

---

A SELF-GUIDED LABORATORY CURRICULUM

---

# Post-X Chapters Laboratory Course

---

*A Comprehensive Project-Based Approach to  
Advanced Analog, Digital, and Embedded Systems Design*

Based on the teachings of

*The Art of Electronics: The X Chapters*

by Paul Horowitz and Winfield Hill

CURRICULUM GUIDE & PROJECT ROADMAP

December 1, 2025

---

## **About This Document**

This curriculum guide is designed to complement *The Art of Electronics: The X Chapters* by providing a structured, project-based approach to mastering advanced electronics concepts. The projects herein are organized as progressive “boss fights” that demonstrate mastery of the skills taught in the book.

## **Prerequisites**

This curriculum assumes familiarity with basic electronics concepts as covered in the main volume of *The Art of Electronics*. Students should be comfortable with fundamental circuit analysis, basic op-amp configurations, digital logic principles, and have some experience with laboratory equipment and PCB design.

## **Safety Notice**

Several projects in this curriculum involve high voltages, high currents, and potentially hazardous conditions. Always follow proper safety protocols, use appropriate protective equipment, and work within your skill level. When in doubt, seek guidance from experienced practitioners.

## **Document Version**

Version 1.0 — December 1, 2025

# Contents

<b>List of Projects</b>	<b>4</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Purpose and Scope . . . . .	5
1.2 Learning Objectives . . . . .	5
1.3 Curriculum Structure . . . . .	5
1.4 Using This Guide . . . . .	6
<b>2 The X Chapters as Design Reference</b>	<b>7</b>
2.1 Mapping Projects to Book Sections . . . . .	7
2.2 Pre-Flight Checklist Philosophy . . . . .	7
2.3 Cross-Cutting Concerns . . . . .	8
2.3.1 Noise and Signal Integrity . . . . .	8
2.3.2 Thermal Management . . . . .	8
2.3.3 Electromagnetic Compatibility . . . . .	9
<b>3 Phase 1: Advanced Analog &amp; Power Core</b>	<b>10</b>
3.1 Project A1: Multi-Output Lab Power Supply . . . . .	10
3.1.1 Design Considerations . . . . .	11
3.2 Project A2: Precision Transresistance Amplifier . . . . .	13
3.2.1 Design Considerations . . . . .	14
3.3 Project A3: High-Speed Op-Amp Circuit . . . . .	16
3.3.1 Design Considerations . . . . .	17
3.4 Project A4: High-Energy Pulser / Coil Driver / LED Driver . . . . .	18
3.4.1 Design Considerations . . . . .	19
3.5 Project A5: Precision Measurement Instrument . . . . .	21
3.5.1 Design Considerations . . . . .	22
<b>4 Phase 2: Digital &amp; Embedded Foundations</b>	<b>24</b>
4.1 Project D1: Breadboard Computer or Minimal Microprocessor . . . . .	24
4.1.1 Design Considerations . . . . .	25
4.2 Project D2: FPGA-Based System . . . . .	27
4.2.1 Design Considerations . . . . .	28
<b>5 Phase 3: Embedded Systems with RTOS</b>	<b>30</b>
5.1 Project E1: ARM MCU + RTOS Lab Platform . . . . .	30
5.1.1 Design Considerations . . . . .	31
5.2 Project E2: Embedded Instrument / Controller . . . . .	33
5.2.1 Design Considerations . . . . .	34
<b>6 Project Management and Workflow</b>	<b>36</b>
6.1 Kanban-Based Project Tracking . . . . .	36
6.1.1 Recommended Workflow Stages . . . . .	36
6.1.2 Example Task Cards . . . . .	36

## CONTENTS

---

6.2 Documentation Standards . . . . .	37
6.2.1 Design Documentation . . . . .	37
6.2.2 Version Control . . . . .	37
6.3 Tool Recommendations . . . . .	37
6.3.1 Electronic Design Automation (EDA) . . . . .	37
6.3.2 Embedded Development . . . . .	38
<b>A Quick Reference Tables</b>	<b>39</b>
A.1 SI Unit Prefixes . . . . .	39
A.2 Common Voltage Noise Densities . . . . .	39
A.3 Thermal Design Quick Reference . . . . .	40
<b>B Suggested Reading</b>	<b>41</b>
B.1 Primary References . . . . .	41
B.2 Supplementary References . . . . .	41
B.2.1 Analog Design . . . . .	41
B.2.2 Power Electronics . . . . .	41
B.2.3 Digital and Embedded . . . . .	41

# List of Projects

ID	Project Title	Phase	Primary Focus
A1	Multi-Output Lab Power Supply	Phase 1	Power Electronics
A2	Precision Transresistance Amplifier	Phase 1	Low-Noise Analog
A3	High-Speed Op-Amp Circuit	Phase 1	High-Frequency Design
A4	High-Energy Pulser / Coil Driver	Phase 1	Switching & Protection
A5	Precision Measurement Instrument	Phase 1	Instrumentation
D1	Breadboard Computer	Phase 2	Digital Fundamentals
D2	FPGA-Based System	Phase 2	HDL & Synthesis
E1	ARM MCU + RTOS Lab Platform	Phase 3	Embedded Systems
E2	Embedded Instrument / Controller	Phase 3	System Integration

# Chapter 1

## Introduction

### 1.1 Purpose and Scope

*The Art of Electronics: The X Chapters* represents an advanced treatment of practical electronics design, delving into the subtleties and real-world considerations that separate functional circuits from truly excellent ones. This companion curriculum transforms the theoretical knowledge gained from studying the X Chapters into demonstrated practical competence through a carefully structured series of projects.

The projects presented herein are designed as “boss fights”—challenging, comprehensive exercises that require the integration of multiple concepts and skills. Successfully completing these projects demonstrates not just understanding of individual techniques, but the ability to synthesize them into complete, working systems.

### 1.2 Learning Objectives

Upon completion of this curriculum, you will be able to:

1. **Design high-performance analog and power circuits** that behave well in the real world, accounting for parasitics, stability, EMI, and layout considerations.
2. **Build precision measurement instruments** that actually achieve their specified noise, bandwidth, and accuracy targets through careful attention to noise budgeting, shielding, and calibration.
3. **Design and debug fast amplifiers**—including voltage-feedback (VFB) and current-feedback (CFB) op-amps, transresistance amplifiers, and fast drivers—with oscillation or other misbehavior.
4. **Build robust power stages** including high-current supplies, high-voltage pulsers, coil drivers, and LED drivers with proper switching techniques, snubbing, and thermal management.
5. **Progress through digital system design** from discrete logic through microprocessors, FPGAs, and modern MCUs with real-time operating systems, making them interface reliably with real hardware.

### 1.3 Curriculum Structure

The curriculum is organized into three progressive phases:

---

Phase	Focus Area	Projects
Phase 1	Advanced Analog & Power Core	A1–A5
Phase 2	Digital & Embedded Foundations	D1–D2
Phase 3	Embedded Systems with RTOS	E1–E2

---

The phases are designed to build upon each other, with earlier projects creating infrastructure (both physical and conceptual) for later ones. Project A1, the multi-output lab power supply, literally becomes the power infrastructure for all subsequent work.

## 1.4 Using This Guide

Each project section includes:

- A clear **goal statement** defining the project deliverable
- **Key skills** that map directly to X Chapters content
- **Implementation suggestions** for approaching the design
- **Specifications** where appropriate
- **Connections** to other projects in the curriculum

Before committing any design to a PCB, use the relevant X Chapters sections as a pre-flight checklist. Ask yourself: Have I accounted for parasitics? Is the loop stable? Are protection and derating appropriate? Is the layout consistent with the frequencies and currents involved?

# Chapter 2

## The X Chapters as Design Reference

### 2.1 Mapping Projects to Book Sections

The X Chapters can be treated as a comprehensive design reference throughout this curriculum. The following mapping shows how book content relates to specific projects:

X Chapters Topic	Relevant Projects
FETs & Power Handling	Pass elements, switching stages, gate drive (A1, A4)
Op-Amps & High-Speed Design	Transresistance amps, VFB/CFB designs, differential stages (A2, A3)
Parasitics & Non-Idealities	PCB layout, cabling, shielding (All projects)
Digital Design & Modern Logic	Breadboard CPU, FPGA platforms (D1, D2)
Embedded & Interfaces	ARM + RTOS designs, mixed-signal integration (E1, E2)

### 2.2 Pre-Flight Checklist Philosophy

Treat the book as a pre-flight checklist before committing to any PCB design. The following questions should be answered affirmatively before proceeding to fabrication:

### Pre-Flight Checklist

#### 1. Parasitics Considered?

- Have I identified all significant parasitic capacitances, inductances, and resistances?
- Are stray capacitances accounted for in high-impedance nodes?
- Have I considered skin effect and proximity effect at the frequencies involved?

#### 2. Loop Stability Verified?

- Have I analyzed the feedback loop for phase margin?
- Is compensation adequate for all loading conditions?
- Have I considered the effect of cable capacitance on amplifier stability?

#### 3. Protection and Derating Appropriate?

- Are components operated within their safe operating area (SOA)?
- Is thermal design adequate for worst-case conditions?
- Are appropriate protection mechanisms in place (overcurrent, overvoltage, ESD)?

#### 4. Layout Consistent with Requirements?

- Does the layout minimize loop areas for high-frequency currents?
- Are sensitive analog sections properly isolated from noisy digital sections?
- Is the grounding scheme appropriate (single point vs. ground plane)?

## 2.3 Cross-Cutting Concerns

Several topics from the X Chapters apply across multiple projects:

### 2.3.1 Noise and Signal Integrity

Every analog project requires attention to noise. The X Chapters provide detailed treatment of:

- Johnson-Nyquist (thermal) noise in resistors
- Op-amp voltage and current noise
- $1/f$  (flicker) noise at low frequencies
- Noise bandwidth and equivalent noise bandwidth
- Ground loops and common-mode rejection

### 2.3.2 Thermal Management

Power dissipation and thermal design affect:

- Pass device selection and heatsinking (A1)
- Resistor power ratings and thermal noise (A2, A5)
- MOSFET gate drive and switching losses (A4)
- Long-term reliability of all designs

### 2.3.3 Electromagnetic Compatibility

EMC considerations pervade all designs:

- Radiated emissions from fast switching edges
- Conducted noise on power supply rails
- Susceptibility to external interference
- Shielding and filtering techniques

# Chapter 3

## Phase 1: Advanced Analog & Power Core

### Phase 1 Overview

Phase 1 develops core competencies in analog and power electronics through five interconnected projects. The multi-output lab power supply (A1) provides infrastructure for all subsequent work, while the remaining projects explore precision measurement, high-speed design, and power switching.

### 3.1 Project A1: Multi-Output Lab Power Supply

#### Project A1 — Multi-Output Lab Power Supply

##### Goal

Design and build a comprehensive bench power supply providing multiple regulated outputs suitable for powering all subsequent projects in this curriculum. This supply becomes the **infrastructure** for all later work.

##### Target Specifications

Output	Voltage	Current	Notes
Fixed Digital 1	5.0 V	3 A	Logic supply
Fixed Digital 2	3.3 V	3 A	Modern logic/MCU
Adjustable	0 V to 24 V	2 A to 3 A	General purpose
Symmetric 1	12 V	1 A	Op-amp positive rail
Symmetric 2	-12 V	1 A	Op-amp negative rail

**Key X-Chapters Skills Exercised**

- **FETs and Pass Devices:** Understanding linear vs. switching regulator topologies, MOSFET selection for pass elements, thermal considerations for power semiconductors.
- **Compensation and Loop Stability:** Designing stable feedback loops, understanding pole/zero placement, ensuring adequate phase margin under varying load conditions.
- **Current Limiting and Protection:** Implementing foldback current limiting, safe operating area (SOA) protection, thermal shutdown.
- **Layout for Performance:** PCB design for low noise and good transient response, ground plane strategies, decoupling and filtering.

**Implementation Suggestions****Staged Development Approach:****Stage 1: Single Regulated Rail**

Begin with one regulated output (e.g., the 5 V rail). Focus on getting the basic regulation loop working correctly with proper compensation. Verify stability under step load changes.

**Stage 2: Add Current Limit and Metering**

Implement current limiting with appropriate fold-back characteristics. Add voltage and current measurement circuitry for the front panel display. Consider implementing a digital monitoring header ( $I^2C/SPI$ ) for future MCU integration (Project E1).

**Stage 3: Symmetric Rails**

Add the  $\pm 12$  V symmetric rails. Pay attention to tracking between positive and negative supplies. Consider the start-up sequencing to avoid latch-up in powered circuits.

**Stage 4: Adjustable Rail**

The adjustable 0 V to 24 V rail presents additional challenges in maintaining loop stability across the full output range. This should be implemented last.

**Additional Features to Consider:**

- Remote sense lines for accurate voltage at the load
- Over-temperature protection on all pass devices
- Soft-start to limit inrush current
- Output enable/disable control
- Status LEDs for current limit, over-temperature, etc.

### 3.1.1 Design Considerations

#### Topology Selection

The choice between linear and switching regulation involves trade-offs:

Aspect	Linear	Switching
Noise	Very low output noise	Higher ripple, switching noise
Efficiency	Poor at large $V_{in} - V_{out}$	High efficiency (80–95%)
Thermal	Significant heat dissipation	Lower power dissipation
Complexity	Simpler design	More complex control
EMI	Minimal	Requires careful layout

A hybrid approach often works well: use switching pre-regulators to reduce the voltage drop across linear post-regulators, achieving both low noise and reasonable efficiency.

### Loop Compensation

The feedback loop must be stable under all load conditions. Key considerations include:

- Output capacitor ESR creates a zero that can aid or hinder compensation
- Load capacitance affects loop dynamics
- Phase margin should exceed 45° for good transient response
- Gain margin should exceed 10 dB

### Protection Circuits

Robust protection is essential for a lab supply:

- **Overcurrent:** Current limiting should engage smoothly without oscillation. Foldback limiting reduces pass device dissipation during shorts.
- **SOA Protection:** Power MOSFETs must operate within their safe operating area. This is particularly challenging during current limiting at high  $V_{DS}$ .
- **Thermal:** Temperature sensors on pass devices should trigger shutdown before damage occurs.
- **Reverse Current:** Protection against current flowing backward into the supply from charged capacitors or other sources.

## 3.2 Project A2: Precision Transresistance Amplifier

### Project A2 — Precision Transresistance Amplifier (Photodiode Front-End)

#### Goal

Build a low-noise photodiode amplifier that converts small currents (pA to  $\mu$ A) into voltages for precision optical measurements. This project develops essential skills in low-noise analog design and demonstrates the challenges of measuring small signals.

#### Target Specifications

Parameter	Specification
Input Current Range	1 pA to 10 $\mu$ A
Transimpedance	100 k $\Omega$ to 1 M $\Omega$ (switchable)
Bandwidth (Low-Noise Mode)	1 kHz to 10 kHz
Bandwidth (Fast Mode)	100 kHz to 1 MHz
Input-Referred Noise	< 10 fA/ $\sqrt{\text{Hz}}$ at high gain

#### Key X-Chapters Skills Exercised

- **Noise Budgeting:** Calculating contributions from op-amp voltage noise, current noise, and resistor thermal noise. Understanding the trade-off between bandwidth and noise.
- **Stability with Capacitive Sources:** Photodiodes present significant capacitance (tens to hundreds of pF). This capacitance, combined with feedback resistance, creates a pole that can cause instability.
- **Guarding, Shielding, and Layout:** At pA current levels, leakage currents from contamination, moisture, and PCB substrate become significant. Guard rings and proper enclosure design are essential.
- **Bias and Dark Current Management:** Photodiode reverse bias affects speed and dark current. Operating conditions must be carefully controlled.

### Implementation Suggestions

#### Design Two Versions:

##### Version 1: Ultra-Low-Noise, Slow Version

- Bandwidth: 1 kHz to 10 kHz
- Optimized for minimum noise
- Use FET-input op-amps with very low current noise
- High transimpedance (1 MΩ or higher)
- Applications: precision photometry, low-light detection

##### Version 2: Faster Version with Explicit Trade-offs

- Bandwidth: 100 kHz to 1 MHz
- Accept higher noise for increased speed
- May use BJT-input op-amps for lower voltage noise
- Lower transimpedance (10 kΩ to 100 kΩ)
- Applications: fiber optics, fast pulse detection

Document the noise-bandwidth trade-off explicitly by measuring both versions under identical conditions.

### 3.2.1 Design Considerations

#### Noise Analysis

The total input-referred current noise has three main components:

$$i_{n,total}^2 = i_{n,amp}^2 + \frac{4kT}{R_f} + \left( \frac{e_{n,amp}}{R_f} \right)^2 \cdot (1 + 2\pi f C_{in} R_f)^2 \quad (3.1)$$

Where:

- $i_{n,amp}$  is the op-amp input current noise
- $4kT/R_f$  is the feedback resistor thermal noise (as current)
- $e_{n,amp}$  is the op-amp input voltage noise
- $C_{in}$  is the total input capacitance (photodiode + strays)

At low frequencies, the resistor and current noise dominate. At high frequencies, the voltage noise term grows due to the capacitive divider effect.

#### Stability Compensation

The photodiode capacitance and feedback resistor create a pole at:

$$f_p = \frac{1}{2\pi R_f C_d} \quad (3.2)$$

A feedback capacitor  $C_f$  is typically needed to maintain stability:

$$C_f \geq \sqrt{\frac{C_d}{2\pi R_f \cdot GBW}} \quad (3.3)$$

This capacitor also limits bandwidth, creating a fundamental trade-off in transimpedance amplifier design.

### Practical Layout Techniques

- **Guard Rings:** Surround high-impedance nodes with driven guards at the same potential to eliminate leakage currents.
- **Teflon or Ceramic Standoffs:** Standard PCB materials can have leakage currents of nA at high impedance. Use Teflon or ceramic for critical connections.
- **Shielding:** A grounded metal enclosure prevents pickup of ambient electromagnetic interference.
- **Cleanliness:** Even fingerprints can cause leakage. Handle boards carefully and consider conformal coating.

### 3.3 Project A3: High-Speed Op-Amp Circuit

#### Project A3 — High-Speed Op-Amp Circuit (VFB or CFB)

##### Goal

Design a high-speed amplifier (tens of MHz bandwidth) in a non-trivial configuration: a gain-of-10 stage, wideband buffer, or differential stage. This project develops skills in high-frequency analog design and transmission line techniques.

##### Target Configurations

- **Configuration A:** Non-inverting amplifier with gain of 10, bandwidth > 50 MHz
- **Configuration B:** Differential buffer (single-ended to differential or vice versa)
- **Configuration C:** Wideband unity-gain buffer with low output impedance

##### Key X-Chapters Skills Exercised

- **VFB vs. CFB Selection:** Understanding when to use voltage-feedback versus current-feedback amplifiers. VFB offers constant gain-bandwidth product; CFB maintains bandwidth nearly independent of gain.
- **Compensation and Layout:** At high frequencies, PCB parasitics become part of the circuit. Trace inductance, pad capacitance, and ground plane impedance all affect performance.
- **Transmission Lines and Termination:** When trace lengths approach a significant fraction of wavelength, transmission line effects appear. Proper termination eliminates reflections and ringing.
- **Minimizing Overshoot and Ringing:** Achieving clean step response requires careful attention to feedback network layout and capacitive loading effects.

##### Implementation Suggestions

###### Implementation Approach:

###### 1. Start with Gain-of-10 Non-Inverting Stage

- Choose appropriate VFB or CFB op-amp based on required GBW
- Design feedback network with attention to parasitic capacitances
- Layout for minimum loop area in feedback path

###### 2. Add Differential Buffer Stage

- Implement single-ended to differential conversion
- Ensure matched delays in both paths
- Verify common-mode rejection at high frequency

###### 3. Test and Characterize

- Measure step response with function generator and oscilloscope
- Evaluate stability margins (phase margin from step response)
- Test effect of capacitive loading on stability
- Document frequency response with network analyzer if available

### 3.3.1 Design Considerations

#### Voltage-Feedback vs. Current-Feedback

Characteristic	Voltage-Feedback (VFB)	Current-Feedback (CFB)
Bandwidth vs. Gain	$BW \propto 1/\text{Gain}$ (constant GBW)	BW nearly independent of gain
DC Precision	Generally better	Limited by input offset
Slew Rate	Limited	Very high
Feedback Resistor	Any value	Fixed optimal value
Noise	Can be very low	Typically higher
Applications	Precision, low noise	High speed, video

#### PCB Layout for High-Speed Circuits

- **Ground Plane:** Use continuous ground plane under all high-speed traces. Avoid slots or gaps that force return currents to detour.
- **Trace Geometry:** Match trace impedance to source/load. For a  $50\Omega$  system on FR4 with a ground plane, trace width of approximately  $2.5 \times$  the dielectric thickness gives  $50\Omega$ .
- **Component Placement:** Place bypass capacitors as close as possible to power pins. Multiple capacitor values (e.g.,  $0.1\mu\text{F}$ ,  $10\text{nF}$ ,  $100\text{pF}$ ) provide low impedance across a wide frequency range.
- **Feedback Network:** Keep feedback resistors close to the op-amp. Minimize stray capacitance across the feedback resistor.

#### Transmission Line Considerations

At 100 MHz, the wavelength in FR4 (assuming  $\epsilon_r \approx 4$ ) is approximately 1.5 m. Traces longer than about 15 cm ( $\frac{1}{10}$  wavelength) should be treated as transmission lines.

Termination strategies:

- **Source termination:** Resistor at driver output, value =  $Z_0 - R_{out}$
- **Parallel termination:** Resistor at receiver, value =  $Z_0$
- **AC termination:** Series RC at receiver to reduce DC power dissipation

## 3.4 Project A4: High-Energy Pulser / Coil Driver / LED Driver

### Project A4 — High-Energy Pulser / Coil Driver / LED Driver

#### Goal

Build a circuit that rapidly switches significant power into an inductive or resistive load. Options include a high-voltage pulser, fast coil driver for electromagnets, or high-current LED driver for nanosecond-scale pulses. This project develops skills in power switching, protection, and EMI management.

#### Example Configurations

- A. Fast LED Driver:** Drive high-power LEDs with nanosecond to microsecond pulses for time-resolved fluorescence or optical communication testing.
- B. Coil Driver:** Rapidly energize and de-energize an inductor for solenoid actuation, magnetic field pulsing, or motor drive testing.
- C. Lower-Voltage Pulser:** A safer introduction to pulsed power that still requires good switching technique (e.g., 48 V at 10 A peak).

#### Key X-Chapters Skills Exercised

- **MOSFET/IGBT Gate Drive:** Proper gate drive is critical for fast, efficient switching. Understand gate charge, Miller effect, and the need for low-impedance gate drivers.
- **dV/dt and dI/dt Management:** Fast switching creates high rates of voltage and current change that can cause oscillation, EMI, and component stress. Controlled slew rates balance speed against these issues.
- **Snubbers and Clamps:** Inductive loads generate voltage spikes when current is interrupted. Snubber networks and clamp diodes protect switching devices from overvoltage.
- **Layout for Power Switching:** Loop inductance in the power path causes ringing and voltage overshoot. Minimize loop area for high  $di/dt$  circuits.

### Implementation Suggestions

#### Safety-First Development:

Power switching circuits can be hazardous. Follow this progression:

##### 1. Start with Low Voltage

Develop and debug the circuit at reduced voltage (e.g., 12 V) before increasing to target levels. All switching problems will be visible at low voltage without the safety risk.

##### 2. Use Current Limiting

Include current sensing and limiting during development. A series power resistor can also limit fault currents.

##### 3. Proper Gate Drive

Use dedicated gate driver ICs rather than trying to drive MOSFETs directly from logic or op-amps. Gate drivers provide the current needed for fast switching and proper voltage levels.

##### 4. Protection First

Install snubbers and clamp diodes before connecting inductive loads. It's much easier to remove unnecessary protection than to replace destroyed transistors.

### 3.4.1 Design Considerations

#### MOSFET Gate Drive Requirements

The gate of a power MOSFET appears as a capacitor (typically tens to hundreds of nano-farads for power devices). To switch in nanoseconds, significant current is required:

$$I_{gate} = C_{iss} \cdot \frac{dV_{gs}}{dt} \quad (3.4)$$

For a MOSFET with  $C_{iss} = 5 \text{ nF}$  switching in 20 ns:

$$I_{gate} = 5 \text{ nF} \times \frac{10 \text{ V}}{20 \text{ ns}} = 2.5 \text{ A} \quad (3.5)$$

This is well beyond what a typical logic gate can provide. Dedicated gate drivers are essential.

#### Inductive Switching Transients

When current through an inductor is interrupted, the inductor generates a voltage:

$$V_L = L \cdot \frac{di}{dt} \quad (3.6)$$

For a 100  $\mu\text{H}$  inductor with 10 A being switched off in 100 ns:

$$V_L = 100 \mu\text{H} \times \frac{10 \text{ A}}{100 \text{ ns}} = 10 \text{ kV} \quad (3.7)$$

This will destroy almost any semiconductor. Clamping and snubbing are mandatory.

### Snubber Design

A basic RC snubber across the switch reduces  $dV/dt$ :

$$R_{snub} \approx \sqrt{\frac{L_{stray}}{C_{oss}}} \quad (3.8)$$

$$C_{snub} \approx \frac{I_{peak} \cdot t_{fall}}{V_{overshoot,max}} \quad (3.9)$$

Where  $L_{stray}$  is the parasitic inductance in the switching loop,  $C_{oss}$  is the MOSFET output capacitance, and  $t_{fall}$  is the current fall time.

### Layout Considerations

- **Minimize Loop Area:** The power switching loop (supply → switch → load → return) should have minimum area to reduce inductance and EMI.
- **Kelvin Connections:** For current sensing, use separate sense and power connections to avoid errors from resistive drops.
- **Gate Drive Path:** Keep the gate drive loop separate from the power loop to avoid coupling switching noise into the control circuit.
- **Decoupling:** Place bulk and high-frequency decoupling capacitors directly across the DC bus, close to the switching device.

### 3.5 Project A5: Precision Measurement Instrument

#### Project A5 — Precision Measurement Instrument

##### Goal

Synthesize analog skills from previous projects into a complete measurement instrument. Examples include a precision voltmeter front-end, low-noise AC probe, or simple LCR/impedance measurement front-end.

##### Example Instruments

###### A. Precision DC Voltmeter Front-End

- Input range:  $\pm 10 \mu\text{V}$  to  $\pm 10 \text{ V}$
- Input impedance:  $> 1 \text{ G}\Omega$
- Resolution:  $1 \mu\text{V}$

###### B. Low-Noise AC Probe

- Bandwidth: 1 Hz to 100 kHz
- Noise floor:  $< 10 \text{ nV}/\sqrt{\text{Hz}}$
- High CMRR for differential measurements

###### C. Simple Impedance Probe

- Measure R, L, C at selected frequency
- Basic phase detection
- Useful for component verification

#### Key X-Chapters Skills Exercised

- **Managing Parasitics:** At high sensitivity, stray capacitance, leakage currents, and thermoelectric EMFs all contribute errors. Understanding and minimizing these effects is essential.
- **EMI/Hum Rejection:** Power line interference at 50 Hz/60 Hz and its harmonics is omnipresent. Shielding, balanced inputs, and filtering techniques reduce pickup.
- **Cabling and Shielding:** The cable connecting the probe to the instrument is part of the measurement system. Cable capacitance, shield termination, and connector quality all matter.
- **Calibration and References:** Absolute accuracy requires calibration against known standards. Internal voltage references must be stable versus temperature, time, and load.

### Implementation Suggestions

#### **Integration Opportunity:**

This project is an excellent opportunity to combine skills from earlier projects:

- Power the instrument from Project A1
- Use transresistance amplifier techniques from Project A2 for current measurement
- Apply high-speed design principles from Project A3 for wide-bandwidth probes
- Prepare interface for digital readout (Phase 3 integration)

Consider including a digital monitoring header compatible with Project E1's MCU platform for future digitization of measurements.

### 3.5.1 Design Considerations

#### Error Budget Analysis

A precision instrument requires tracking all error sources:

Error Source	Typical Magnitude	Mitigation
Op-amp offset voltage	10 $\mu$ V–1 mV	Chopper stabilization, auto-zero
Op-amp offset drift	0.1 $\mu$ V/ $^{\circ}$ C–10 $\mu$ V/ $^{\circ}$ C	Temperature control, low-drift amps
Resistor tolerance	0.01%–1%	Precision resistors, matching
Resistor tempco	1 ppm/ $^{\circ}$ C–100 ppm/ $^{\circ}$ C	Low-tempco types
Thermoelectric EMF	1 $\mu$ V–100 $\mu$ V	Material selection, isothermal design
Reference drift	1 ppm/ $^{\circ}$ C–50 ppm/ $^{\circ}$ C	Precision references

#### Shielding and Grounding

For sensitive measurements:

- Use a **driven shield** at the input to eliminate leakage currents and reduce effective cable capacitance.
- **Guard rings** on PCB around high-impedance nodes.
- **Single-point grounding** for low-frequency precision circuits to avoid ground loops.
- **Balanced (differential) inputs** provide high common-mode rejection.
- Consider **galvanic isolation** for measurements on circuits at different potentials.

#### Calibration Strategy

A practical calibration approach:

1. **Zero calibration:** Short input and measure offset
2. **Gain calibration:** Apply known reference and adjust
3. **Linearity check:** Verify at multiple points across range

4. **Temperature characterization:** Document drift with temperature  
Store calibration constants for correction during measurement (especially useful when integrating with MCU in Phase 3).

# Chapter 4

## Phase 2: Digital & Embedded Foundations

### Phase 2 Overview

Phase 2 transitions from analog to digital, building foundational understanding of digital systems from first principles. Starting with discrete logic and progressing through FPGA implementation develops intuition for timing, synchronization, and the translation from concept to silicon.

### 4.1 Project D1: Breadboard Computer or Minimal Microprocessor

#### Project D1 — Breadboard Computer or Minimal Microprocessor

##### Goal

Build a simple CPU or microcomputer from discrete logic ICs. This project provides deep understanding of processor architecture by implementing it at the gate level, making abstract concepts concrete and debuggable.

##### Architecture Options

###### A. Classic 8-Bit CPU

Implement a minimal processor with control unit, register file, ALU, program counter, and instruction decoder. Target a simple instruction set with perhaps 8–16 instructions.

###### B. Microcoded State Machine

Build a simpler microcoded controller that drives a bus connecting RAM, ROM, and I/O. The microcode ROM defines the instruction set, making changes easier.

**Key X-Chapters Skills Exercised**

- **Digital Timing:** Understanding propagation delays through logic gates, setup and hold times at flip-flops, and how these constrain maximum clock frequency.
- **Metastability:** When asynchronous signals meet synchronous logic, metastability can occur. Learn to recognize and handle asynchronous inputs properly.
- **Bus Timing:** Read and write cycles, address decoding, bus contention—the fundamental protocols that let multiple devices share a bus.
- **Clock Distribution:** Generating a clean clock, distributing it with minimal skew, and implementing glitch-free reset circuits.

**Implementation Suggestions****Implementation Strategy:****Stage 1: Clock and Reset**

Build a reliable clock generator (crystal oscillator or 555 timer for slow clocking) and reset circuit. Test that reset properly initializes all flip-flops.

**Stage 2: Program Counter**

Implement a simple counter that sequences through memory addresses. Add ability to load new values for jumps.

**Stage 3: Memory and ROM**

Connect ROM (EEPROM or static RAM configured as ROM) for program storage. Implement address decoding for memory-mapped I/O.

**Stage 4: Register File**

Build a small set of general-purpose registers. Implement read and write ports with proper timing.

**Stage 5: ALU**

Implement basic arithmetic and logic operations. Start with ADD, AND, OR; expand as instruction set develops.

**Stage 6: Control Unit**

The heart of the CPU—decodes instructions and sequences control signals. Can be hardwired logic or microcode ROM.

**Debugging Tips:**

- Use slow clock (even single-step) during bring-up
- Add LEDs on key signals (clock, reset, bus lines)
- Build and test subsystems before integration
- Document timing diagrams for each operation

### 4.1.1 Design Considerations

#### Timing Analysis

At each clock edge, signals must be stable. The fundamental constraint is:

$$T_{clk} > T_{prop,max} + T_{setup} + T_{skew} \quad (4.1)$$

Where  $T_{prop,max}$  is the maximum propagation delay through combinational logic between

registers,  $T_{setup}$  is the flip-flop setup time, and  $T_{skew}$  is clock skew between source and destination registers.

For a breadboard computer using 74-series logic:

- Typical gate delay: 5 ns to 15 ns
- Setup time: 5 ns to 20 ns
- With multiple gates in series, clock speeds above 1 MHz–5 MHz become challenging

### Bus Architecture

A simple three-bus architecture provides clear data flow:

1. **Address Bus:** Unidirectional, from CPU to memory/peripherals
2. **Data Bus:** Bidirectional, connects all data sources/sinks
3. **Control Bus:** Various control signals (RD, WR, clock, etc.)

Use tristate buffers for devices sharing the data bus. Ensure only one device drives the bus at any time to avoid contention.

### Instruction Set Design

A minimal instruction set for first implementation:

Instruction	Operation	Description
NOP	—	No operation
LDA addr	$A \leftarrow [addr]$	Load accumulator from memory
STA addr	$[addr] \leftarrow A$	Store accumulator to memory
ADD addr	$A \leftarrow A + [addr]$	Add memory to accumulator
JMP addr	$PC \leftarrow addr$	Unconditional jump
JZ addr	if Z: $PC \leftarrow addr$	Jump if zero flag set
HLT	—	Halt execution

## 4.2 Project D2: FPGA-Based System

### Project D2 — FPGA-Based System (Verilog/VHDL)

#### Goal

Re-implement or extend Project D1's CPU on an FPGA, or create a more sophisticated digital system such as a VGA controller, digital filter, or peripheral interface. This project bridges the gap between discrete logic understanding and modern digital design tools.

#### Implementation Options

##### A. Port the Breadboard CPU

Translate the discrete logic design to Verilog or VHDL. Compare resource usage and maximum clock speed with the breadboard version.

##### B. VGA Display Controller

Generate VGA timing signals and display patterns or text. A good exercise in timing-critical design.

##### C. Digital Signal Processing

Implement a FIR filter, CORDIC algorithm, or similar DSP function. Understand fixed-point arithmetic and pipelining.

##### D. Peripheral Bridge

Create interfaces (SPI, I<sup>2</sup>C, UART) that can connect to Phase 1 analog circuits.

#### Key X-Chapters Skills Exercised

- **HDL Coding Style:** Writing synthesizable Verilog or VHDL that maps efficiently to FPGA resources. Understanding the difference between simulation and synthesis.
- **Timing Constraints:** Specifying clock frequencies and I/O timing to the synthesis tools. Understanding timing reports and meeting timing closure.
- **Clock Domain Crossing:** When multiple clock domains exist, special techniques (synchronizers, FIFOs) are needed to safely transfer data.
- **On-Chip Debugging:** Using integrated logic analyzers (ILA, SignalTap) to observe internal signals in the running FPGA.

### Implementation Suggestions

#### **Development Flow:**

##### **1. Behavioral Simulation**

Write and verify design in simulation before synthesis. Create comprehensive test-benches that exercise all functionality.

##### **2. Synthesis and Implementation**

Synthesize the design and review resource utilization and timing reports. Iterate on design if timing is not met.

##### **3. Hardware Testing**

Program the FPGA and verify basic functionality. Use on-chip debugging to observe internal signals.

##### **4. Analog Integration**

Connect FPGA I/O to Phase 1 analog circuits:

- ADC interface for reading Project A5 measurement front-end
- DAC interface for control signals
- Digital interface for Project A1 power supply monitoring

#### **Recommended FPGA Platforms:**

- Lattice iCE40 series (open-source toolchain available)
- Xilinx Artix-7 (widely used, extensive documentation)
- Intel/Altera Cyclone (good for beginners)

### 4.2.1 Design Considerations

#### Synthesizable vs. Behavioral Code

Not all valid Verilog/VHDL code is synthesizable. Common non-synthesizable constructs:

- `#delay` statements (delays are ignored in synthesis)
- Initial blocks (use reset instead)
- Dynamic memory allocation
- Real (floating-point) data types

Write code with synthesis in mind: think about what hardware your code describes.

#### Timing Closure

Meeting timing requires understanding the critical path. Common optimization strategies:

- **Pipelining:** Insert registers to break long combinational paths
- **Retiming:** Allow tools to move registers for better balance
- **Logic restructuring:** Rewrite logic to reduce depth
- **Constraint refinement:** Ensure constraints accurately reflect system requirements

#### Clock Domain Crossing (CDC)

When signals cross between clock domains, metastability is a concern. Standard solutions:

1. **Two-flop synchronizer:** For single-bit signals, reduces metastability probability to negligible levels.
2. **Gray code counters:** For multi-bit values, only one bit changes at a time, making synchronization safer.
3. **Asynchronous FIFOs:** For data streams, implement dual-clock FIFO with proper pointer synchronization.

# Chapter 5

## Phase 3: Embedded Systems with RTOS

### Phase 3 Overview

Phase 3 brings together analog, digital, and software skills into complete embedded systems. Using modern ARM microcontrollers with real-time operating systems, these projects integrate with hardware from earlier phases to create sophisticated instruments and controllers.

### 5.1 Project E1: ARM MCU + RTOS Lab Platform

#### Project E1 — ARM MCU + RTOS Lab Platform

##### Goal

Create a reusable embedded development platform based on an ARM Cortex-M microcontroller running a real-time operating system. This platform will serve as the brain for Project E2 and can interface with analog hardware from Phase 1.

##### Platform Requirements

Component	Specification
MCU Family	ARM Cortex-M4 or M7 (e.g., STM32F4/F7, IMXRT)
RTOS	FreeRTOS, Zephyr, or similar
Interfaces	UART, I <sup>2</sup> C, SPI, USB, Ethernet (optional)
Analog	ADC (multiple channels), DAC (if available)
Timing	PWM outputs, timer/counters, RTC
Debug	SWD/JTAG interface, ITM trace

**Key X-Chapters Skills Exercised**

- **Hardware/Software Partitioning:** Deciding what functions are implemented in hardware peripherals versus software. Understanding the capabilities and limitations of on-chip peripherals.
- **Interrupt Design:** Structuring interrupt handlers for minimal latency and deterministic behavior. Understanding interrupt priorities and preemption.
- **RTOS Primitives:** Using tasks, queues, semaphores, mutexes, and software timers effectively. Avoiding common pitfalls like priority inversion and deadlock.
- **Hardware Integration:** Interfacing with sensors, actuators, and external analog circuits. Managing level shifting, protection, and signal conditioning.

**Implementation Suggestions****Platform Development Stages:****Stage 1: Base Platform**

- Development board or custom PCB with MCU, power, and debug
- Basic peripherals: LEDs, buttons, UART console
- Bootloader and programming interface

**Stage 2: RTOS Bring-up**

- Port or configure RTOS for target MCU
- Create basic tasks: status LED, console I/O
- Verify preemption and timing behavior

**Stage 3: Peripheral Drivers**

- UART driver with interrupt/DMA and RTOS integration
- I<sup>2</sup>C and SPI masters for sensor/device communication
- ADC driver with sampling control and buffering
- PWM driver for motor/LED control

**Stage 4: Phase 1 Integration**

- Connect to Project A1 power supply monitoring header
- Interface with Project A2/A5 measurement front-ends via ADC
- Control Project A4 pulser/driver via PWM/GPIO

**5.1.1 Design Considerations****RTOS Architecture**

A typical RTOS-based application structure:

Task	Priority	Function
Communication	High	Handle time-critical serial protocols
Control Loop	High	Real-time control algorithms
Measurement	Medium	Acquire and process sensor data
User Interface	Low	Display updates, button handling
Background	Idle	Housekeeping, power management

### Interrupt Latency

For real-time response, understand what contributes to interrupt latency:

1. Interrupt recognition time (hardware)
2. Context save time
3. Time in higher-priority interrupts
4. Time to reach application code in ISR

On Cortex-M, typical latencies are 12–16 cycles minimum, plus any time spent in higher-priority handlers.

### Mixed-Signal Considerations

When the MCU interfaces with sensitive analog circuits:

- Separate analog and digital power domains where possible
- Use quiet clock sources (spread-spectrum can help or hurt)
- Filter analog reference and supply pins
- Careful PCB layout to prevent digital noise coupling to analog
- Consider isolation (digital isolators, isolated power) for sensitive measurements

## 5.2 Project E2: Embedded Instrument / Controller

### Project E2 — Embedded Instrument / Controller

#### Goal

Combine the embedded platform from Project E1 with analog hardware from Phase 1 into a complete, standalone instrument or controller. This capstone project integrates all skills developed throughout the curriculum.

#### Implementation Options

##### A. Digitally-Controlled Power Supply

Integrate Project A1's power supply with MCU control:

- Digital setpoints via DAC or PWM
- Voltage/current monitoring via ADC
- Protection and limiting in firmware
- User interface: OLED display, rotary encoder
- Communication: USB, SCPI commands

##### B. Data Acquisition System

Build around Project A2/A5 measurement front-ends:

- High-resolution ADC sampling
- Data logging to SD card or streaming via USB
- Triggering and capture modes
- Display of waveforms or statistics

##### C. Pulse Generator / Function Generator

Extend Project A4 with programmable control:

- Programmable pulse timing and amplitude
- Burst modes, triggering
- Waveform synthesis via DAC
- User interface for parameter entry

### Key X-Chapters Skills Exercised

- **Mixed-Signal System Management:** Successfully combining sensitive analog front-ends with digital control without mutual interference.
- **Grounding and Reference Planes:** Implementing proper grounding strategies that serve both analog accuracy and digital integrity.
- **Firmware Architecture:** Designing maintainable, reliable embedded software with proper error handling, state management, and configurability.
- **System Integration:** Making multiple subsystems work together reliably, including power sequencing, initialization, and fault handling.

### Implementation Suggestions

#### **System Integration Checklist:**

**Power Architecture**

Define power domains, sequencing requirements, and protection. Consider which supplies come from Project A1 and which are local.

**Signal Interfaces**

Document all signals between analog and digital domains. Include voltage levels, impedances, and bandwidth requirements.

**Grounding Plan**

Define how analog and digital grounds connect. Single point? Star? Ground plane with strategic gaps?

**EMC Strategy**

Plan for emissions and susceptibility. Shielding, filtering, layout techniques.

**Calibration Procedure**

Define how the system will be calibrated. Where are calibration constants stored? How are they applied?

**User Interface**

Specify display, controls, indicators. Consider remote control via USB or other interface.

**Test Plan**

Define acceptance tests for the complete system. What specifications must be verified?

### 5.2.1 Design Considerations

#### Firmware Architecture

A well-structured embedded application separates concerns:

1. **Hardware Abstraction Layer (HAL):** Provides consistent interface to peripherals, hiding register-level details.
2. **Board Support Package (BSP):** Configures HAL for specific hardware, defines pin assignments, clock configuration.
3. **Drivers:** Higher-level interfaces for specific devices (display, sensors, etc.).
4. **Application:** Business logic, user interface, communication protocols.
5. **RTOS Services:** Task management, inter-task communication, timing.

#### Reliability and Robustness

Production-quality firmware should handle:

- **Watchdog timer:** Recovery from firmware hangs
- **Brown-out detection:** Safe shutdown on power failure
- **Configuration validation:** Detect and recover from corrupted settings
- **Error logging:** Record faults for debugging

- **Safe defaults:** Fail to known-safe state

### Enclosure and Packaging

A complete instrument needs attention to mechanical design:

- **Thermal management:** Heatsinking, ventilation, thermal paths
- **EMC:** Shielding effectiveness of enclosure
- **User interface:** Accessible controls, readable display
- **Connectors:** Appropriate type and placement for all I/O
- **Safety:** Insulation, fusing, strain relief

# Chapter 6

## Project Management and Workflow

### 6.1 Kanban-Based Project Tracking

A Kanban board provides visual tracking of project progress. For this curriculum, a modified workflow accommodates the iterative nature of electronics development.

#### 6.1.1 Recommended Workflow Stages

Stage	Description
<b>Backlog</b>	Ideas and requirements not yet started
<b>Design</b>	Schematic capture, component selection, calculations
<b>Simulate</b>	SPICE simulation, HDL simulation, verification
<b>Prototype</b>	PCB layout, fabrication, assembly
<b>Measure</b>	Characterization, testing against specifications
<b>Iterate</b>	Revisions based on test results
<b>Document</b>	Complete documentation, user guides

#### 6.1.2 Example Task Cards

Break projects into manageable tasks. Example cards for various projects:

Project	Task	Acceptance Criteria
A1	Design 5V/3.3V rails	Schematic complete, components selected, BOM
A1	Simulate load regulation	Step load response < 50 mV deviation
A2	Transresistance amp noise	Simulated noise < 10 fA/ $\sqrt{\text{Hz}}$
A3	Layout high-speed PCB	DRC clean, transmission line impedance verified
A4	LED pulser gate drive	Clean 10 A pulses, < 50 ns rise
D1	Breadboard CPU instruction set	All basic instructions execute correctly
E1	RTOS task blinking	LED blinks from RTOS task, correct timing
E2	Complete power supply	Digital control, display, meets all specs

## 6.2 Documentation Standards

Good documentation supports learning and enables future reference.

### 6.2.1 Design Documentation

For each project, maintain:

1. **Requirements Document:** What the circuit must do, quantitative specifications
2. **Design Notes:** Calculations, trade-off analyses, component selection rationale
3. **Schematic:** Complete, annotated schematic with revision history
4. **Simulation Results:** Key simulation plots with interpretation
5. **PCB Documentation:** Layout files, fabrication notes, assembly drawings
6. **Test Results:** Measured performance against specifications
7. **Lessons Learned:** What worked, what didn't, improvements for next time

### 6.2.2 Version Control

Use version control (Git) for all project files:

- Schematic and PCB project files
- Simulation files and testbenches
- Firmware source code
- Documentation sources (LaTeX, Markdown)
- Test data and analysis scripts

Commit early and often with meaningful messages. Tag releases corresponding to hardware revisions.

## 6.3 Tool Recommendations

### 6.3.1 Electronic Design Automation (EDA)

---

Tool	Use Case	Cost
KiCad	Schematic & PCB design	Free/Open source
LTspice	Analog circuit simulation	Free
ngspice	Open-source SPICE	Free/Open source
Verilog/VHDL tools	Digital simulation & synthesis	Varies
OpenROAD	Open-source ASIC flow	Free/Open source

---

### 6.3.2 Embedded Development

Tool	Use Case	Cost
ARM GCC	Cross-compilation	Free
OpenOCD	Debug interface	Free/Open source
VS Code + Extensions	IDE environment	Free
PlatformIO	Build system	Free (with paid tiers)
STM32CubeIDE	STM32 development	Free

# Appendix A

## Quick Reference Tables

### A.1 SI Unit Prefixes

Prefix	Symbol	Factor
femto	f	$10^{-15}$
pico	p	$10^{-12}$
nano	n	$10^{-9}$
micro	$\mu$	$10^{-6}$
milli	m	$10^{-3}$
kilo	k	$10^3$
mega	M	$10^6$
giga	G	$10^9$

### A.2 Common Voltage Noise Densities

Source	Typical Noise Density
1 k $\Omega$ resistor @ 25 °C	4 nV/ $\sqrt{\text{Hz}}$
10 k $\Omega$ resistor @ 25 °C	13 nV/ $\sqrt{\text{Hz}}$
100 k $\Omega$ resistor @ 25 °C	40 nV/ $\sqrt{\text{Hz}}$
1 M $\Omega$ resistor @ 25 °C	128 nV/ $\sqrt{\text{Hz}}$
Low-noise op-amp (e.g., LT1028)	0.9 nV/ $\sqrt{\text{Hz}}$
FET-input op-amp (e.g., OPA627)	5 nV/ $\sqrt{\text{Hz}}$
General-purpose op-amp (e.g., TL072)	18 nV/ $\sqrt{\text{Hz}}$

### A.3 Thermal Design Quick Reference

Parameter	Typical Values
Junction-to-case thermal resistance (TO-220)	0.5 °C/W to 2 °C/W
Case-to-sink thermal resistance (with paste)	0.2 °C/W to 0.5 °C/W
Maximum junction temperature (Si)	150 °C–175 °C
Derating guideline	Operate at < 80% of max ratings

# Appendix B

## Suggested Reading

### B.1 Primary References

1. Horowitz, P. and Hill, W. *The Art of Electronics*, 3rd Edition. Cambridge University Press, 2015.
2. Horowitz, P. and Hill, W. *The Art of Electronics: The X Chapters*. Cambridge University Press, 2020.

### B.2 Supplementary References

#### B.2.1 Analog Design

- Jung, W. *Op Amp Applications Handbook*. Analog Devices, 2005. (Available free online)
- Ott, H. *Electromagnetic Compatibility Engineering*. Wiley, 2009.
- Zumbahlen, H. *Linear Circuit Design Handbook*. Newnes, 2008.

#### B.2.2 Power Electronics

- Erickson, R. and Maksimović, D. *Fundamentals of Power Electronics*. Springer, 2020.
- Pressman, A. *Switching Power Supply Design*. McGraw-Hill, 2009.

#### B.2.3 Digital and Embedded

- Wakerly, J. *Digital Design: Principles and Practices*. Pearson, 2017.
- White, E. *Making Embedded Systems*. O'Reilly, 2011.
- Barry, R. *Mastering the FreeRTOS Real Time Kernel*. (Available free at freertos.org)

*“The best way to learn electronics is to build  
things.  
The second best way is to fix them.”*

*— Paul Horowitz & Winfield Hill*