rise of handmade electronic music communities around the world. Growing up in Belo Horizonte, Brazil, Vanessa De Michelis

Figure 36.5 S1 Synth Library workshop.

was initially excited by the possibilities of DIY collective organization through her involvement in punk scenes; the only problem was that she was interested in noise and didn't want to make punk music. Eventually, she explains, a variety of musicians joined together because they didn't play music in traditional ways, didn't fit genre expectations, or were female, lesbian, or queer.[15] After helping form the Azucrina collective, De Michelis grew critical of the noise community, which she saw as perpetuating the stereotypical link between technology and masculinity and emphasizing loudness and harshness in a way that reduced sound to a brutalist, implicitly masculine, aesthetic. De Michelis subverts expectations about noise artists and their work through her presentation as a non-binary person and in sonic choices that stress quieter sounds. As musicologist Tânia Mello Neiva notes, since De Michelis "does not fit into the normative standards of sexuality and gender identity, the subtle and delicate sonorities associated with the normative feminine are corrupted and questioned" by putting this work in the masculinized context of noise.[16]

De Michelis links her sound work to other forms of activism and community building, tailoring handmade audio workshops to females, lesbian women, LGBT communities, and youth from poor neighborhoods. In 2013, harking back to her days doing sound for punk venues, she taught a course in concert sound and stage assembly for women, noting that one of the barriers to entry for women in independent music was a lack of technical familiarity with performance. Constanza Piña, a Chilean musician who performs as Corazón de Robota with homemade and recycled circuits, connects the gendering of sound production to a social bias against women's work in general:

> Latin America, as the rest of the world, is quite *machista*. With my work I develop another idea of technology. Technology can be knitting, sewing, programming, cooking; the first manifestation of technology is fire, therefore, I avoid establishing hierarchies between technologies or diminishing people who work in technical labour. It is true. Most of the time I arrive to places where only men work. But my work is my activism, I am a woman working with technologies, teaching other women to work with electronics, to revalorize techniques, manual and craft practices, traditionally made by women. To me, there is no such thing as gender anymore. The word "robot" comes from the Czech word "robot", which means "forced labour" or the tedious work that the "man" doesn't want to do. For me, Corazón de Robota means tedious work made with love.[17]

ONGOING CHALLENGES

Today, makers of handmade electronic music are facing a number of systemic challenges from fast-changing social and economic circumstances. In many Western cities, the electronics districts and junk shops that are still ubiquitous in many Latin American and Asian cities have largely disappeared. The United States, the world's largest producer of e-waste, ships most of it to other countries, moving reusable parts out of artists' reach while reducing the visibility of its environmental impact. Once legendary for its electronics parts stores, New York City's Canal Street is now filled with knockoff purses and souvenirs; Radio Shack filed for bankruptcy and closed its retail stores in 2017, and shopping for electronic components has, for most Americans, moved online. Companies such as Adafruit, SparkFun, and Maker Shed (the online shopping division of *Make* magazine) sell pre-packaged kits that have begun to turn DIY electronics into a cottage industry of its own. The boom in modular synthesis is now represented by hundreds of small manufacturers introducing new modules for Eurorack systems on a weekly basis.

Meanwhile, many artist collectives are under threat in major cities, as rising rents make it difficult to maintain spaces dedicated to experimental arts safely and responsibly. The 2016 fire that killed 36 people at the Ghost Ship warehouse and live-work space in Oakland brutally highlighted the problem. In the weeks after the disaster, city fire inspectors shut down similar quasi-legal spaces around the country, and organizers canceled shows and scrambled to get their spaces up to code.[18] All-ages collective spaces are crucial, argued Ghost Ship resident Max Ohr, to allow for sound-making "on the fringes of music. [This] wasn't the kind of music that would make a lot of money for a bar. It was true self expression—people playing modular hardware, playing synthesizers, and really crafting it all themselves."[19]

In the United States, some collectives have begun reaching out to their local governments, explaining their value and creating new channels for developing public arts.[20] But this process can result in fundamental shifts in organizational structures and priorities, transforming loosely improvised groups into non-profit organizations that can receive grants and foundation support or into aspirationally self-sustaining businesses. The Santa Fe–based Meow Wolf has, since its founding in 2008, morphed from an art collective to a B Corporation (a classification that "balances purpose and profit") that bills itself as "an immersive experiences company," with branches in Denver and Las Vegas.[21] It currently employs over 400 artist-workers, some of whom filed suit in 2019, alleging unfair labor practices and gender discrimination.[22]

The changeable and ambiguous nature of organizations established to share technological access and information sometimes confuses the search for new social and cultural forms with more market-driven aspirations. Even the most experimental rationales for circuit bending can be folded into the rhetoric of "makerspaces," and the term "hack" can become lost in the mundane consumerist logic of "life hacks." In her work on hackathons and other makerspace events in India, Lilly Irani argues that these improvisational spaces can become a programmatic way for emergent middle classes "to remake culture by drawing legitimacy from the global prestige of technology industry work practices."[23] As Lauren Flood points out in her ethnography of makerspaces and electronic music hardware hacking in New York and Berlin, the "Maker Movement"

> purports to revolutionize creativity and the means of production by sharing knowledge about DIY projects . . . [merging] the sense of electronics hobbyism and the idea of being an amateur inventor with the fast-paced business savvy of twenty-first-century technology start-up companies. As a result, it embodies numerous tensions between DIY as an *alternative* to or an *escape* from commercialism with the call to profit-driven entrepreneurship.[24]

Even events as low to the ground as circuit-bending workshops are not immune to these effects. While makerspaces can provide one-off initial exposures to electronics, they are often absorbed into a STEAM (science, technology, engineering, arts, and mathematics) curriculum that provides few opportunities to build a long-term educational structure.

In validating the social and educational productivity of DIY entrepreneurialism, hacking workshops also regularly downplay aesthetics in favor of the techy start-up gleam associated with exposing beginners to electronic circuitry. Artist and engineer Andreas Siagian lamented that after years of conducting introductory circuit-building workshops: "Even though I'm always happy to see people's responses when they join the workshop and see the results, I get tired of hearing the same sounds over and over again . . . it's like reinventing the wheel every time."[25]

Collectives also face the functional drawbacks of allocating funding and credit among members, as when debates about who should be recognized as the primary author of an artwork split the House of Natural Fiber in Yogyakarta, with several members departing and reforming as Lifepatch.[26] Further, the recognition of global collectives, while highly productive (i.e., the 2019 selection of Jakarta-based collective ruangrupa as the first Asian curators of the prestigious Documenta contemporary art festival in Kassel, Germany) can also result in a politically reductive reception in which artists from the Global South are viewed primarily by the international art world through the lens of regional culture. The very aspect that makes the do-it-together model so appealing, then—its ideological refusal to highlight any particular individual or art product—also generates substantial interpersonal, socioeconomic, and transcultural challenges.

Despite these complexities, many artists remain committed to handmade electronic music and to collaborative forms of organizing. Maintaining spaces for performance and exchange is essential for these communities as they bring techniques from the world of technological experimentation into the sites of new social movements. When black boxes are opened and people peer in . . . when people gather to change sound and listen to sound change together . . . when electronic circuits flow into open environments that combine learning with mystery . . . when the vitality of noise overwhelms, rather than reinforces, the norms of social identity . . . when cycles of technological creativity feed back into process instead of hardening into product . . . when a collective holds open a space in a modern city that provides respite from the hegemony of private property . . . then we can hear it again—the sound of the circuit, the sound that communities make when they embody the spirit of hacking, breaking it down and starting again from scratch, to bring people into new worlds of sonic and social possibility.

NOTES

1. Vanessa De Michelis, Personal Communication with Lisa Kori, email, 27 March 2019.
2. Trevor Pinch, "'Bring on Sector Two!' The Sounds of Bent and Broken Circuits." *Sound Studies* Vol. 2(1) (2016). Pp. 36–51.

3. Nozu, Kanami. 1998. Interview by Shinsuke Inoue. *ReSist* 1(spring). Cited in David Novak, *Japanoise: Music at the Edge of Circulation*, p. 165. Duke University Press, 2013.
4. Waft Lab website. Accessed 4 June 2019, www.waft-lab.com/electrowork.html
5. Daniel Llermaly, Skype Interview with Lisa Kori. 13 May 2019.
6. Nguyen, Lilly. 2016. "Infrastructural action in Vietnam: Inverting the techno-politics of hacking in the global South." *New Media & Society* Vol. 18(4) (2016). Pp. 637–652.
7. Steven J. Jackson, "Rethinking Repair." In *Media Technologies: Essays in Communication, Materiality and Society* (eds. T. Gillespie, P. Boczkowski and K. Foot), pp. 221–240. Cambridge: MIT Press, 2014.
8. Gabriel Castillo Agüero, Personal Communication with Lisa Kori, email, 26 March 2019.
9. Lintang Radittya, "Javanoise: An Interview with Lintang Raditya by Danny Martin." *InSituMag*, 1, 2017. Accessed 5 June 2019, http://insiturec.com/mag/issue-1/java-noise/
10. Lintang Radittya, Interview with David Novak. 16 September 2018, Yogyakarta.
11. Jorge Crowe, Personal Communication with Lisa Kori, email, 3 June 2019.
12. Gambiologia website, www.gambiologia.net/blog/portfolio/gambiocycle.
13. Alissa DeRubeis, Interview with David Novak. 30 July 2018. Portland, OR.
14. Madeleine Campbell, "S1 Synth Library." *Women in Sound*, 4, 2017. Accessed 7 June 2019, www.womeninsound.com/issue-4/s1-synth-library
15. Vanessa De Michelis, Skype Interview with Lisa Kori. 9 May 2019.
16. Tânia Mello Neiva, "Inspiring Sonorities: Politics and Gender Issues in the Work of Brazilian Sound Artists." In *Making it Heard: A History of Brazilian Sound Art* (eds. Rui Chaves and Fernando Iazzetta). Bloomsbury, 2019.
17. Peter Gonda, "Against Techno Patriarchy." *norient.com*, 17 December 2017. Accessed 20 May 2019, https://norient.com/stories/constanza-pina/
18. Closures of quasi-legal spaces were urged on by "alt-right" leaning hacktivists on the social media site 4chan, where members listed addresses for DIY venues that they claimed were "open hotbeds of liberal radicalism." One group called the Right Wing Safety Squad claimed to have contributed to shutting down 16 art spaces around the country, including Rhinoceropolis in Denver, Bell Foundry in Baltimore, and Glass Ménage in Nashville. See Aaron Davis, "Online Spaces Become Home to Battle Over DIY Spaces Around Country." *East Bay Times*, 24 December 2016.
19. Gabrielle Canon, "After the Fire, an Uncertain Future for Artists in Oakland." *Vice* website, 2016, https://thump.vice.com/en_us/article/mgwxgx/oakland-ghost-ship-warehouse-fire-feature-interview-100-silk
20. Caroline Lewis. Skype Interview with Lisa Kori, 9 May 2019.
21. Meow Wolf website. Accessed 25 November 2019, https://meowwolf.com/about/community/diy
22. "Meow Wolf Is Being Sued by Former Employees for Unfair Labor Practices." *Hyperallergic* website. Accessed 25 November 2019, https://hyperallergic.com/508005/meow-wolf-former-employees/
23. Lilly Irani, "Hackathons and the Making of Entrepreneurial Citizenship." *Science, Technology and Human Values* Vol. 40(5) (2015). Pp. 1–26.
24. Lauren Flood, "Building and Becoming: DIY Music Technology in New York and Berlin." PhD dissertation, Columbia University, 2016: 19, 20, emphasis in original.
25. Andreas Siagian. Interview with David Novak. 17 September 2018, Yogyakarta.
26. Lifepatch website, https://lifepatch.org/Category:Workshop

CHAPTER 37

Hello World!

In the 15 years since I stapled together the crude PDF classroom manual that eventually became this book, I've presented some 200 hacking workshops on six continents (one workshop, in Punta Arenas, got within 1,000 miles of the seventh continent).[1] Subsequent to the publication of the first edition in 2006, I was initially puzzled that anyone would want to attend a group workshop, much less produce one: why not just buy or borrow the book and follow the acronym to DIY? I came to realize, however, that hacking had a social side: it's easier to fix a problem or answer a question when you're immersed in a group of people who are also trying to answer that question and solve that problem. As Lisa Kori and David Novak articulate in the previous chapter, hacking is not just about the technology, it is "a social and sonic practice of community building, in which people 'do it *together*.'" And whereas the majority of electronic technology available in Indonesia is no different than that in Indiana or Israel or India, the music and art produced with it can vary greatly from country to country. I concluded the previous edition of this book with a chapter that attempted to provide an overview of the international hacking scene as a proliferation of technological concerns, unwittingly imposing a slightly naïve "It's a Small World After All" approach that stressed similarities in the work of a diverse group of artists. That chapter, I now realize, needs corrective supplementation. For this update, I called upon a dozen hackers and writers from around the world to contribute on-the-ground analyses of their scenes and concerns.

A HISTORY OF JAPANESE HACKING AND DIY MUSIC (ADACHI TOMOMI) (FULL CHAPTER ON WEBSITE)

"Is it unreasonable to start a history of Japanese hacking and DIY music with the year 1924?" asks ADACHI Tomomi at the start of his survey of how electronics, contingent instrument building, experimentation, and improvisation combined in recent Japanese art and music. While giving renowned composers such

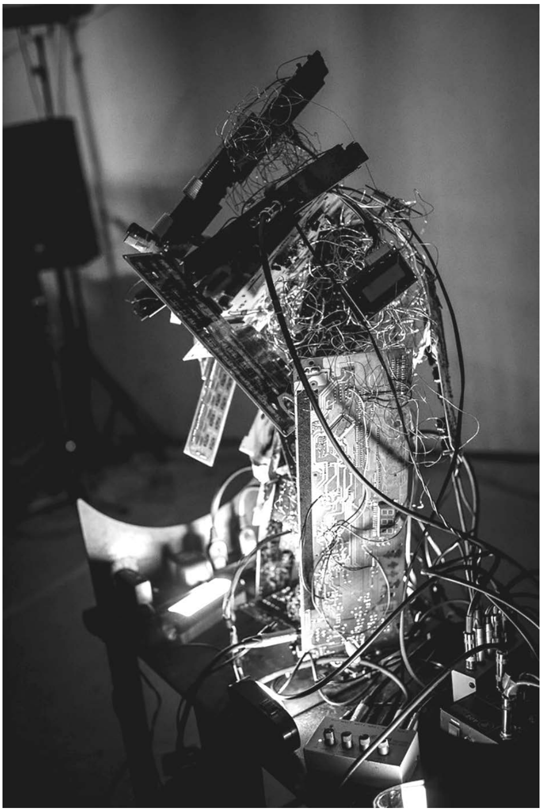


Figure 37.1
Kayu Nakada, Bug Synth.

as Takehisha Kosugi and Yasunao Tone their due, the essay also acknowledges the engineers whose influence on musical developments worldwide is often overlooked and charts relationships between makers, performers, composers, and artists of multiple generations. ADACHI underlines Japan's connections with international movements and—as important—the impact of local phenomena, not least the Akihabara district of Tokyo and its role as a haven of electronic surplus and inspiration (Figure 37.1).

LIVENING THINGS UP: AUSTRALIAN HAND-BUILT ELECTRONIC INSTRUMENTS (CALEB KELLY AND PIA VAN GELDER) (FULL CHAPTER ON WEBSITE)

A "larrikin" in Australia is someone who does not observe convention and plays or behaves in a mischievous way; it also offers a useful way of characterizing the insouciant, irreverent flavor of electronic instrument building and performance in Australia over the past few decades. Caleb Kelly and Pia van Gelder survey a range of artists, composers, bands, and collectives who tinker with found materials

Figure 37.2 Vincent O'Connor's *Millionth Acre* (2015), detail of recording device, Millionth Acre, New South Wales.

and hack toys, microwave ovens, automobiles, bandsaws, breadboards, e-waste, and volatile gases. "Some," the authors note, "are more serious about it than others" (Figure 37.2).

GAMBIOLUTHIERY: HACKING AND DIY IN BRAZIL (GIULIANO OBICI) (FULL CHAPTER ON WEBSITE)

The popular Brazilian expression *gambiarra* describes an improvised, informal way of solving an everyday problem when needed tools or resources are not available. Like "hacking" and DIY, it reflects a way of dealing with the objects and issues that occupy the daily life of post-industrial societies and can be charted through the attitudes, ambitions, and art of musicians, composers, and instrument builders. Artist and researcher Obici has coined the term "gambioluthiery" for the construction of instruments oriented around the logic of *gambiarra*, involving activities such as composing, decomposing, inventing, adapting, and appropriating materials, artifacts, or system setups. This essay offers a panorama of the Brazilian scene, viewed through the lens of gambioluthiery (Figure 37.3).

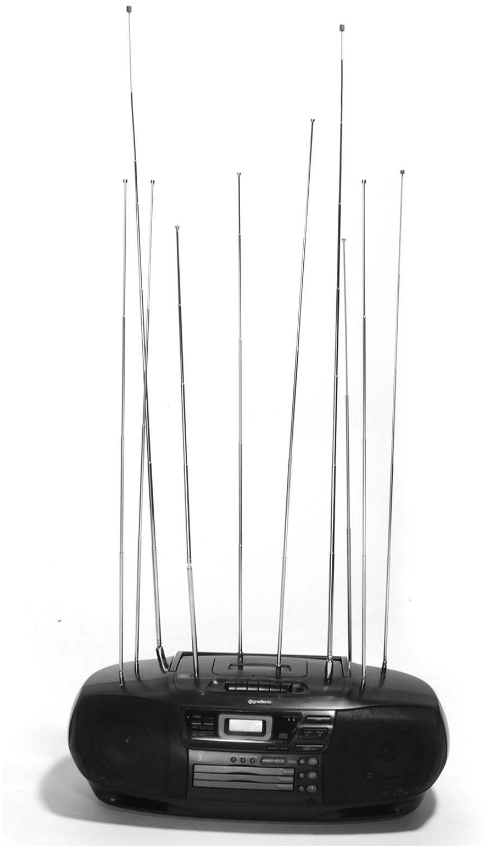


Figure 37.3
Vivian Caccuri, *Adeus* (2004), mini sound system with microprocessor devices, FM receivers, and antennas.

DO IT WITH OTHERS: HARDWARE HACKING IN SOUTH AMERICA (FLORENCIA CURCI, ALMA LAPRIDA, AND SEBASTIÁN REY) (FULL CHAPTER ON WEBSITE)

COMPANION @ WEBSITE

The hacking virus has been spreading through Latin America for several years now, fostering crossovers between disciplines and instigating various kinds of information sharing. Many readers understand "hardware hacking" as the act of inventing alternative uses for existing devices, but for Argentinians Florencia Curci, Alma Laprida, and Sebastián Rey, it also means a collective attitude that expands to create social and artistic networks. Reaching out to self-identified hackers in Argentina, Bolivia, Brazil, Chile, Colombia, Ecuador, and Peru, they gathered information about local scenes and the myriad ways in which invention, reuse, and improvisatory problem solving play out in art and social practice. Particular attention is paid to groups in which hacking engages with growing feminist movements in South America (Figure 37.4).

Figure 37.4 Students at a hardware hacking workshop in Puerto Libertad, Misiones, Argentina.

A BRIEF PERSONAL HISTORY OF DORKBOT-NYC (DOUGLAS REPETTO) (FULL CHAPTER ON WEBSITE)

Few things testify more eloquently to the attractions of the eccentric, improvisational ethos of hacking than the flourishing of dorkbot events in the early 2000s. Begun by artist Douglas Repetto as a monthly meeting for "people doing strange things with electricity," dorkbot eventually spread to 80 locations, supporting and spawning nascent artist and musician communities around the globe. Repetto recounts dorkbot's aims and trajectory from its origin: "I knew hardly anyone in New York, but I knew the place was full of weirdos, and I wanted to know them. I started a series of informal meetings called dorkbot-nyc, with the idea that people from around the city could share their experiments with one another." The rest is history (Figure 37.5).

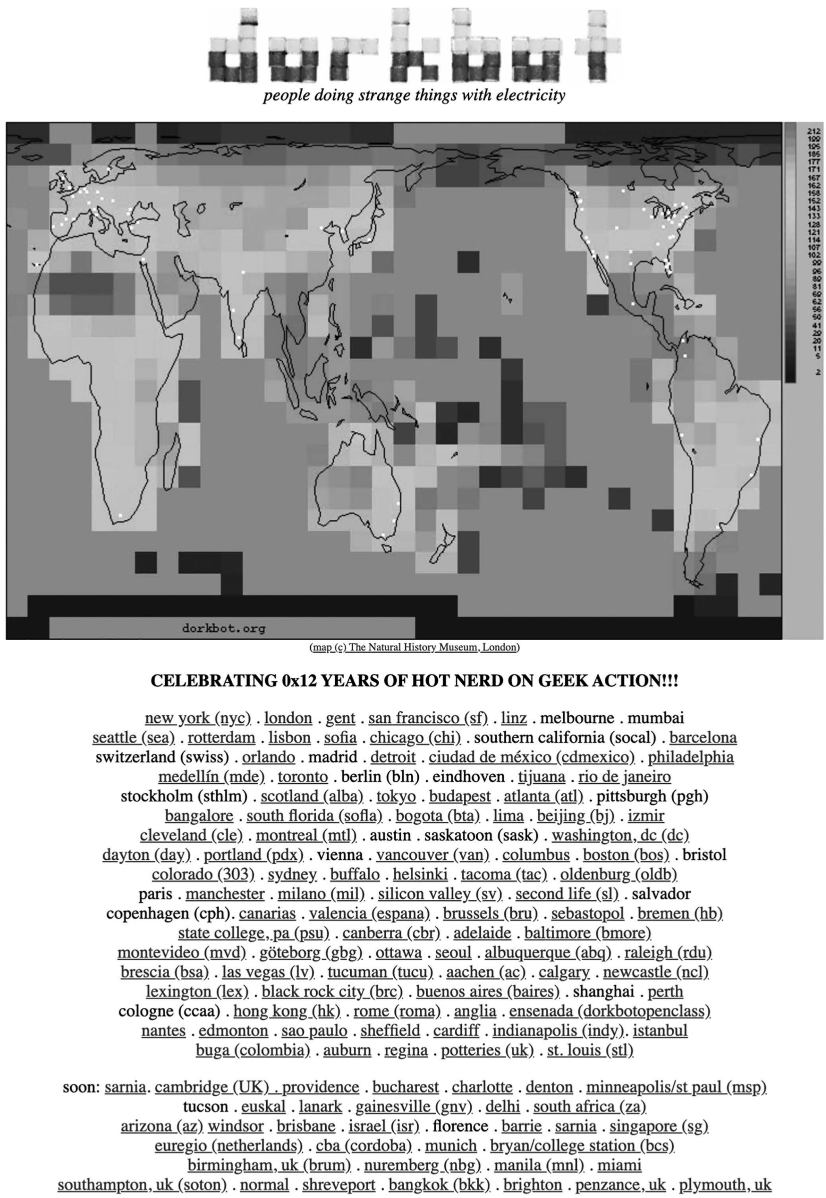


Figure 37.5 dorkbot.org homepage.

BLEEP LISTENING (EZRA TEBOUL) (FULL CHAPTER ON WEBSITE)

Handmade electronic music—the "music *implicit* in technology"—requires attention to many things, musicologist Ezra Teboul observes: from the electrical properties of materials to the sonic consequences of those materials' evolving interconnections to the creation of situations favorable to the exploration of those evolutions. Conditions specific to each project—available parts, information, ideas, individuals—shape the final results. Studying the practice means considering both the assembled and the assembler: who and what share responsibility for the resulting ever-changing communities and artifacts? In this essay Teboul discusses a wide variety of recent projects by artists and musicians and considers how they address chaos, control, and the creation of bespoke culture from technological jetsam (Figure 37.6).

Figure 37.6 Ragnhild May, *The Flute Player.*

THE CONTACT MICROPHONE: A CULTURAL OBJECT (DANIELA FANTECHI) (FULL CHAPTER ON WEBSITE)

Daniela Fantechi observes the last half century of experimental music by charting the sonic and conceptual impact of the contact microphone. Cheap and simple, the contact microphone has inspired generations of composers and artists to listen to the world—or at least the amplified objects therein—in new ways. Bypassing synthesis, it facilitates the use of "found sound" as a working material and, like the saxophone and the electric guitar, is both instrument and invitation to rethink what music might be. It is also, Fantechi shows, a lens through which to view the history of electronic music beginning with John Cage's 1960 *Cartridge Music* and David Tudor's groundbreaking permutational *Rainforest*. Its possibilities dislodged mid-century European composition from its commitment to tape music and sparked an intercontinental fascination with live electronic music, unconventional instrument building, and improvisation that continues to be an active field of play (Figure 37.7).

Figure 37.7 *Shozyg I* (1968), self-built electro-acoustic musical instrument by Hugh Davies.
Photo credit: © Science Museum/Science & Society Picture Library.

We began this book with speakers and batteries linked with clip leads, contact mikes clamped to springs, and fingers resting on the guts of old radios. From there we hacked our way through a jungle of electronic components before emerging into the world of computers—a more familiar environment for most readers, even if we struck out on new paths. There was a time when all these materials were identified with "advanced industrialized nations," such as the United States, and usually with the dreaded military industrial complex: in the era of Reagan's proxy wars in Central America, I remember cringing at the "made in El Salvador" stamp on the chips I bought. Now this technology is globally ubiquitous and more likely to be found in a Prius than a missile. And yet, as these contributors have demonstrated so ably, not everyone is building Priuses. Some hackers are, as David Behrman hoped (echoing Antonin Artaud), inventing new and noisy creatures with three arms and three legs.

NOTE

1. www.nicolascollins.com/texts/originalhackingmanual.pdf

APPENDIX A

Resources

THE WEB

Sometime around 2002 I walked into the office of a technically minded colleague at my school to ask for a reference manual in which I could look up the pinout and schematic of an unfamiliar chip. Clamping a large, Chicago-style hand to my shoulder, Ed replied, "Nic, I *could* loan you the book, but let me ask you this: give a man a fish and he's fed for one day, teach him how to fish and he. . .?" ". . . Wastes all his time fishing when he should be helping out around the house?" I continued. "No," sighed a disappointed Ed. "Type the part number into Google and you will find the data sheet in the first hit," he muttered as he closed the door. The point I missed in his parable: it's never been easier to hack. In the early days of homemade electronic music, schematics and suggestions were exchanged by word of mouth and sleight of hand, like cures for colicky babies. Then a few dumbed-down circuits crawled out of engineering journals into magazines for electronic hobbyists and aspiring electric guitarists; one or two books appeared, written in something vaguely like English rather than Technese. Finally, Tim Berners-Lee birthed the World Wide Web, and a hundred fuzztones flowered.

Anything you want to know is out there; all you need to do is find it and understand it. Finding it is easy—understanding it may take some work. You will have to teach yourself a bit more of the vocabulary of electronics than was demanded by this book—you'll need to start reading schematics. As Ed suggested, typing a part number or name of a component will usually retrieve a PDF of a manufacturer's data sheet—here you'll find all the basic information you'll need to start working with it: what pins are connected to power, which are inputs, which are outputs, etc. Enter a descriptive phrase instead ("'Phase Shifter'+schematic") and you'll be directed to any number of wacky websites hosted by people who seem to have nothing better to do than compile vast collections of circuit diagrams and provide links to like-minded fanatics. All you need to do is figure out how to translate the schematic onto the breadboard. A little trial and error, persistence, patience, and an occasional glance back at this book should get you there.

BOOKS

PDF data sheets can be downloaded as you need them; thick data books are still available from the major chip manufacturer, but they're expensive and usually don't justify the shelf space—"Application Notes" for specific chips are worth the hunt, though.

There are a several books that can help fill in theoretical gaps between this handbook and the more engineer-oriented data you will find on the Web or in the data books. Some date from the 1970s but are still relevant. Don Lancaster's *CMOS Cookbook* (Indianapolis: SAMS Publications, 1977) provides a thorough introduction to the chip family we have misused throughout my book. Walter Jung's *OpAmp Cookbook* (Indianapolis: SAMS Publications, 1974) will introduce you to the component out of which most audio circuits are built. Craig Anderton, the grandfather of electronic hacking for musicians, published *Electronic Projects for Musicians* back in 1975 (New York: Amsco Publications), and it's still an excellent guide to basic musical circuits and general principles of design and construction, written in large, reassuring, musician-friendly letters. For super low-tech, "foxhole technology," *Sneaky Uses for Everyday Things* by Cy Tymony (Kansas City: Andrews McMeel Publishing, 2003) is a wonderful source of circuit designs built from little more than stationary supplies, salt, spare change, and wet paper towel. Reed Ghazala, the patron saint of circuit bending, put out a book that is an excellent companion to the one you are reading now: Reed Ghazala, *Circuit Bending: Build Your Own Alien Instruments* (New York: Wiley Publications, 2005). Brian Wampler's *How to Modify Effect Pedals for Guitar and Bass* (CreateSpace, 2007) provides detailed instructions on hacks to numerous stomp boxes.

An excellent guide to amplifying *any* instrument (on a small budget) is Bart Hopkin's *Getting a Bigger Sound: Pickups and Microphones for Your Musical Instrument* (Tucson, AZ: Sharp Press, 2003). Hopkin covers making and using coil pickups, contact mikes, and air mikes and includes a neat trick for rewiring electret elements to improve their ability to record very loud sounds without distortion.

STUFF

What holds true for information also goes for material resources. Although the Big Apple's Canal Street no longer teems with the warrens of weird electronic and mechanical surplus shops that enthralled me throughout my childhood, the Web has become a virtual medina of the misplaced and unwanted. Add "+price" to the search field after anything you desire—plugs, piezo disks, tape heads, tilt switches—and you'll soon find a place to buy it. Since you're not manufacturing missiles or airbags in quantity, you'll need to find a source that will sell to the common man or woman. For ICs, resistors, capacitors, and other small components, a straightforward electronic retailer is probably the best bet. As of the time of writing, some good sources that stock a wide range of parts include Digi-Key (www.digikey.com), Jameco (www.jameco.com), and the delightfully named Mouser Electronics (www.mouser.com). For microcontrollers like the Arduino (Chapter 32) and Raspberry Pi (Chapter 33), as well as all manner of sensors, Adafruit (www.adafruit.com) and SparkFun (www.sparkfun.com) are great sources; their websites are also loaded with helpful tutorials.

For soft circuitry, these are some useful sources:

www.lessemf.com/ (lots of conductive fabrics and conductive and resistive thread, inks, and paints)
www.shopvtechtextiles.com/ (go-to for industry-grade conductive materials)
www.lustersheen.com/ (bronze metallic wools)
sewiy.com/ (solderable conductive thread)
www.kitronik.co.uk/e-textiles-conductive-thread.html
Your local hardware store is good for steel metallic wool for spinning and carding (be careful: pure steel wool ignites with low power!).

For hackable gizmos, used equipment, pots, jacks, boxes, and general inspiration, however, you should try the "surplus" outlets. Here are a few reputable sources that have been around for a while selling cool stuff:

All Electronics: www.allelectronics.com Marlin P. Jones: www.mpja.com Electronic Goldmine: www.goldmine-elec.com
B.G. Micro: www.bgmicro.com
Surplus Shed: www.surplusshed.com
Unicorn Electronics: www.unicornelex.com

Kits are a great way to ease the transition from circuit bending to free-form electronic design. The project-specific printed circuit boards minimize the chances for mis-wiring, and they often come with suggestions for variations. Many of the prior vendors carry a range of kits.

The Web knows no national boundaries, but sadly the US Post Office does. The aforementioned US sources will charge a premium for shipping abroad. In England, Maplin (www.maplin.co.uk) carries a wide range of components. In Germany, Conrad Electronics is a good bet (and they have retails shops in Berlin and several other cities): www.conrad.de. RS Components is a full range dealer of new parts that delivers across Europe: http://rswww.com. More and more components are coming from China. Many large Chinese vendors have English-language websites, and their international shipping costs are surprisingly modest and not as slow as you would think; many of them also maintain listings on eBay in various countries.

For better or for worse, Radio Shack—overpriced but for many years the American street corner source for last-minute jacks and switches—is gone.

The authentic, old-fashioned "electronic junkyard" is not entirely extinct. Apex Surplus (at 8909 San Fernando Road in Sun Valley, California) is packed with decades of electronic technology and just plain weird stuff. The Web address is www.apexelectronic.com, but this is no time for a virtual experience: having read this book, you owe it to yourself to visit Apex once before you (or it) die, even if you never get to Mecca, Jerusalem, or St. Peters. Weird Stuff Warehouse (www.weirdstuff.com) in Sunnyvale, California, is the elephant graveyard of obsolete computers and computer accessories—just the place to pick up a fully functional Fat Mac. American Science and Surplus in Chicago (www.sciplus.com) has provided my students with a wide range of excellent materials. The Ax-Man (www.ax-man.com), with several locations in the Minneapolis/St. Paul area, includes a fair amount of electronic parts among a wide range of general surplus material (such as East German crossing guard blinking braces). And in the Netherlands, Twente Electronics is well worth the trip to The Hague. There are many other wonderful stores around the world, I've been told, but these are ones I have visited and can personally vouch for.

APPENDIX B

The Rules of Hacking

Rule #1: Fear not (Chapter 2)!

Rule #2: Don't take apart anything that plugs directly into the wall (Chapter 2).

Rule #3: It is easier to take something apart than put it back together (Chapter 2).

Rule #4: Make notes of what you are doing as you go along, not after (Chapter 2).

Rule #5: Avoid connecting the battery backwards (Chapter 2).

Rule #6: Many hacks are like butterflies: beautiful but short-lived (Chapter 2).

Rule #7: In general try to avoid short circuits (Chapter 2).

Rule #8: In electronics some things are reversible with interesting results, but some things are reversible only with irreversible results (Chapter 4).

Rule #9: Use shielded cable to make all audio connections longer than your hand is wide, unless they go between an amplifier and a speaker (Chapter 7).

Rule #10: Every audio connection consists of two parts: the signal and a ground reference (Chapter 7).

Rule #11: Don't drink and solder (Chapter 7).

Rule #12: After a hacked circuit crashes, you may need to disconnect and reconnect the batteries before it will run again (WEBSITE: Hack the Clock: Changing the Clock Speed for Cool New Sounds).

Rule #13: The net value of two resistors connected in parallel is a little bit less than the smaller of the two resistors; the net value of two resistors connected in series is the sum of the two resistors (WEBSITE: Ohm's Law for Dummies).

Rule #14: Kick me off if I stick (Zummo's rule) (WEBSITE: Jack, Batt, and Pack: Powering and Packaging Your Hacked Toy).

Rule #15: You can always substitute a larger 1.5-volt battery for a smaller one, just make sure you use the same number of batteries, in the same configuration (WEBSITE: Jack, Batt, and Pack: Powering and Packaging Your Hacked Toy).

Rule #16: It's always safer to use separate batteries for separate circuits (WEBSITE: Jack, Batt, and Pack: Powering and Packaging Your Hacked Toy).

Rule #17: Start simple and confirm that the circuit still works after every addition you make (Chapter 13).

Rule #18: If it sounds good and doesn't smoke, don't worry if you don't understand it (Chapter 13).

Rule #19: Always leave your original breadboard design intact and functional until you can prove that the soldered-up version works (Chapter 14).

Rule #20: All chips may look alike on the outside without being the same on the inside—read the fine print (Chapter 15)!

Rule #21: All chips expect "+" and "−" power connections to their designated power supply pins, even if these voltages are also connected to other pins for other reasons—withhold them at your own risk (or entertainment; Chapter 15).

Rule #22: Always use a resistor when powering an LED, otherwise the circuit and/or LED might blow out (Chapter 17).

Rule #23: Distortion is Truth (Poss's law, Chapter 19).

Rule #24: It is easier to drill round holes than slots (Chapter 18).

Rule #25: Never trust the writing on the wall wart (WEBSITE: Power Supplies: Carbon Footprints from AA to EEE).

THE LAWS OF THE AVANT-GARDE

Law #1: Do it backwards (Chapter 4).

Law #2: Make it louder, a lot (Chapter 7).

Law #3: Slow it down, a lot (WEBSITE: Hack the Clock: Changing the Clock Speed for Cool New Sounds).

Contributing Authors

Jess Rowland is a sound artist, musician, and composer based in New York City.

www.jessrowland.com

Lara Grant is a designer, educator, fabricator, and researcher with focuses in community building, sustainability, and accessibility. She builds experimental and practical interfaces that primarily involve wearable and e-textile techniques and materials.

Sarah Grant is an American media artist, educator, and software developer with a focus on radio art and computer networking. She organizes the Radical Networks conference in New York and Berlin, a community event for social justice, critical investigations, and creative experiments in telecommunications.

Alex Inglizian is a Chicago-based artist, composer, musician, engineer, and educator. He is a graduate of the School of The Art Institute of Chicago and is currently the technical director and chief recording engineer at Experimental Sound Studio, an internationally recognized non-profit sound art organization.

LoVid, the NY-based artist duo comprised of **Tali Hinkis** and **Kyle Lapidus** produces immersive installations, sculptural analog A/V synthesizers, single channel videos, textile, participatory projects, mobile media cinema, works on paper, and multimedia performance. These projects juxtapose media and material, incorporating technology to observe relationships between the human body, the natural world, and artificial systems; highlight challenges/possibilities of the networked age; and explore how contemporary culture seeps into our collective consciousness to shape perception.

www.lovid.org

Jon Satrom is a Chicago-based artist, educator, and organizer who specializes in creative problem creating and performs real-time audio/video new-media/noise.

http://jonsatrom.com

Michael Johnsen (circuit designer, performer, researcher, Pittsburgh) is known for live electronic performances with a menagerie of custom devices whose social interactions produce teeming chirps, sudden transients, and charming failure modes; embracing the dirt in pure electronics. His work is characterized by a relative lack of ideas per se and an intense focus on observation, the way a shepherd watches sheep.

You Nakai makes music(ians), dance(rs), haunted musical houses, and other kinds of work as a member of No Collective (nocollective.com) and Already Not Yet (alreadynotyet.org). His current concerns include teaching a course on fake music history at the Kyoto City University of Arts, organizing Unplugged Silicons, a chorus group dedicated to reenacting the classics of electronic music in a cappella, and finishing a book titled *Reminded by the Instruments: David Tudor's Music* to be released from Oxford University Press in 2020.

Peter Blasser, designer and builder of synthesizers, practices deep consultation with clients who seek rich interactions with their electronics. The instruments manifest electronic modulations through pins, case flexure, and radio fields.

http://ciat-lonbarde.net

Brett Balogh is a Chicago-based artist, designer, and tool-maker making aural, sculptural, and cartographic explorations of electromagnetic space. Brett teaches courses in bio-art, electronics, creative coding, and robotics in the Art and Technology Studies program at the School of the Art Institute of Chicago.

https://brettbalogh.com

Holger Heckeroth became obsessed with synthesizers at the age of six when his teacher brought one to school; at the age of eight, he was introduced to computers and electronics by his older brother, and this was the starting point of his passion for technology, noise, and DIY. After a university diploma in economics, and a detour into the fashion industry, Holger went back to his roots and is now the CEO of an IT service company devoting his spare time to engineering and building synthesizers and noisemakers of all kinds.

Peter Speer is an artist, musician, and dad based out of Asheville, NC.

www.diode-ring.com

Wolfgang Spahn is an Austrian artist based in Berlin who specializes in installations and performances of light and sound, for which he develops diverse analog and digital electronic systems. Spahn is faculty member of the University of the Arts, Sound Studies and Sonic Arts, Berlin.

http://wolfgang-spahn.de/

Eduardo F. Rosario (b. 1988) is an experimental musician and sound artist from Caguas, Puerto Rico.

www.eduardofrosario.com

Joseph Kramer uses technology to create instruments and sound works, most notably as half of the long-running duo Coppice (with Noé Cuéllar). He lives and teaches in Chicago, IL.

https://josephmkramer.com/

Robb Drinkwater is an artist, educator, creator, and crafter whose work ranges from silly to scientific with tools ranging from clothespins to computers. More can be found about him at www.splintergroup.com, or follow him on social media.

Nick Briz is an internationally recognized new media artist, educator, and organizer who makes work with, but also about, the technology that has come to define the era we're living in. He is an active participant in various online communities and conversations including glitch art, net art, remix culture, digital rights, Internet ecology, and digital literacy and an adjunct associate professor at the School of the Art Institute of Chicago, co-founder of netizen.org, and the creative director of the digital agency Branger_Briz.

Lisa Kori is an artist and musician based in New Mexico. Her 2011 research project, The Medium and the Mayhem, documented underground electronic arts around the globe.

http://lisakori.net

David Novak is Associate Professor at the University of California Santa Barbara and Director of the Center for the Interdisciplinary Study of Music. He is the author of *Japanoise: Music at the Edge of Circulation* (Duke, 2013, www.japanoise.com) and co-editor of *Keywords in Sound* (Duke, 2015).

www.japanoise.com/

https://keywordsinsound.com/

Florencia Curci is an artist and curator. Her research involves noise, rhythm, and voices as connective forces in order to build affective communities. She created CASo (Sound Art Center) in Buenos Aires, Argentina.

www.florenciacurci.com

Alma Laprida is a sound artist and educator. She took her first hacking workshops with Jorge Crowe in 2009 and then studied electronic arts. She composes and plays pieces for tromba marina, synthesizers, lyre, and other non-conventional instruments and objects.

https://almalaprida.wordpress.com/

Sebastián Rey is an artist, teacher, and music therapist. In 2010 together with Leonello Zambón they created the sound research laboratory SOUND C!NICO, where they carry out hacking meetings and sound activism for Latin America.

www.sonidocinico.wordpress.com

Ezra Teboul is an artist and researcher documenting the music implicit in electronics. Particularly interested in the political and cultural meaning of technical decisions, his work adapts reverse engineering to the elucidation of mechanisms that connect material conditions to personal ideas and ideals, starting in sound and branching out to everyday life.

https://redthunderaudio.com/

ADACHI Tomomi is a performer/composer, instrument builder, sound poet, and visual artist. Known for his versatile style, from solo performance with voice and self-made electronics, to site-specific large ensemble compositions, to 3D texts.

www.adachitomomi.com

Caleb Kelly is an associate professor in media art and art theory at UNSW Art & Design in Sydney, where he co-directs the research group Sound, Energies, and Environments. He is the author of *Cracked Media* (MIT Press) and most recently *Gallery Sound* (Bloomsbury Academic).

http://see.unsw.edu.au

Pia van Gelder is an artist, historian, and researcher at the Australian National University. Her work explores the influence of cultural conceptions of energy, particularly esotericism, on electronic instruments in the arts of the twentieth century.

piavangelder.com

Giuliano Obici is an artist/researcher with background in music, communication, and psychology.

giulianobici.com

Douglas Repetto is an artist and teacher; he has been involved in many art/technology/research projects and communities over the years, including Sound Arts and the CMC at Columbia University, ArtBots, and dorkbot. He lives in Memphis with his wife, writer Amy Benson, their nine-year-old son, a thriving worm bin, and a small garden.

Daniela Fantechi is an Italian composer and a doctoral researcher working at the Orpheus Institute in Ghent (Belgium). Her research focuses on the use of piezoelectric microphones within the context of instrumental composition with electronics.

www.danielafantechi.eu

soundcloud.com/daniela-84

Illustration Credits

All photographs and illustrations by Simon Lonergan except:

Figures 1.4, 1.6, 5.1, 5.4, 7.4, 8.4, 8.7, 8.8, 12.1, 14.2, and 23.1–23.6 © Nicolas Collins.
Figure 4.2 © Lorin Parker, used by permission.
Figure 6.2 © Gert-Jan Prins, used by permission.
Figure 6.3 courtesy Ikon Gallery Birmingham, used by permission of Christina Kubisch.
Figure 6.4 © Jonáš Gruska, used by permission.
Figure 6.5 © Sawako Kato, used by permission.
Figure 6.6 © Vasco Alves, used by permission.
Figure 8.2 by David Tudor, used by permission of Jean Rigg and the David Tudor Trust. © David Tudor Trust
Figure 8.10 © Ute Wassermann, used by permission.
Figure 8.12 © Gert Jan van Rooij, used by permission of Nicolas Collins.
Figure 8.13 © Bob Seay, used by permission of Jesse Seay.
Figure 8.14 © Chris Black and Christine White, used by permission.
Figures 9.1, 9.2, 9.3, 9.4, 9.5, 9.6 © Jess Rowland, used by permission.
Figure 10.2 drawing by Laurie Anderson © Laurie Anderson, used by permission of Laurie Anderson/Canal Street Communications.
Figure 10.3 © César Eugenio Dávila-Irizarry, used by permission.
Figures 12.4 and 18.13 © Seth Cluett, used by permission.
Figures 12.5 and 17.13 (enclosure) © Robert Poss, used by permission.
Figure 15.14 © Paul DeMarinis, used by permission.
Figure 15.15 © Norbert Möslang, used by permission.
Figure 15.16 © Christian Terstegge, used by permission.
Figure 16.1 © Becky Stern, used by permission.
Figures 16.2 and 16.3 © Hannah Perner-Wilson, used by permission.
Figures 16.4 and 16.5 © Liza Stark, used by permission.
Figures 16.6 and 16.8 © Afroditi Psarra, used by permission.
Figure 16.7 © Adrian Freed, used by permission.
Figure 16.9 © Lara Grant, used by permission.
Figure 16.10 © Sarah Grant, used by permission.
Figure 16.11 © Irene Posch, used by permission.
Figure 16.12 © Rachel Freire and Sophia Brueckner, used by permission.
Figures 16.13 and 16.14 © Leah Buechley, used by permission.
Figure 16.15 © Emilie Giles and Janet van der Linden, used by permission.
Figure 16.16 © Sam Topley, used by permission.

Figures 16.17–16.50 © Lara and Sarah Grant, used by permission.
Figure 17.6 © Frederic Rzewski, used by permission.
Figure 18.8 lower left © Douglas Ferguson, used by permission.
Figure 18.8 lower right © Steve Marsh, used by permission.
Figures 18.8 upper left, 21.1–21.13 © Alex Inglizian, used by permission.
Figures 22.1, 22.2, 22.4, 22.7, 22.8, 22.9 © Jon Satrom, used by permission.
Figures 22.3, 22.5 © LoVid, used by permission.
Figure 22.4 © Jonas Bers, used by permission.
Figures 25.1–25.4 © Michael Johnsen, used by permission.
Figures 26.1–26.7 © Peter Blasser, used by permission.
Figures 27.1–27.16 © Brett Ian Balogh, used by permission.
Figures 28.1–28.15, 29.1–29.9 © Holger Heckeroth, used by permission.
Figures 30.1–30.4 Peter Speer, used by permission.
Figures 31.1–31.8 © Wolfgang Spahn, used by permission.
Figures 32.1–32.26 © Eduardo F. Rosario, used by permission.
Figures 33.1 and 33.2 © Joseph Kramer, used by permission.
Figures 34.1–33.8 © Robb Drinkwater, used by permission.
Figures 35.1–35.9 © Nick Briz, used by permission.
Figure 36.1 © Manuel Andrade, used by permission.
Figure 36.2 © Waft Lab, used by permission.
Figure 36.3 © David Novak, used by permission.
Figure 36.4 © Pedro David, used by permission.
Figure 36.5 © Sam Berry Loper, used by permission.
Figure 37.1 © Kayu Nakada, used by permission.
Figure 37.2 © Vincent O'Connor, used by permission.
Figure 37.3 © Vivian Caccuri, used by permission.
Figure 37.4 © Sebastián Rey, used by permission.
Figure 37.5 © Douglas Repetto, used by permission.
Figure 37.6 © Ragnhild May, used by permission.
Figure 37.7 © Science Museum/Science and Society Picture Library, used by permission.

Index

Note: Page numbers in *italic* indicate a figure on the corresponding page.