Learning the Art of Electronics

This introduction to circuit design is unusual in several respects.

First, it offers not just explanations, but a full lab course. Each of the 25 daily sessions begins with a discussion of a particular sort of circuit followed by the chance to try it out and see how it actually behaves. Accordingly, students understand the circuit's operation in a way that is deeper and much more satisfying than the manipulation of formulas.

Second, it describes circuits that more traditional engineering introductions would postpone: thus, on the third day, we build a radio receiver; on the fifth day, we build an operational amplifier from an array of transistors. The digital half of the course centers on applying microcontrollers, but gives exposure to Verilog, a powerful Hardware Description Language.

Third, it proceeds at a rapid pace but requires no prior knowledge of electronics. Students gain intuitive understanding through immersion in good circuit design.

- Each session is divided into several parts, including Notes, Labs; many also have Worked Examples and Supplementary Notes
- An appendix introducing Verilog
- Further appendices giving background facts on oscilloscopes, Xilinx, transmission lines, pinouts, programs etc, plus advice on parts and equipment
- Very little math: focus is on intuition and practical skills
- A final chapter showcasing some projects built by students taking the course over the years

Thomas C. Hayes reached electronics via a circuitous route that started in law school and eventually found him teaching Laboratory Electronics at Harvard, which he has done for thirty-five years. He has also taught electronics for the Harvard Summer School, the Harvard Extension School, and for seventeen years in Boston University's Department of Physics. He shares authorship of one patent, for a device that logs exposure to therapeutic bright light. He and his colleagues are trying to launch this device with a startup company named Goodlux Technologies. Tom designs circuits as the need for them arises in the electronics course. One such design is a versatile display, serial interface and programmer for use with the microcomputer that students build in the course.

Paul Horowitz is a Research Professor of Physics and of Electrical Engineering at Harvard University, where in 1974 he originated the Laboratory Electronics course from which emerged *The Art of Electronics*.

Learning the Art of Electronics

A Hands-On Lab Course

Thomas C. Hayes

with the assistance of Paul Horowitz

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning, and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9780521177238

First published 2016

5th printing with corrections 2020

Printed in the United Kingdom by TJ International Ltd. Padstow Cornwall

A catalogue record for this publication is available from the British Library.

Library of Congress Cataloguing in Publication Data

ISBN 978-0-521-17723-8 Paperback

For Debbie, Tessa, Turner and Jamie

And in memory of my beloved friend, Jonathan

Contents

Preface — *page* xx
Overview, as the Course begins — xxv

Part I Analog: Passive Devices — 1

1N DC Circuits — 3
1N.1 Overview — 3
1N.2 Three laws — 5
1N.3 First application: voltage divider — 11
1N.4 Loading, and "output impedance" — 14
1N.5 Readings in AoE — 24

1L Lab: DC Circuits — 25
1L.1 Ohm's law — 25
1L.2 Voltage divider — 26
1L.3 Converting a meter movement into a voltmeter and ammeter — 27
1L.4 The diode — 29
1L.5 I versus V for some mystery boxes — 30
1L.6 Oscilloscope and function generator — 32

1S Supplementary Notes: Resistors, Voltage, Current — 35
1S.1 Reading resistors — 35
1S.2 Voltage versus current — 38

1W Worked Examples: DC circuits — 42
1W.1 Design a voltmeter, current meter — 42
1W.2 Resistor power dissipation — 44
1W.3 Working around imperfections of instruments — 45
1W.4 Thevenin models — 47
1W.5 "Looking through" a circuit fragment, and R_{in}, R_{out} — 48
1W.6 Effects of loading — 49

2N RC Circuits — 51
2N.1 Capacitors — 51
2N.2 Time-domain view of RCs — 53
2N.3 Frequency domain view of RCs — 58
2N.4 Blocking and decoupling — 74

	2N.5 A somewhat mathy view of *RC* filters	76
	2N.6 Readings in AoE	77
2L	**Labs: Capacitors**	78
	2L.1 Time-domain view	78
	2L.2 Frequency domain view	81
2S	**Supplementary Notes: RC Circuits**	85
	2S.1 Reading capacitors	85
	2S.2 *C* notes: trying for an intuitive grip on capacitors' behavior	90
	2S.3 Sweeping frequencies	93
2W	**Worked Examples: RC Circuits**	100
	2W.1 *RC* filters	100
	2W.2 *RC* step response	105
3N	**Diode Circuits**	108
	3N.1 Overloaded filter: another reason to follow our 10× loading rule	108
	3N.2 Scope probe	109
	3N.3 Inductors	112
	3N.4 *LC* resonant circuit	113
	3N.5 Diode Circuits	118
	3N.6 The most important diode application: DC from AC	119
	3N.7 The most important diode application: (unregulated-) power supply	123
	3N.8 Radio!	126
	3N.9 Readings in AoE	130
3L	**Lab: Diode Circuits**	131
	3L.1 *LC* resonant circuit	131
	3L.2 Half-wave rectifier	133
	3L.3 Full-wave bridge rectifier	134
	3L.4 Design exercise: AM radio receiver (fun!)	135
	3L.5 Signal diodes	136
3S	**Supplementary Notes and Jargon: Diode Circuits**	138
	3S.1 A puzzle: why *LC*'s ringing dies away despite Fourier	138
	3S.2 Jargon: passive devices	139
3W	**Worked Examples: Diode Circuits**	141
	3W.1 Power supply design	141
	3W.2 Z_{IN}	144
Part II	**Analog: Discrete Transistors**	149
4N	**Transistors I**	151
	4N.1 Overview of Days 4 and 5	151
	4N.2 Preliminary: introductory sketch	154

	4N.3	The simplest view: forgetting beta	155
	4N.4	Add quantitative detail: use beta explicitly	158
	4N.5	A strikingly different transistor circuit: the switch	166
	4N.6	Recapitulation: the important transistor circuits at a glance	167
	4N.7	AoE Reading	168
4L	**Lab: Transistors I**		169
	4L.1	Transistor preliminaries: look at devices out of circuit	169
	4L.2	Emitter follower	170
	4L.3	Current source	172
	4L.4	Common-emitter amplifier	172
	4L.5	Transistor switch	174
	4L.6	A note on power supply noise	176
4W	**Worked Examples: Transistors I**		178
	4W.1	Emitter follower	178
	4W.2	Phase splitter: input and output impedances of a transistor circuit	181
	4W.3	Transistor switch	185
5N	**Transistors II**		188
	5N.1	Some novelty, but the earlier view of transistors still holds	188
	5N.2	Reviewish: phase splitter	189
	5N.3	Another view of transistor behavior: Ebers–Moll	190
	5N.4	Complication: distortion in a high-gain amplifier	194
	5N.5	Complications: temperature instability	196
	5N.6	Reconciling the two views: Ebers–Moll meets $I_C = \beta \times I_B$	201
	5N.7	“Difference” or “differential” amplifier	201
	5N.8	Postscript: deriving r_e	207
	5N.9	AoE Reading	208
5L	**Lab: Transistors II**		209
	5L.1	Difference or differential amplifier	209
5S	**Supplementary Notes and Jargon: Transistors II**		220
	5S.1	Two surprises, perhaps, in behavior of differential amp	220
	5S.2	Current mirrors; Early effect	222
	5S.3	Transistor summary	230
	5S.4	Important circuits	232
	5S.5	Jargon: bipolar transistors	235
5W	**Worked Examples: Transistors II**		237
	5W.1	High-gain amplifiers	237
	5W.2	Differential amplifier	238
	5W.3	Op-amp innards: diff-amp within an IC operational amplifier	239

Part III Analog: Operational Amplifiers and their Applications 243

6N Op-amps I 245
6N.1 Overview of feedback 245
6N.2 Preliminary: negative feedback as a general notion 248
6N.3 Feedback in electronics 249
6N.4 The op-amp golden rules 251
6N.5 Applications 252
6N.6 Two amplifiers 252
6N.7 Inverting amplifier 254
6N.8 When do the Golden Rules apply? 256
6N.9 Strange things can be put into feedback loop 259
6N.10 AoE Reading 261

6L Lab: Op-Amps I 262
6L.1 A few preliminaries 262
6L.2 Open-loop test circuit 263
6L.3 Close the loop: follower 263
6L.4 Non-inverting amplifier 265
6L.5 Inverting amplifier 265
6L.6 Summing amplifier 266
6L.7 Design exercise: unity-gain phase shifter 266
6L.8 Push–pull buffer 268
6L.9 Current to voltage converter 269
6L.10 Current source 271

6W Worked Examples: Op-Amps I 273
6W.1 Basic difference amp made with an op-amp 273
6W.2 A more exotic difference amp 276
6W.3 Problem: odd summing circuit 277

7N Op-amps II: Departures from Ideal 280
7N.1 Old: subtler cases, for analysis 281
7N.2 Op-amp departures from ideal 284
7N.3 Four more applications 294
7N.4 Differentiator 300
7N.5 Op-amp Difference Amplifier 301
7N.6 AC amplifier: an elegant way to minimize effects of op-amp DC errors 301
7N.7 AoE Reading 302

7L Labs: Op-Amps II 303
7L.1 Integrator 303
7L.2 Differentiator 306
7L.3 Slew rate 308
7L.4 AC amplifier: microphone amplifier 308

7S Supplementary Notes: Op-Amp Jargon 310

7W	**Worked Examples: Op-Amps II**	311
	7W.1 The problem	311
	7W.2 Op-amp millivoltmeter	314
8N	**Op-Amps III: Nice Positive Feedback**	319
	8N.1 Useful positive feedback	319
	8N.2 Comparators	320
	8N.3 *RC* relaxation oscillator	327
	8N.4 Sine oscillator: Wien bridge	331
	8N.5 AoE Reading	335
8L	**Lab. Op-Amps III**	336
	8L.1 Two comparators	336
	8L.2 Op-amp *RC* relaxation oscillator	338
	8L.3 Easiest *RC* oscillator, using IC Schmitt trigger	339
	8L.4 Apply the sawtooth: PWM motor drive	340
	8L.5 IC *RC* relaxation oscillator: ’555	341
	8L.6 ’555 for low-frequency frequency modulation (“FM”)	342
	8L.7 Sinewave oscillator: Wien bridge	343
8W	**Worked Examples: Op-Amp III**	345
	8W.1 Schmitt trigger design tips	345
	8W.2 Problem: heater controller	348
9N	**Op-Amps IV: Parasitic Oscillations; Active Filter**	353
	9N.1 Introduction	353
	9N.2 Active filters	354
	9N.3 Nasty “parasitic” oscillations: the problem, generally	356
	9N.4 Parasitic oscillations in op-amp circuits	356
	9N.5 Op-amp remedies for keeping loops stable	361
	9N.6 A general criterion for stability	365
	9N.7 Parasitic oscillation without op-amps	367
	9N.8 Remedies for parasitic oscillation	370
	9N.9 Recapitulation: to keep circuits quiet...	372
	9N.10 AoE Reading	372
9L	**Labs. Op-Amps IV**	373
	9L.1 VCVS active filter	373
	9L.2 Discrete transistor follower	374
	9L.3 Op-amp instability: phase shift can make an op-amp oscillate	377
	9L.4 Op-amp with buffer in feedback loop	378
9S	**Supplementary Notes. Op-Amps IV**	380
	9S.1 Op-amp frequency compensation	380
	9S.2 Active filters: how to improve a simple *RC* filter	384
	9S.3 Noise: diagnosing fuzz	389
	9S.4 Annotated LF411 op-amp schematic	395
	9S.5 Quantitative effects of feedback	396

CAMBRIDGE
Cambridge University Press
978-0-521-17723-8 — Learning the Art of Electronics
Thomas C. Hayes , Paul Horowitz
Frontmatter
More Information

9W Worked Examples: Op-Amps IV 401
9W.1 What all that op-amp gain does for us 401
9W.2 Stability questions 402

10N Op-Amps V: PID Motor Control Loop 407
10N.1 Examples of real problems that call for this remedy 408
10N.2 The PID motor control loop 408
10N.3 Designing the controller (custom op-amp) 410
10N.4 Proportional-only circuit: predicting how much gain the loop can tolerate 412
10N.5 Derivative, D 415
10N.6 AoE Reading 420

10L Lab. Op-Amps V 421
10L.1 Introduction: why bother with the PID loop? 421
10L.2 PID motor control 422
10L.3 Add derivative of the error 428
10L.4 Add integral 430
10L.5 Scope images: effect of increasing gain, in P-only loop 432

11N Voltage Regulators 433
11N.1 Evolving a regulated power supply 434
11N.2 Easier: 3-terminal *IC* regulators 439
11N.3 Thermal design 441
11N.4 Current sources 443
11N.5 Crowbar overvoltage protection 444
11N.6 A different scheme: switching regulators 445
11N.7 AoE Readings 450

11L Lab: Voltage Regulators 451
11L.1 Linear voltage regulators 451
11L.2 A switching voltage regulator 457

11W Worked Examples: Voltage Regulators 462
11W.1 Choosing a heat sink 462
11W.2 Applying a current-source IC 463

12N MOSFET Switches 465
12N.1 Why we treat FETs as we do 465
12N.2 Power switching: turning something ON or OFF 469
12N.3 A power switch application: audio amplifier 471
12N.4 Logic gates 473
12N.5 Analog switches 474
12N.6 Applications 475
12N.7 Testing a sample-and-hold circuit 480
12N.8 AoE Reading 485

12L	**Lab: MOSFET Switches**	486
	12L.1 Power MOSFET	486
	12L.2 Analog switches	489
	12L.3 Switching audio amplifier	495
12S	**Supplementary Notes: MOSFET Switches**	497
	12S.1 A physical picture	497
13N	**Group Audio Project**	503
	13N.1 Overview: a day of group effort	503
	13N.2 One concern for everyone: stability	506
	13N.3 Sketchy datasheets for LED and phototransistor	507
13L	**Lab: Group Audio Project**	508
	13L.1 Typical waveforms	508
	13L.2 Debugging strategies	509
Part IV	**Digital: Gates, Flip-Flops, Counters, PLD, Memory**	511
14N	**Logic Gates**	513
	14N.1 Analog versus digital	513
	14N.2 Number codes: Two's-complement	518
	14N.3 Combinational logic	520
	14N.4 The usual way to do digital logic: programmable arrays	526
	14N.5 Gate types: TTL and CMOS	528
	14N.6 Noise immunity	530
	14N.7 More on gate types	533
	14N.8 AoE Reading	535
14L	**Lab: Logic Gates**	537
	14L.1 Preliminary	537
	14L.2 Input and output characteristics of integrated gates: TTL and CMOS	540
	14L.3 Pathologies	541
	14L.4 Applying IC gates to generate particular logic functions	543
	14L.5 Gate innards; looking within the black box of CMOS logic	544
14S	**Supplementary Notes: Digital Jargon**	548
14W	**Worked Examples: Logic Gates**	550
	14W.1 Multiplexing: generic	550
	14W.2 Binary arithmetic	554
15N	**Flip-Flops**	567
	15N.1 Implementing a combinational function	568
	15N.2 Active-low, again	569
	15N.3 Considering gates as "Do this/do that" functions	573
	15N.4 XOR as Invert/Pass* function	574

	15N.5	OR as Set/Pass* function	575
	15N.6	Sequential circuits generally, and flip-flops	575
	15N.7	Applications: more debouncers	582
	15N.8	Counters	583
	15N.9	Synchronous counters	584
	15N.10	Another flop application: shift-register	586
	15N.11	AoE Reading	587
15L	**Lab: Flip-Flops**		**588**
	15L.1	A primitive flip-flop: *SR* latch	588
	15L.2	*D* type	588
	15L.3	Counters: ripple and synchronous	591
	15L.4	Switch bounce, and three debouncers	592
	15L.5	Shift register	594
15S	**Supplementary Note: Flip-Flops**		**597**
	15S.1	Programmable logic devices	597
	15S.2	Flip-flop tricks	599
16N	**Counters**		**603**
	16N.1	Old topics	603
	16N.2	Circuit dangers and anomalies	607
	16N.3	Designing a larger, more versatile counter	610
	16N.4	A recapitulation of useful counter functions	614
	16N.5	Lab 16L's divide-by-*N* counter	615
	16N.6	Counting as a digital design strategy	616
16L	**Lab: Counters**		**617**
	16L.1	A fork in the road: two paths into microcontrollers	617
	16L.2	Counter lab	619
	16L.3	16-bit counter	621
	16L.4	Make horrible music	629
	16L.5	Counter applications: stopwatch	631
16W	**Worked Examples: Applications of Counters**		**634**
	16W.1	Modifying count length: strange-modulus counters	634
	16W.2	Using a counter to measure period, thus many possible input quantities	636
	16W.3	Bullet timer	642
17N	**Memory**		**648**
	17N.1	Buses	648
	17N.2	Memory	651
	17N.3	State machine: new name for old notion	655
17L	**Lab: Memory**		**661**
	17L.1	RAM	661
	17L.2	State machines	663
	17L.3	State machine using a PAL programmed in Verilog	669

17S	**Supplementary Notes: Digital Debugging and Address Decoding**	671
	17S.1 Digital debugging tips	671
	17S.2 Address decoding	675
17W	**Worked Examples: Memory**	678
	17W.1 A sequential digital lock	678
	17W.2 Solutions	681
Part V	**Digital: Analog–Digital, PLL, Digital Project Lab**	687
18N	**Analog ↔ Digital; PLL**	689
	18N.1 Interfacing among logic families	689
	18N.2 Digital ⇔ analog conversion, generally	693
	18N.3 Digital to analog (DAC) methods	697
	18N.4 Analog-to-digital conversion	701
	18N.5 Sampling artifacts	712
	18N.6 Dither	714
	18N.7 Phase-locked loop	716
	18N.8 AoE Reading	723
18L	**Lab: Analog ↔ Digital; PLL**	724
	18L.1 Analog-to-digital converter	724
	18L.2 Phase-locked loop: frequency multiplier	729
18S	**Supplementary Notes: Sampling Rules; Sampling Artifacts**	734
	18S.1 What's in this chapter?	734
	18S.2 General notion: sampling produces predictable artifacts in the sampled data	734
	18S.3 Examples: sampling artifacts in time- and frequency-domains	735
	18S.4 Explanation? The images, intuitively	739
18W	**Worked Examples: Analog ↔ Digital**	745
	18W.1 ADC	745
	18W.2 Level translator	748
19L	**Digital Project Lab**	749
	19L.1 A digital project	749
Part VI	**Microcontrollers**	755
20N	**Microprocessors 1**	757
	20N.1 Microcomputer basics	757
	20N.2 Elements of a minimal machine	760
	20N.3 Which controller to use?	762
	20N.4 Some possible justifications for the hard work of the big-board path	764
	20N.5 Rediscover the micro's control signals...	765
	20N.6 Some specifics of our lab computer: big-board branch	771
	20N.7 The first day on the SiLab branch	773
	20N.8 AoE Reading	778

20L	**Lab: Microprocessors 1**	780
	20L.1 Big-board Dallas microcomputer	780
	20L.2 Install the GLUEPAL; wire it partially	781
	20L.3 SiLabs 1: startup	792
20S	**Supplementary Notes: Microprocessors 1**	803
	20S.1 PAL for microcomputers	803
	20S.2 Note on SiLabs IDE	805
20W	**Worked Examples: A Garden of Bugs**	809
21N	**Microprocessors 2. I/O, First Assembly Language**	813
	21N.1 What is assembly language? Why bother with it?	813
	21N.2 Decoding, again	818
	21N.3 Code to use the I/O hardware (big-board branch)	821
	21N.4 Comparing assembly language with C code: keypad-to-display	824
	21N.5 Subroutines: CALL	826
	21N.6 Stretching operations to 16 bits	830
	21N.7 AoE Reading	831
21L	**Lab: Microprocessors 2**	832
	21L.1 Big-board: I/O. Introduction	832
	21L.2 SiLabs 2: input; byte operations	844
21S	**Supplementary Notes: 8051 Addressing Modes**	857
	21S.1 Getting familiar with the 8051's addressing modes	857
	21S.2 Some 8051 addressing modes illustrated	867
22N	**Micro 3: Bit Operations**	869
	22N.1 BIT operations	869
	22N.2 Digression on conditional branching	874
22L	**Lab Micro 3. Bit Operations; Timers**	881
	22L.1 Big-board lab. Bit operations; interrupt	881
	22L.2 SiLabs 3: Timers; PWM; Comparator	886
22W	**Worked Examples. Bit Operations: An Orgy of Error**	901
	22W.1 The problem	901
	22W.2 Lots of poor, and one good, solutions	901
	22W.3 Another way to implement this "Ready" key	904
23N	**Micro 4: Interrupts; ADC and DAC**	905
	23N.1 Big ideas from last time	905
	23N.2 Interrupts	906
	23N.3 Interrupt handling in C	911
	23N.4 Interfacing ADC and DAC to the micro	912

	23N.5 Some details of the ADC/DAC labs	917
	23N.6 Some suggested lab exercises, playing with ADC and DAC	921
23L	**Lab Micro 4. Interrupts; ADC and DAC**	926
	23L.1 ADC → DAC	926
	23L.2 SiLabs 4: Interrupt; DAC and ADC	931
23S	**Supplementary Notes: Micro 4**	946
	23S.1 Using the RIDE assembler/compiler and simulator	946
	23S.2 Debugging	951
	23S.3 Waveform processing	955
24N	**Micro 5. Moving Pointers, Serial Buses**	959
	24N.1 Moving pointers	959
	24N.2 DPTR can be useful for SiLabs ’410, too: tables	964
	24N.3 End tests in table operations	964
	24N.4 Some serial buses	966
	24N.5 Readings	974
24L	**Lab Micro 5. Moving Pointers, Serial Buses**	975
	24L.1 Data table; SPI bus; timers	976
	24L.2 SiLabs 5: serial buses	982
24S	**Supplementary Note: Dallas Program Loader**	993
	24S.1 Dallas downloader	993
	24S.2 Hardware required	993
	24S.3 Procedure to try the loader: two versions	994
	24S.4 Debugging: LOADER420, in case you can’t write to flash	999
	24S.5 Debugging in case of trouble with COM port assignments	1000
24W	**Worked Example: Table Copy, Four Ways**	1003
	24W.1 Several ways to copy a table	1003
25N	**Micro 6: Data Tables**	1006
	25N.1 Input and output devices for a microcontroller	1006
	25N.2 Task for big-board users: standalone micro	1008
	25N.3 Task for SiLabs users: off-chip RAM	1009
25L	**Lab: Micro 6: Standalone Microcontroller**	1012
	25L.1 Hardware alternatives: two ways to program the flash ROM	1012
	25L.2 SiLabs 6: SPI RAM	1018
	25L.3 Appendix: Program Listings	1021
26N	**Project Possibilities: Toys in the Attic**	1022
	26N.1 One more microcontroller that may interest you	1023
	26N.2 Projects: an invitation and a caution	1025
	26N.3 Some pretty projects	1025

	26N.4	Some other memorable projects	1030
	26N.5	Games	1041
	26N.6	Sensors, actuators, gadgets	1043
	26N.7	Stepper motor drive	1049
	26N.8	Project ideas	1051
	26N.9	Two programs that could be useful: LCD,Keypad	1052
	26N.10	And many examples are shown in AoE	1052
	26N.11	Now go forth	1052
A	**A Logic Compiler or HDL: Verilog**		1053
	A.1	The form of a Verilog file: design file	1053
	A.2	Schematics can help one to debug	1054
	A.3	The form of a Verilog file: simulation testbench	1055
	A.4	Self-checking testbench	1058
	A.5	Flip-flops in Verilog	1060
	A.6	Behavioral versus structural design description: easy versus hard	1064
	A.7	Verilog allows hierarchical designs	1065
	A.8	A BCD counter	1068
	A.9	Two alternative ways to instantiate a sub-module	1070
	A.10	State machines	1071
	A.11	An instance more appropriate to state form: a bus arbiter	1073
	A.12	Xilinx ISE offers to lead you by the hand	1076
	A.13	Blocking versus non-blocking assignments	1077
B	**Using the Xilinx Logic Compiler**		1080
	B.1	Xilinx, Verilog, and ABEL: an overview	1080
C	**Transmission Lines**		1089
	C.1	A topic we have dodged till now	1089
	C.2	A new case: transmission line	1090
	C.3	Reflections	1092
	C.4	But why do we care about reflections?	1094
	C.5	Transmission line effects for sinusoidal signals	1097
D	**Scope Advice**		1099
	D.1	What we don't intend to tell you	1099
	D.2	What we'd like to tell you	1099
E	**Parts List**		1105
F	**The Big Picture**		1113
G	**"Where Do I Go to Buy Electronic Goodies?"**		1114
H	**Programs Available on Website**		1116

I Equipment 1119
I.1 Uses for This List 1119
I.2 Oscilloscope 1119
I.3 Function generator 1120
I.4 Powered breadboard 1120
I.5 Meters, VOM and DVM 1121
I.6 Power supply 1121
I.7 Logic probe 1121
I.8 Resistor substitution box 1121
I.9 PLD/FPGA programming pod 1122
I.10 Hand tools 1122
I.11 Wire 1122

J Pinouts 1123
J.1 Analog 1123
J.2 Digital 1125
Index 1128

Preface

A book and a course

This is a book for the impatient. It's for a person who's eager to get at the fun and fascination of putting electronics to work. The course squeezes what we facetiously call "all of electronics" into about twenty-five days of class. Of course, it is nowhere near *all*, but we hope it is enough to get an eager person launched and able to design circuits that do their tasks well.

Our title claims that this volume, which obviously is a *book*, is also a *course*. It is that, because it embodies a class that Paul Horowitz and I taught together at Harvard for more than 25 years. It embodies that course with great specificity, providing what are intended as day-at-a-time doses.

A day at a time: Notes, Lab, Problems, Supplements

Each day's dose includes not only the usual contents of a *book* on electronics – notes describing and explaining new circuits – but also a *lab* exercise, a chance to try out the day's new notions by building circuits that apply these ideas. We think that building the circuits will let you understand them in a way that reading about them cannot.

In addition, nearly every day includes a *worked example* and many days include what we call "supplementary notes." These – for example, early notes on how to read resistors and capacitors – are not for every reader. Some people don't need the note because they already understand the topic. Others will skip the note because they don't want to invest the time on a first pass through the book. That's fine. That's just what we mean by "supplementary:" it's something (like a supplementary vitamin) that may be useful, but that you can quite safely live without.

What's new?

If any reader is acquainted with the *Student Manual...*, published in 1989 to accompany the second edition of *The Art of Electronics*, it may be worth noting principal differences between this book and that one. First, this book means to be self-sufficient, whereas the earlier book was meant to be read alongside the larger work. Second, the most important changes in content are these:

- Analog:
  - we devote a day primarily to the intriguing and difficult topic of *parasitic* oscillations and their cures;
  - we give a day to building a "PID" circuit, stabilizing a feedback loop that controls a motor's position. We apply signals that form three functions of an *error* signal, the difference between target voltage and output voltage: "Proportional" (P), "Integral" (I), and "Derivative" (D) functions of that difference.

- Digital:
  - application of Programmable Logic Devices (PLDs or "PALs"), programmed with the high-level *hardware description language* (HDL), Verilog;
  - a shift from use of a microprocessor to a *microcontroller*, in the computer section that concludes the course. This microcontroller, unlike a microprocessor, can operate with little or no additional circuitry, so it is well-suited to the construction of useful devices rather than computers.
- Website: The book's website `learningtheartofelectronics.com` has a lot more things, in particular code in machine readable form. Appendix H lists these.

. . . And the style of this book

A reader will gather early on that this book, like the *Student Manual* is strikingly informal. Many figures are hand-drawn; notation may vary; explanations aim to help intuition rather than to offer a mathematical view of circuits. We emphasize *design* rather than *analysis*. And we try hard to devise applications for circuits that are fun: we like it when our designs make sounds (on a good day they emit *music*), and we like to see motors spin.

Who's likely to enjoy this book and course

You need not resemble the students who take our course at the university, but you may be interested to know who they are, since the course evolved with them in mind. We teach the course in three distinct forms. Most of our students take it during fall and spring daytime classes at the College. There, about half are undergraduates in the sciences and engineering; the other half are graduate students, including a few cross-registered from MIT who need an introduction quicker (and, admittedly, less deep) than electronics courses offered down there. (We don't get EE majors from there; we get people who want a less formal introduction to the subject.)

In the night version of the course, we get mostly older students, many of whom work with technology and who have become curious about what's in the "box" that they work with. Most often the mysterious "box" is simply a computer, and the student is a programmer. Sometimes the "box" is a lab setup (we get students from medical labs, across the river), or an industrial control apparatus that the student would like to demystify.

In the summer version of the course, about half our students are rising high school seniors – and the ablest of these prove a point we've seen repeatedly: to learn circuit design you don't need to know any substantial amount of physics or sophisticated math. We see this in the College course, too, where some of our outstanding students have been Freshmen (though most students are at least two or three years older).

And we can't help boasting, as we did in the preface to the 1989 *Student Manual*, that once in a great while a professor takes our course, or at least sits in. One of these buttonholed one of us recently in a hallway, on a visit to the University where he was to give a talk. "Well, Tom," he said, "one of your students finally made good." He was modestly referring to the fact that he'd recently won a Nobel Prize. We wish we could claim that we helped him get it. We can't. But we're happy to have him as an alumnus.[1]

We expect that some of these notes will strike you as elementary, some as excessively dense: your

[1] This was Frank Wilczek. He did sit quietly at the back of our class for a while, hoping for some insights into a simulation that he envisioned. If those insights came, they probably didn't come from us.

reaction naturally will reflect the uneven experience you have had with the topics we treat. Some of you are sophisticated programmers, and will sail through the assembly-language programming near the course's end; others will find it heavy going. That's all right. The course out of which this book grew has a reputation as fun, and not difficult in one sense, but difficult in another: the concepts are straightforward; abstractions are few. But we do pass a lot of information to our students in a short time; we do expect them to achieve literacy rather fast. This course is a lot like an introductory language course, and we hope to teach by the method sometimes called immersion. It is the laboratory exercises that do the best teaching; we hope this book will help to make those exercises instructive. I have to add though, in the spirit of modern jurisprudence, a reminder to read the legal notice appended to this Preface.

The mother ship: Horowitz & Hill's The Art of Electronics

Paul Horowitz launched this course, 40-odd years ago, and he and Winfield Hill wrote the book that, in its various editions, has served as textbook for this course. That book, now in its third edition and which we will refer to as "AoE," remains the reference work on which we rely. We no longer require that students buy it as they take our course. It is so rich and dense that it might cause intellectual indigestion in a student just beginning his study of electronics. But we know that some of our students and readers will want to look more deeply into topics treated in this book, and to help those people we provide cross-references to AoE throughout this book. The fortunate student who has access to AoE can get more than this book by itself can offer.

Analog and digital: a possible split

In our College course we go through all the book's material in one term of about thirteen weeks. In the night course, which meets just once each week, we do the same material in two terms. The first term treats analog (Days 1–13), the second treats digital (Days 14–26). We know that some other universities use the same split, analog versus digital. It is quite possible to do the digital half before the analog. Only on the first day of digital – when we ask that people build a logic gate from MOSFET switches – would a person without analog training need a little extra guidance. For the most part, the digital half treats its devices as black boxes that one need not crack open and understand. We do need to be aware of input and output properties, but these do not raise any subtle analog questions

It is also possible to pare the course somewhat, if necessary. We don't like to see any of our labs missed, but we know that the summer version of the class, which compresses it all into a bit more than six weeks, makes the tenth lab optional (Day 10 presents a "PID" motor controller). And the summer course omits the gratifying but not-essential digital project lab, 20L, in which students build a device of their own design.

Who helped especially with this book

First, and most obviously, comes Paul Horowitz, my teacher long ago, my co-teacher for so many years, and all along a demanding and invaluable critic of the book as it evolved. Most of the book's hand-drawn figures, as well, still are his handiwork. Without Paul and his support, this book would not exist.

Second, I want to acknowledge the several friends and colleagues who have looked closely at parts of the book and have improved and corrected these parts. Two are friends with whom I once taught,

and who thus not only are expert in electronics but also know the course well. These are Steve Morss and Jason Gallicchio. Steve and I taught together nearly thirty years ago. Back then, he helped me to try out and to understand new circuits. He then went off to found a company, but we stayed in touch, and when we began to use a logic compiler in the course (Verilog) I took advantage of his experience. Steve was generous with his advice and then with a close reading of our notes on the subject. As I first met Verilog's daunting range of powers it was very good to be able to consult a patient and experienced practitioner.

Jason helped especially with the notes on sampling. He has the appealing but also intimidating quality of being unable to give half-power, light criticism. I was looking for pointers on details. The draft of my notes came back glowing red with his astute markups. I got more help than I'd hoped for – but, of course, that was good for the notes.

A happy benefit of working where I do is to be able to draw on the extremely knowledgeable people about me, when I'm stumped. Jim MacArthur runs the electronics shop, here, and is always overworked. I could count on finding him in his lab on most weekends, and, if I did, he would accept an interruption for questions either practical or deep. David Abrams is a similarly knowledgeable colleague who twice has helped me to explain to students results that I and the rest of us could not understand. With experience in industry as well as in teaching our course, David is another specially valuable resource.

Curtis Mead, one of Paul Horowitz's graduate students, gave generously of his skill in circuit layout, to help us make the LCD board that we use in the digital parts of this course. Jake Connors, who had served as our teaching assistant, also helped to produce the LCD boards that Curtis had laid out. Randall Briggs, another of our former TAs, helped by giving a keen, close reading.

It probably goes without saying, but let's say it: whatever is wrong in this book, despite the help I've had, is my own responsibility, my own contribution, not that of any wise advisor.

In the laborious process of producing readable versions of the book's thousand-odd diagrams, two people gave essential help. My son, Jamie Hayes, helped first by drawing, and then by improving the digital images of scanned drawings. Ray Craighead, a skilled illustrator whom we found online,[2] made up intelligently rendered computer images from our raggedy hand-drawn originals. He was able to do this in a style that does not jar too strikingly when placed alongside our many hand-drawn figures. We found no one else able to do what Ray did.

Then, when the pieces were approximately assembled, but still very ragged, the dreadfully hard job of putting the pieces together, finding inconsistencies and repetitions, cutting references to figures that had been cut, attempting to impose some consistency, ("Carry_Out" rather than "$\text{Carry}_{\text{OUT}}$" or "$C_{\text{out}}$" – at least on the same page – and so on), in 1000 pages or so, fell to my editor, David Tranah. He put up not only with the initial raggedness, but also with continual small changes, right to the end, and he did this soon after he had completed a similarly exhausting editing of AoE. For this unflagging effort I am both admiring and grateful.

And, finally, I should thank my wife, Debbie Mills, for tolerating the tiresome sight of me sitting, distracted, in many settings – on back porch, vacation terrace in Italy, fireside chair – poking away at revisions. She will be glad that the book, at last, is done.

[2] See craighead.com.

Legal notice

In this book we have attempted to teach the techniques of electronic design, using circuit examples and data that we believe to be accurate. However, the examples, data, and other information are intended solely as teaching aids and should not be used in any particular application without independent testing and verification by the person making the application. Independent testing and verification are especially important in any application in which incorrect functioning could result in personal injury or damage to property.

For these reasons, we make no warranties, express or implied, that the examples, data, or other information in this volume are free of error, that they are consistent with industry standards, or that they will meet the requirements for any particular application. THE AUTHORS AND PUBLISHER EXPRESSLY DISCLAIM THE IMPLIED WARRANTIES OF MERCHANTABILITY AND OF FITNESS FOR ANY PARTICULAR PURPOSE, even if the authors have been advised of a particular purpose, and even if a particular purpose is indicated in the book. The authors and publisher also disclaim all liability for direct, indirect, incidental, or consequential damages that result from any use of the examples, data, or other information in this book.

In addition, we make no representation regarding whether use of the examples, data, or other information in this volume might infringe others' intellectual property rights, including US and foreign patents. It is the reader's sole responsibility to ensure that he or she is not infringing any intellectual property rights, even for use which is considered to be experimental in nature. By using any of the examples, data, or other information in this volume, the reader has agreed to assume all liability for any damages arising from or relating to such use, regardless of whether such liability is based on intellectual property or any other cause of action, and regardless of whether the damages are direct, indirect, incidental, consequential, or any other type of damage. The authors and publisher disclaim any such liability.

Overview, as the Course begins

The circuits of the first three days in this course are humbler than what you will see later, and the devices you meet here are probably more familiar to you than, say, transistors, operational amplifiers – or microprocessors: Ohm's Law will surprise none of you; $I = CdV/dt$ probably sounds at least vaguely familiar.

But the circuit elements that this section treats – passive devices – appear over and over in later active circuits. So, if a student happens to tell us, "I'm going to be away on the day you're doing Lab 2," we tell her she will have to make up the lab somehow. We tell her that the second lab, on *RC* circuits, is the most important in the course. If you do not use that lab to cement your understanding of *RC* circuits – especially filters – then you will be haunted by muddled thinking for at least the remainder of the analog part of the course.

Resistors will give you no trouble; diodes will seem simple enough, at least in the view that we settle for: they are one-way conductors. Capacitors and inductors behave more strangely. We will see very few circuits that use inductors, but a great many that use capacitors. You are likely to need a good deal of practice before you get comfortable with the central facts of capacitors' behavior – easy to state, hard to get an intuitive grip on: they pass AC, block DC, and only *rarely* cause large phase shifts.

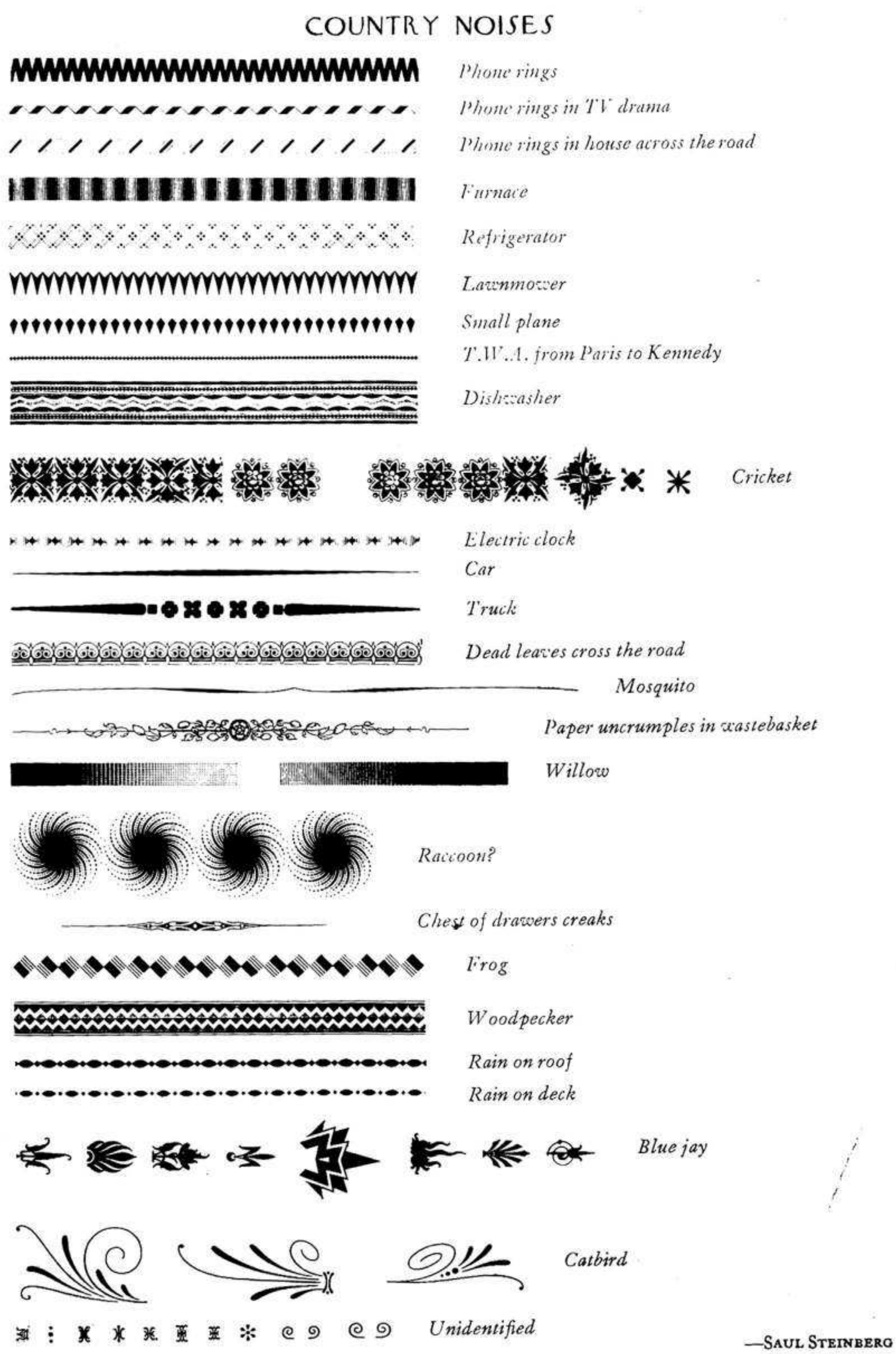
We should also restate a word of reassurance: you can manage this course perfectly even if the "$-j$" in the expression for the capacitor's impedance is completely unfamiliar to you. If you consult AoE, and after reading about complex impedances in AoE's spectacularly dense Math Review (Appendix A) you feel that you must be spectacularly dense, don't worry. That is the place in the course where the squeamish may begin to wonder if they ought to retreat to some slower-paced treatment of the subject. Do not give up at this point; hang on until you have seen transistors, at least. One of the most striking qualities of this book is its cheerful evasion of complexity whenever a simpler account can carry you to a good design. The treatment of transistors offers a good example, and you ought to stay with the course long enough to see that: the transistor chapter is difficult, but wonderfully simpler than most other treatments of the subject. You will begin designing useful transistor circuits on your first day with the subject.

It is also in the first three labs that you will get used to the lab instruments – and especially to the most important of these, the oscilloscope. It is a complex machine; only practice will teach you to use it well. Do not make the common mistake of thinking that the person next to you who is turning knobs so confidently, flipping switches and adjusting trigger level – all on the first or second day of the course – is smarter than you are. No, that person has done it before. In two weeks, you too will be making the scope do your bidding – assuming that you don't leave the work to that person next to you, who knew it all from the start.

The images on the scope screen make silent and invisible events visible, though strangely abstracted as well; these scope traces will become your mental images of what happens in your circuits. The scope will serve as a time microscope that will let you see events that last a handful of nanoseconds:

the length of time light takes to get from you to the person sitting a little way down the lab bench. You may even find yourself reacting emotionally to shapes on the screen, feeling good when you see a smooth, handsome sinewave, disturbed when you see the peaks of the sine clipped, or its shape warped; annoyed when fuzz grows on your waveforms.

Anticipating some of these experiences, and to get you in the mood to enjoy the coming weeks in which small events will paint their self-portraits on your screen, we offer you a view of some scope traces that never quite occurred, and that nevertheless seem just about right: just what a scope would show if it could. This drawing was posted on my door for years, and students who happened by would pause, peer, hesitate – evidently working a bit to put a mental frame around these not-quite-possible pictures. Sometimes a person would ask if these are scope traces. They are not, of course; the leap beyond what a scope can show was the artist's: Saul Steinberg's. Graciously, he has allowed us to show his drawing here. We hope you enjoy it. Perhaps it will help you to look on your less exotic scope displays with a little of the respect and wonder with which we have to look on the traces below.



Drawing by Saul Steinberg, copyright Saul Steinberg Foundation; originally published in *The New Yorker Magazine*, 1979. reproduced with permission.

Cambridge University Press
978-0-521-17723-8 — Learning the Art of Electronics
Thomas C. Hayes , Paul Horowitz
Frontmatter
More Information