

MEGN 301

Mechanical Integration & Design

Master Reference Document

Systems Engineering Process & Technical Reference

Colorado School of Mines

Department of Mechanical Engineering

Adam Duran

Assistant Teaching Professor

PE · PMP

Publication Date: March 2026

A comprehensive guide to the design-build process from problem definition through operations and maintenance, with technical reference material for electromechanical system design.

Contents

How to Use This Document	xv
What Should I Be Reading Right Now?	xvii
Laboratory Safety and Equipment Care	xix
I The Design-Build Process	1
1 Problem Definition & Stakeholder Engagement	2
1.1 Defining the Problem	2
1.1.1 Market Analysis and Competitive Positioning	2
1.1.2 Customer Discovery and the Entrepreneurial Mindset	2
1.1.3 Problem vs. Solution Thinking	3
1.1.4 The Five-Element Framework	4
1.1.5 Scope and Boundaries	4
1.2 Stakeholder Engagement	5
1.2.1 Identifying Stakeholders	5
1.2.2 The Power/Interest Grid	5
1.2.3 Needs Elicitation Techniques	5
1.2.4 Managing Conflicting Needs	6
1.2.5 Needs vs. Wants	6
1.3 From Problem to ConOps	6
1.3.1 Concept of Operations	6
1.3.2 Use Cases and Operational Scenarios	7
1.4 Artifact Handoff: What Chapter 1 Produces	8
1.5 Stakeholder Analysis Techniques	8
1.5.1 Stakeholder Identification	8
1.5.2 Stakeholder Interviews	8
1.5.3 Needs vs. Wants vs. Constraints	8
1.6 Problem Statement Formulation	9
1.7 Competitive Benchmarking	9
1.8 Practice Questions	9
1.8.1 Answers	10
2 Value Proposition & Triple Bottom Line	12
2.1 Why Value Proposition Matters in Engineering Design	12
2.2 The Four-Dimension Value Framework	12
2.2.1 Feasibility: Can We Build It?	12
2.2.2 Desirability: Does Anyone Want It?	13
2.2.3 Viability: Can It Sustain Itself Economically?	14
2.2.4 Sustainability: What Are the Long-Term Consequences?	14
2.3 The Triple Bottom Line	15

2.3.1	People (Social)	15
2.3.2	Planet (Environmental)	15
2.3.3	Profit (Economic)	15
2.4	Applying TBL to the GreenShield Project	16
2.5	Value Proposition Canvas	16
2.6	Stakeholder Value Mapping	16
2.7	Business Model Canvas Integration	17
2.8	Competitive Analysis Framework	17
2.8.1	Direct Competitors	17
2.8.2	Indirect Competitors	18
2.8.3	Non-Consumption	18
2.9	Quantifying Value: The Business Case	18
2.10	Practice Questions	18
2.10.1	Answers	19
3	Writing Requirements & Defining Architecture	20
3.1	Writing Requirements	20
3.1.1	The System Design Requirements Document (SDRD)	21
3.1.2	What Is a Requirement?	21
3.1.3	Requirement Types	21
3.1.4	Requirements Hierarchy and Traceability	21
3.1.5	The Requirements Quality Checklist	22
3.1.6	Writing Good Requirements: Five Rules	22
3.2	Defining Architecture	23
3.2.1	Functional Decomposition	23
3.2.2	From Functions to Architecture Blocks	23
3.2.3	Interface Definition	23
3.2.4	Architecture Patterns	24
3.2.5	Requirements–Architecture Iteration	24
3.3	Environmental and Regulatory Constraints	24
3.4	Accessibility and Universal Design	24
3.4.1	Regulatory Requirements	25
3.5	Trade Studies and Decision Making	25
3.5.1	The Trade Study Process	25
3.5.2	Pugh Matrix vs. Weighted Decision Matrix	25
3.5.3	Worked Example: MCU Trade Study	26
3.6	Requirements-to-Architecture Flow	26
3.7	Artifact Handoff: What Chapter 3 Produces	28
3.8	Writing Good Requirements: Advanced Techniques	28
3.8.1	The SHALL/SHOULD/MAY Hierarchy	28
3.8.2	Requirement Attributes	29
3.8.3	Deriving Subsystem Requirements	29
3.9	Architecture: Making the Right Decisions	29
3.9.1	Functional vs. Physical Architecture	29
3.9.2	Interface Control Documents (ICD)	29
3.9.3	Architecture Tradeoff Analysis	30
3.10	Practice Questions	30
3.10.1	Answers	30

4	Concept Generation & Early Design	32
4.1	Reverse Engineering	32
4.1.1	The Reverse Engineering Process	32
4.1.2	What Reverse Engineering Reveals	34
4.1.3	Reverse Engineering Outputs	34
4.1.4	From Reverse Engineering to Concept Generation	34
4.2	Generating Concepts	35
4.2.1	Diverge Then Converge	35
4.2.2	Brainstorming Rules	35
4.2.3	SCAMPER	35
4.2.4	Morphological Charts	35
4.2.5	Bio-Inspired Design	36
4.2.6	Stage 1: Pugh Screening Matrix	36
4.2.7	Stage 2: Weighted Decision Matrix	36
4.3	Risk Assessment	36
4.3.1	The Risk Matrix	36
4.3.2	Risk Mitigation Strategies	37
4.4	Failure Modes and Effects Analysis (FMEA)	37
4.4.1	Conducting a Design-Phase FMEA	37
4.4.2	Prototype Types in Detail	38
4.5	Prototyping and Feasibility	39
4.5.1	Feasibility Assessment	39
4.6	Design Documentation	39
4.6.1	Bill of Materials (BOM)	39
4.6.2	Wiring Diagrams and Schematics	39
4.6.3	Software Architecture and Flowcharts	40
4.6.4	CAD Models and Engineering Drawings	40
4.7	Artifact Handoff: What Chapter 4 Produces	41
4.8	Structured Creativity Methods	41
4.8.1	Morphological Analysis	41
4.8.2	SCAMPER Method	41
4.8.3	Biomimicry	42
4.9	Concept Selection: Beyond the Pugh Matrix	42
4.9.1	Weighted Decision Matrix	42
4.9.2	Concept Testing	43
4.10	Design for the GreenShield Competition	43
4.11	Failure Modes and Effects Analysis (FMEA) During Design	43
4.11.1	When to Perform FMEA	43
4.11.2	The FMEA Process	43
4.11.3	FMEA for GreenShield Projects	44
4.12	Practice Questions	44
4.12.1	Answers	45
5	Verification, Validation & Test Plans	46
5.1	Verification vs. Validation	46
5.2	Four Verification Methods: TAID	47
5.3	Writing Test Plans	47
5.3.1	What Makes a Good Test Plan	47

5.3.2	Test Procedure Structure	48
5.3.3	Writing Good Procedure Steps	48
5.4	Handling Results	48
5.4.1	PASS	48
5.4.2	FAIL	49
5.4.3	PARTIAL PASS	49
5.4.4	Root Cause Analysis	49
5.5	The VCRM	49
5.5.1	VCRM Evolution Over Time	49
5.5.2	End-to-End Traceability Example	50
5.6	V&V and the V-Model	50
5.7	Statistical Confidence and Test Rigor	50
5.7.1	Developmental Testing vs. Production Qualification	50
5.7.2	Practical Implications for Student Projects	50
5.7.3	The Test Environment Matters	50
5.7.4	Measurement Uncertainty	50
5.8	Edge Case and Boundary Testing	51
5.9	Minimum Viable V&V for MEGN 301	51
5.10	External Team Testing	51
5.11	Artifact Handoff: What Chapter 5 Produces	52
5.12	Test Method Selection: T-A-I-D	52
5.13	Designing for Testability	52
5.14	Writing Test Procedures	53
5.15	Practice Questions	53
5.15.1	Answers	53
5.15.2	Answers (continued)	54
6	Subsystem Integration	55
6.1	Integration Strategy	55
6.1.1	Bottom-Up Integration	55
6.1.2	Integration Stages and the V-Model	55
6.1.3	Integration Sequence Planning	58
6.1.4	Test Gates	58
6.1.5	Critical Subsystem Demonstrations	58
6.1.6	The N ² (N-Squared) Diagram	58
6.2	Artifact Handoff: What Chapter 6 Produces	59
6.3	Interface Management	59
6.3.1	Interface Control Documents (ICDs)	60
6.4	Debugging	60
6.5	Configuration Management	60
6.5.1	Critical Subsystem Demonstrations	60
6.6	The Integration Challenge	60
6.7	Integration Order Strategy	61
6.8	Interface Verification	61
6.9	Integration Testing Protocol	61
6.10	Debugging During Integration	62
6.10.1	Integration Engineer's Toolkit	62
6.11	Interface Testing	63

6.12	Integration Sequence for a Typical GreenShield Project	63
6.13	Practice Questions	64
6.13.1	Answers	64
7	Operations & Maintenance	66
7.1	Lifecycle Thinking	66
7.2	Design for Serviceability	68
7.2.1	Reliability Metrics	68
7.2.2	The Bathtub Curve	68
7.2.3	FMECA: Failure Modes, Effects, and Criticality Analysis	68
7.2.4	Writing Maintainability Requirements	69
7.3	Artifact Handoff: What Chapter 7 Produces	69
7.4	Maintenance Planning	69
7.5	Documentation, Training, and End-of-Life	69
7.5.1	Design for Disassembly (DfD)	70
7.6	Operational Concepts	70
7.7	Reliability Engineering Basics	70
7.8	Maintainability Design Principles	71
7.9	User Documentation	71
7.10	Preventive Maintenance Scheduling	71
7.11	Failure Analysis and Root Cause Investigation	72
7.12	Warranty and Service Planning	72
7.13	Technical Data Package	72
7.14	Obsolescence Management	73
7.15	Practice Questions	73
7.15.1	Answers	73
8	Post-Build Testing, Verification & Validation	75
8.1	Why Testing Is a Separate Engineering Activity	75
8.2	Test Planning and Documentation	76
8.2.1	The Test Plan	76
8.2.2	Verification Cross-Reference Matrix (VCRM)	76
8.3	Functional Testing	76
8.4	Performance Testing	77
8.5	Environmental and Robustness Testing	77
8.6	Validation: Does It Solve the Problem?	78
8.7	Non-Conformance and Corrective Action	78
8.8	Test Report and FDR Evidence Package	78
8.9	Test Equipment for MEGN 301	79
8.10	Automated Testing	79
8.11	Statistical Analysis of Test Results	79
8.12	Common Testing Mistakes	80
8.13	Practice Questions	80
8.13.1	Answers	81

9	End of Life & System Retirement	82
9.1	Why End of Life Matters	82
9.2	Failure Modes That Trigger Retirement	83
9.3	Design for End of Life (DfEOL)	83
9.3.1	Design for Recycling (DfR)	83
9.3.2	Design for Remanufacturing	83
9.4	Battery End of Life	84
9.5	Electronics End of Life	84
9.6	Mechanical Component End of Life	85
9.7	System Retirement Procedure	85
9.8	Life Cycle Cost and Total Cost of Ownership	85
9.9	Life Cycle Assessment: A Simplified Approach for MEGN 301	86
9.10	Circular Economy Principles	86
9.11	Regulatory Landscape	87
9.12	Product Takeback and Extended Producer Responsibility	87
9.13	Practice Questions	87
9.13.1	Answers	88
II	Technical Reference	89
10	Electrical & Fluid Connectors	90
10.1	Electrical Connectors	90
10.1.1	Connector Anatomy	90
10.1.2	Gender Convention	90
10.1.3	Keying and Polarization	91
10.1.4	Connection Types	91
10.1.5	Harsh-Environment Connectors	91
10.1.6	Termination Methods	92
10.1.7	Key Electrical Specifications	94
10.1.8	Electrical Connector Failure Modes	96
10.2	Fluid System Connectors	97
10.2.1	Threaded Pipe Fitting Standards	97
10.2.2	Tube Fittings and Quick-Connect Couplings	100
10.2.3	Key Fluid Connector Specifications	101
10.2.4	Fluid Connector Failure Modes	103
10.3	Selection and Design Practice	104
10.3.1	Connector Selection Logic	104
10.3.2	Design Rules of Thumb	104
10.3.3	Key Standards Reference	105
10.4	Applying Connectors in Your Design Projects	105
10.5	Quick-Reference Summary	106
10.6	Artifact Handoff: What This Chapter Supports	107
10.7	Practice Questions for Quiz Preparation	107
10.7.1	Conceptual Questions	107
10.7.2	Application / Scenario Questions	108
10.7.3	Answers	109
10.7.4	Application Scenario Answers	110

11 Power Supplies, Motors & Pumps	112
11.1 Power Supply Selection	112
11.1.1 What a Power Supply Does	112
11.1.2 Linear Regulators (LDO)	112
11.1.3 Switching Regulators (DC-DC Converters)	113
11.1.4 AC-DC Power Supplies	113
11.1.5 Key Power Supply Specifications	114
11.1.6 Power Supply Failure Modes	114
11.2 DC Motors	114
11.2.1 Motor Fundamentals	114
11.2.2 Brushed DC Motors	115
11.2.3 Brushless DC (BLDC) Motors	115
11.2.4 Stepper Motors	116
11.2.5 Servo Motors	116
11.2.6 Key Motor Specifications	116
11.2.7 Motor Drive Electronics	117
11.3 Pumps	117
11.3.1 Positive Displacement vs. Dynamic	117
11.3.2 Pump Types	117
11.3.3 Key Pump Specifications	118
11.3.4 Worked Example: Motor Selection for a Belt Conveyor	119
11.4 Selection and Design Practice	119
11.4.1 Unified Selection Framework	119
11.4.2 Design Rules	120
11.5 Artifact Handoff: What This Chapter Supports	120
11.6 Power Supply Noise and Its Effect on Measurement	121
11.7 Battery Selection for Portable Systems	121
11.8 Practice Questions	122
11.8.1 Conceptual Questions	122
11.8.2 Application / Scenario Questions	123
11.8.3 Answers	123
12 Power Architecture & Circuit Safety	125
12.1 Power Architecture	125
12.1.1 System-Level Design	125
12.1.2 Grounding Strategy	126
12.1.3 Battery Integration	127
12.1.4 Power Budget Spreadsheet: How to Build One	127
12.2 Protection Circuits	128
12.2.1 Overvoltage and Undervoltage Protection (OVP / UVP)	128
12.2.2 Overcurrent Protection (OCP)	129
12.2.3 Reverse Polarity and Back-Current Protection	130
12.2.4 Transient Voltage Suppression	130
12.3 Electromechanical Integration	131
12.3.1 Gear Reduction	131
12.3.2 Output Mechanisms	132
12.4 Safety Design Checklist	132
12.5 Grounding Architecture	133

12.6	Electromagnetic Compatibility (EMC) Basics	134
12.7	Practice Questions	134
12.7.1	Conceptual Questions	134
12.7.2	Application / Scenario Questions	135
12.7.3	Answers	135
13	Electronics Fundamentals, Safety & Component Selection	138
13.1	Voltage, Current, Resistance, and Power	138
13.2	Kirchhoff's Laws	139
13.2.1	Kirchhoff's Current Law (KCL)	139
13.2.2	Kirchhoff's Voltage Law (KVL)	139
13.3	Series and Parallel Circuits	139
13.3.1	Series Combination	139
13.3.2	Parallel Combination	139
13.4	Voltage Dividers	139
13.5	Thevenin Equivalent Circuits	140
13.6	Capacitors and Inductors	140
13.7	RC and RL Time Constants	141
13.8	AC Fundamentals: Impedance, RMS, and Decibels	141
13.9	Electrical and Electronics Safety	141
13.9.1	Voltage and Current Hazards	141
13.9.2	Benchtop Power Supply Safety	141
13.9.3	Multimeter Safety	142
13.9.4	Oscilloscope Ground Clip Hazard	142
13.9.5	Static Discharge (ESD) Protection	142
13.9.6	Lithium Battery Safety	142
13.9.7	Fire Safety for Electronics	143
13.10	Resistor Color Code	143
13.11	Component Selection Reference	144
13.11.1	Resistor Tolerance and Standard Values	144
13.11.2	Power Rating	144
13.11.3	Capacitor Voltage Rating	144
13.11.4	Wire Gauge Reference (AWG)	144
13.11.5	SI Prefixes	145
13.11.6	Common Schematic Symbols	145
13.12	Grounding and Ground Loops	145
13.13	Multimeter and Oscilloscope Quick Reference	145
13.14	Breadboard Conventions	146
13.15	Practice Questions	146
13.15.1	Answers	147
14	Principles of Sensors and Actuators	148
14.1	Transduction: Converting Between Physical Domains	148
14.2	Sensor Classification by Transduction Mechanism	148
14.3	Sensor Output Types	149
14.4	Key Sensor Specifications	149
14.5	Common Sensor Types for MEGN 301 Projects	150
14.5.1	Temperature	150

14.5.2	Strain, Force, and Pressure	150
14.5.3	Motion and Position	150
14.5.4	Flow	150
14.6	Actuator Principles	150
14.6.1	Electromagnetic Actuators	150
14.6.2	Thermal Actuators	151
14.6.3	Fluid Actuators	151
14.7	Practice Questions	152
14.7.1	Answers	152
15	Mechanical Fasteners & Bearings	153
15.1	Threaded Fasteners	153
15.1.1	Bolt and Screw Anatomy	153
15.1.2	Thread Standards	153
15.1.3	Bolt Grades and Strength	154
15.1.4	Torque and Preload	154
15.1.5	Locking Methods	154
15.1.6	Non-Threaded Fasteners	155
15.2	Bearings	155
15.2.1	What Bearings Do	155
15.2.2	Bearing Types	155
15.2.3	Bearing Selection	155
15.2.4	Mounting Bearings	157
15.2.5	Failure Modes	157
15.2.6	Thread Engagement Table	157
15.2.7	Threaded Inserts for 3D-Printed Parts	157
15.2.8	Bolt Selection Checklist	157
15.3	Design Rules of Thumb	158
15.4	Threaded Fastener Design	158
15.4.1	Bolt Sizing	158
15.4.2	Thread Engagement	158
15.4.3	Thread Locking	159
15.5	Bearing Selection and Installation	159
15.5.1	Bearing Sizing	159
15.5.2	Bearing Installation	159
15.6	Design for Assembly (DfA)	159
15.7	Fastener Material Selection	160
15.8	Metric vs. Imperial Fasteners	160
15.9	Threaded Insert Technology	160
15.10	Adhesive Bonding	161
15.11	Practice Questions	161
15.11.1	Answers	162
16	Mechanical Power Transmission	163
16.1	Fundamental Relationships	163
16.2	Belt and Pulley Drives	163
16.2.1	V-Belts	163
16.2.2	Timing Belts (Synchronous)	163

16.2.3 Flat and Round Belts	164
16.3 Chain Drives	164
16.4 Gear Trains	164
16.4.1 Gear Train Design Process	165
16.4.2 Backlash and Precision	165
16.5 Shafts and Couplings	165
16.5.1 Shaft Design	165
16.5.2 Shaft-Hub Connections	165
16.5.3 Couplings	165
16.6 Linear Motion	165
16.6.1 Lead Screws	165
16.6.2 Ball Screws	165
16.6.3 Rack and Pinion	166
16.6.4 Belt-Driven Linear Stages	166
16.7 Linkages	166
16.8 Selection Framework	166
16.9 Power Transmission Fundamentals	167
16.10 Belt and Chain Drives	167
16.11 Lead Screws and Linear Motion	167
16.12 Coupling Selection	168
16.13 Efficiency and Heat Generation	168
16.14 Gear Design for 3D-Printed Prototypes	168
16.15 Motor-to-Load Matching	169
16.16 Linear Actuator Design	169
16.17 Vibration and Noise in Power Transmission	169
16.18 Practice Questions	170
16.18.1 Answers	171
17 Designing for Injection Molding	172
17.1 Injection Molding Fundamentals	172
17.1.1 Process Overview	172
17.1.2 The Four-Phase Cycle	172
17.1.3 Material Selection	173
17.2 Design for Manufacturing (DFM) Rules	174
17.2.1 Rule 1: Uniform Wall Thickness	174
17.2.2 Rule 2: Draft Angles	174
17.2.3 Rule 3: Ribs and Bosses	174
17.2.4 Rule 4: Radii and Fillets	175
17.2.5 Rule 5: Undercuts	176
17.2.6 Rule 6: Gate Location	176
17.3 Common Defects	176
17.3.1 Worked DFM Example: Sensor Enclosure	177
17.4 Tooling and Economics	177
17.4.1 Mold Construction	178
17.4.2 Mold Classifications (SPI)	178
17.5 Warpage	179
17.5.1 Causes of Warpage	179
17.5.2 Designing to Minimize Warpage	179

17.6	Surface Finish	179
17.6.1	SPI (PIA) Finish Standards	179
17.6.2	Textured Surfaces and Draft	180
17.7	Insert Molding and Overmolding	180
17.7.1	Insert Molding	180
17.7.2	Overmolding	180
17.8	Living Hinges and Snap-Fit Clips	180
17.8.1	Living Hinges	180
17.8.2	Snap-Fit Clips	181
17.9	Artifact Handoff: What This Chapter Supports	181
17.10	DFM Checklist	181
17.11	Practice Questions	182
17.11.1	Conceptual Questions	182
17.11.2	Application / Scenario Questions	182
17.11.3	Answers	183
17.11.4	Application / Scenario Questions	184
17.11.5	Answers to Application Questions	184
18	Printed Circuit Board Design Fundamentals	186
18.1	PCB Structure and Materials	186
18.1.1	Layer Stack-Ups	186
18.1.2	Key Specifications	187
18.1.3	Component Packages	187
18.2	Design Workflow	187
18.3	Schematic Capture Best Practices	188
18.3.1	Schematic Organization	188
18.3.2	Component Selection and Footprint Verification	188
18.3.3	Design Rule Configuration	189
18.4	Critical Layout Rules	189
18.4.1	Decoupling Capacitors	190
18.4.2	Ground Plane	190
18.4.3	Component Placement and Routing	190
18.5	Fabrication, Assembly, and Bring-Up	190
18.5.1	Fabrication Files	190
18.5.2	Fab Houses	190
18.5.3	Board Bring-Up Protocol	190
18.5.4	Design Review Checklist	191
18.5.5	Development Board Integration	191
18.6	EMI, Noise, and Signal Integrity	192
18.6.1	The Return Current Problem	192
18.6.2	Analog-Digital Separation	192
18.6.3	Motor Noise Mitigation on PCB	192
18.7	Thermal Management on PCBs	192
18.8	Common Mistakes	193
18.8.1	Development Boards vs Custom PCBs	194
18.9	Artifact Handoff: What This Chapter Supports	194
18.10	PCB Design Rules for Student Projects	194
18.10.1	Trace Width and Current Capacity	194

18.10.2 Clearance and Creepage	194
18.10.3 Via Design	195
18.11 Component Placement Strategy	195
18.12 Design Review Before Fabrication	195
18.13 Practice Questions	196
18.13.1 Answers	197
19 Microcontrollers & the ESP32	199
19.1 MCU Fundamentals	199
19.1.1 What Is a Microcontroller?	199
19.1.2 MCU Platform Comparison	199
19.2 ESP32 Deep Dive	199
19.2.1 Safe GPIO Selection	201
19.2.2 Wi-Fi and Bluetooth	202
19.2.3 Power Modes	202
19.3 Communication Protocols	202
19.4 Firmware and Integration	203
19.4.1 Firmware Best Practices	203
19.4.2 System Integration	203
19.5 Artifact Handoff: What This Chapter Supports	203
19.5.1 Development Environment Setup	203
19.5.2 Common Integration Patterns	203
19.5.3 Battery Sizing Worked Example	204
19.6 Sensor Interfacing Patterns	204
19.6.1 Analog Sensors	204
19.6.2 Digital Sensors (I2C)	204
19.6.3 Motor and Actuator Control	204
19.7 Debugging and Troubleshooting	205
19.7.1 Serial Monitor Debugging	205
19.7.2 Common ESP32 Failure Modes	205
19.8 Microcontroller Fundamentals	206
19.8.1 Key Specifications	206
19.9 ESP32 Pin Planning	206
19.10 Firmware Architecture for Control Systems	206
19.10.1 Superloop Architecture	206
19.10.2 Task-Based Architecture (FreeRTOS)	207
19.11 Common ESP32 Pitfalls in MEGN 301	207
19.12 Practice Questions	208
19.12.1 Answers	208
20 Systematic Debugging and Troubleshooting	210
20.1 Why Bottom-Up Debugging Matters	210
20.2 Debugging Hardware	210
20.2.1 Visual Inspection	211
20.2.2 Continuity and Short-Circuit Check	211
20.2.3 Power-Up Sequence	211
20.2.4 Signal Tracing: The Signal Walk	211
20.2.5 Common Hardware Faults	212

20.3	Debugging Firmware	212
20.3.1	Incremental Development	212
20.3.2	Common Firmware Bugs	212
20.4	Debugging Integrated Systems	212
20.4.1	The Integration Test Sequence	212
20.4.2	Common Integration Failures	213
20.4.3	The Five-Step Debugging Protocol	213
20.4.4	Debugging Tools	214
20.5	Building a Debugging Mindset	214
20.6	Practice Questions	215
20.6.1	Answers	215
III	Major Project Milestones	216
21	Preliminary Design Review (PDR)	218
21.1	Purpose and Scope	218
21.2	Deliverables	218
21.2.1	Problem Definition and Background Research	218
21.2.2	System Requirements and Traceability	218
21.2.3	Concept Development and Selection	219
21.2.4	Risk and Feasibility	219
21.2.5	Initial Test Plan	219
21.2.6	Concept Development and Selection: Detail	219
21.2.7	Budget and Schedule	220
21.2.8	Team Organization	220
21.3	Evaluation Criteria	220
21.4	The PDR Presentation: Structure and Delivery	221
21.4.1	Handling Questions	221
21.5	PDR Deliverables Checklist	221
21.6	Scoring Rubric	222
21.7	Practice Questions	222
21.7.1	Answers	222
22	Critical Design Review (CDR)	224
22.1	Purpose and Scope	224
22.2	Deliverables	224
22.2.1	Finalized Technical Design	224
22.2.2	Design Analysis and Verification	224
22.2.3	Manufacturing and Procurement	225
22.2.4	Updated Test Plan	225
22.2.5	Software and Firmware	225
22.2.6	Thermal and Structural Analysis	226
22.2.7	Integration and Test Readiness	226
22.2.8	Risk Register Update	226
22.2.9	Procurement Status	226
22.3	Evaluation Criteria	226
22.4	The CDR-to-Fabrication Transition	227

22.5 CDR Presentation Structure	227
22.6 CDR Technical Depth Expectations	228
22.7 CDR Deliverables Checklist	228
22.8 Common CDR Deficiencies	228
22.9 Practice Questions	229
22.9.1 Answers	229
23 Final Design Review (FDR)	230
23.1 Purpose and Scope	230
23.2 Deliverables	230
23.2.1 System Verification and Validation Results	230
23.2.2 Final Design Documentation	230
23.2.3 As-Designed vs. As-Built Comparison	231
23.2.4 Design Process Documentation	231
23.2.5 Operations and Maintenance Materials	231
23.2.6 External Team Testing Results	231
23.2.7 Demonstration Video	232
23.2.8 As-Designed vs. As-Built Comparison: Template	232
23.2.9 Lessons Learned: Beyond “Start Earlier”	232
23.2.10 Reflection and Future Work	232
23.2.11 Budget Final Accounting	232
23.2.12 Schedule Performance	233
23.3 Evaluation Criteria	233
23.4 The Complete Review Progression	234
23.5 Lessons Learned Documentation	234
23.6 Practice Questions	234
23.6.1 Answers	234
Glossary	236
Exam Quick-Reference	238
Why Projects Fail	239
ABET Outcome Mapping	241
GreenShield: Complete Worked Example	242
Quick Reference Tables	244
The 15 Most Expensive Mistakes	246
References	248

How to Use This Document

This reference document accompanies the MEGN 301 lecture modules and is organized to follow the complete design-build lifecycle of an engineering project.

Part I: The Design-Build Process covers systems engineering from start to finish. Nine chapters follow the natural project timeline, from defining the problem through end-of-life retirement. Use these chapters in order as your project progresses through design reviews:

- **Chapter 1** (Problem Definition), **Chapter 2** (Value Proposition), and **Chapter 3** (Requirements) support your work before and at **PDR**.
- **Chapter 4** (Concept Generation) supports the transition from PDR to **CDR**.
- **Chapter 5** (V&V Planning) and **Chapter 6** (Integration) support your work between CDR and **FDR**.
- **Chapter 7** (Operations & Maintenance), **Chapter 8** (Post-Build Testing), and **Chapter 9** (End of Life) support system handoff and your FDR deliverables.

Part II: Technical Reference provides detailed design guidance for common electromechanical subsystems. These chapters are reference material; consult them when your project involves circuit analysis, sensor/actuator selection, connectors, power supplies, motors, pumps, protection circuits, gear trains, injection-molded parts, printed circuit boards, or systematic debugging. Chapter 13 (Electronics Fundamentals) provides the foundational circuit theory and safety practices; Chapter 14 (Sensors & Actuators) covers transduction principles and selection; Chapter 20 (Debugging) teaches the bottom-up methodology for finding and fixing problems.

Every chapter includes practice questions with worked answers. Use them for self-study and exam preparation.

Part III: Major Project Milestones provides detailed guidance for the three formal design reviews, PDR, CDR, and FDR, with deliverable checklists, evaluation criteria, and common failure patterns. These chapters explain what each review evaluates, what evidence you must present, and how the three milestones form a coherent narrative arc across the project.

The Running Case Study: GreenShield Frost Protection System

Throughout Part I, teal-bordered **GS** boxes follow a single design, the GreenShield automated greenhouse frost protection system, from problem definition through operations. Watch for these boxes to see how each artifact builds on the last.

GreenShield is an Example, Not a Template

The GreenShield system illustrates *how* to move from problem statement to requirements, how to conduct a trade study, and how to write a test plan. It is **not** a prescription for your project. Your problem, stakeholders, requirements, and architecture will be different. Use GreenShield to understand the process, not to copy components or design decisions.

The Big Picture: How Sprints Map to the Engineering V-Model

The diagram below maps the MEGN 301 sprint sequence onto the classic systems engineering V-model. The left side of the V (definition and design) and the right side (verification and validation) are mirror images: every requirement written in Sprint 1 must have a corresponding test executed

in Sprints 5–7. Keeping this traceability in mind from the start makes the later sprints dramatically smoother.

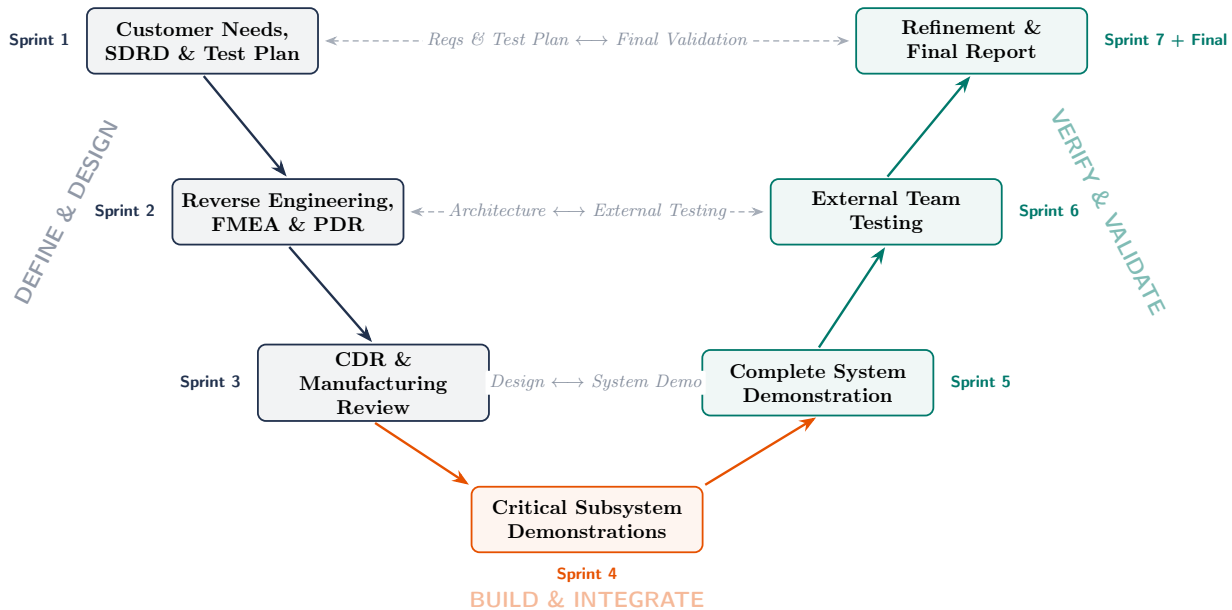


Figure 1: The MEGN 301 design project mapped onto the systems engineering V-model. Matched pairs sit at the same level: Sprint 1 requirements define the success criteria verified in the Sprint 7 final report; Sprint 2 architecture decisions are validated through Sprint 6 external testing; Sprint 3 detailed designs are verified by Sprint 5 system demonstrations. Sprint 4 sits at the vertex where design transitions to integration.

✓ First Week Survival Guide

Before your first team meeting, complete the following:

1. **Read Chapters 1–3** of this document (Problem Definition, Value Proposition, Requirements). These three chapters contain everything you need for Sprint 1.
2. **Read Chapter 20** (Debugging). You will need this methodology the moment you touch hardware. Do not wait until integration fails.
3. **Skim the Student Guide**, Sprint 1 section. It answers the most common first-week questions.
4. **Map out the semester calendar.** Identify Career Day, Presidents' Day Break, and Spring Break. Build sprint timelines around these gaps.
5. **Do the Chapter 1 practice questions.** They calibrate your understanding of problem definition and stakeholder analysis before your first real deliverable.

What Should I Be Reading Right Now?

If you are...	Your problem	Go to
Just starting your project	Need problem statement and stakeholders	Ch. 1, p. 2
Writing requirements	Need SHALL statements and SDRD format	Ch. 3, p. 20
Preparing for PDR	Need concept generation and trade study	Ch. 4, p. 32; Ch. 21, p. 218
Need circuit basics	Ohm's Law, KVL/KCL, component selection	Ch. 13, p. 138
Selecting a sensor or actuator	Transduction principles, specifications	Ch. 14, p. 148
Selecting connectors	Wire-to-wire, fluid fittings, thread types	Ch. 10, p. 90
Choosing a motor or pump	Sizing, driver selection, derating	Ch. 11, p. 112
Designing power architecture	Protection, grounding, battery safety	Ch. 12, p. 125
Designing a PCB	Schematic capture, layout, fab ordering	Ch. 18, p. 186
Programming the ESP32	Pin mapping, protocols, firmware structure	Ch. 19, p. 199
Preparing for CDR	Detailed design freeze, deliverables checklist	Ch. 22, p. 224
Integration is failing	Bottom-up strategy, debugging process	Ch. 6, p. 55; Ch. 20, p. 210
System not working	Systematic debugging methodology	Ch. 20, p. 210
Motor won't spin	Driver wiring, power rail brown-out	Ch. 11, p. 112; Ch. 12, p. 125
Sensor readings are noisy	Grounding, decoupling, layout	Ch. 13, p. 138; Ch. 18, p. 186
Writing test procedures	V&V methods, TAID, VCRM	Ch. 5, p. 46; Ch. 8, p. 75
Preparing for FDR	O&M, post-build V&V, deliverables	Ch. 7, p. 66; Ch. 23, p. 230
Studying for the exam	Formula summary and practice questions	Every chapter

Sprint-to-Chapter Mapping

Sprint	Weeks	Primary Activities	Chapters	Key Deliverables
1	1–2	Team formation, problem definition, stakeholder analysis	Ch. 1	Problem statement, stakeholder map, ConOps
2	3–4	Requirements, architecture, competitive analysis	Ch. 2, 3	SDRD, functional decomposition
3	5–6	Concept generation, trade studies, preliminary design	Ch. 4	Morph. chart, decision matrix, PDR
4	7–8	Detailed design, procurement, prototyping	Ch. 10–19	BOM, schematics, CAD, orders
5	9–10	Fabrication, subsystem testing, integration begins	Ch. 5, 6, 20	Subsystem test results, CDR
6	11–13	System integration, V&V execution, debugging	Ch. 5, 6, 8, 20	VCRM, test reports, system demos
7	14–15	Final V&V, documentation, O&M package	Ch. 7, 9	O&M manual, as-built docs
Final	16	Final design review, reflection, knowledge transfer	Part III	FDR presentation & package

Table 1: Sprint-to-chapter mapping for a 16-week MEGN 301 project.

Laboratory Safety and Equipment Care

This chapter establishes mandatory safety practices for all MEGN 301 lab and project work. Electrical and mechanical hazards are present any time you are building, testing, or demonstrating your projects. The practices below are not suggestions; they are requirements.

Electrical Hazards

The danger of electrical shock depends on current through the body, not voltage alone, but higher voltage drives more current through the body's resistance ($\sim 1\text{ k}\Omega$ hand-to-hand, wet skin). Thresholds: 1 mA is perceptible, 10–20 mA causes involuntary muscle contraction (“can’t let go”), 75–100 mA across the heart is potentially fatal.

Safe voltage thresholds (IEC 61010): below 30 VDC or 25 VAC RMS is generally safe for dry skin contact. Between 30 V and 48 V, use insulated tools and the one-hand rule (keep one hand in your pocket to prevent hand-to-hand current paths across the heart). Above 48 V, including all AC mains: de-energize and lock out before touching.

Benchtop Power Supply

Always set current limiting to the minimum expected value before enabling the output. Increase slowly while monitoring. When using multiple channels, verify isolation between channels before connecting them in series. After powering down, discharge large capacitors before touching the circuit.

Multimeter Safety

Current-Mode Dead-Short Trap

If the multimeter is left in current mode (probes in the current jacks) and connected in parallel across a voltage source, the meter becomes a short circuit. This blows the internal fuse or damages the meter and circuit. After every current measurement, move the red probe back to the voltage jack. Always measure resistance and continuity on de-energized circuits only.

Oscilloscope Ground Clip

The oscilloscope probe's ground clip connects directly to earth ground through the power cord. Clipping it to any node other than circuit ground creates a short circuit through the earth path, which can destroy components and trip breakers. The ground clip must always connect to circuit ground. For floating measurements, use channel math (Ch1 – Ch2) or a differential probe.

ESD Protection

CMOS components (ESP32, op-amps, all modern ICs) are sensitive to electrostatic discharge. A human body can accumulate 3,000–25,000 V. Wear a grounded wrist strap when handling bare ICs and PCBs, store components in anti-static bags, and touch a grounded surface before picking up components.

Lithium Battery Safety

Lithium-ion and lithium-polymer cells present fire and explosion risks if mishandled. Mandatory: never short-circuit a cell, never charge above rated voltage (typically 4.2 V/cell), never discharge below minimum (typically 3.0 V/cell), always use a battery management system (BMS), and never puncture, crush, or overheat cells. If a cell is swelling, hissing, or smoking: clear the area, place in a fire-safe container (metal bucket with sand), and notify lab staff. Lithium fires require a Class D extinguisher or dry sand, not water.

Mechanical Hazards

Rotating machinery (motors, gears, belts) can entangle hair, clothing, and fingers. Tie back long hair, remove dangling jewelry, and keep hands clear of moving parts during operation. Spring-loaded mechanisms and pressurized fluid systems store energy that can release unexpectedly. Wear safety glasses when cutting, drilling, soldering, or working with pressurized systems.

Soldering Safety

Soldering irons operate at 300–400 °C. Always return the iron to its stand when not actively soldering. Work in a ventilated area or use a fume extractor; solder flux fumes are irritating and potentially harmful with prolonged exposure. Lead-free solder is standard; if using leaded solder, wash hands thoroughly afterward and never eat or drink at the soldering bench.

Fire Safety

Electronics fires most commonly result from overcurrent through undersized wire, short circuits without fuse protection, or lithium battery failure. Prevention: fuse every power rail at or below the wire's rated current, inspect solder joints for bridges before powering up, and keep a CO₂ or dry chemical fire extinguisher (Class C rated for electrical fires) accessible in the workspace. Know the location of the nearest fire extinguisher and emergency exit before beginning any lab work.

Part I

The Design-Build Process

Chapter 1

Problem Definition & Stakeholder Engagement

When to Read This Chapter

Sprint 1: Read the full chapter. Sections 1.1–1.3 (problem definition, stakeholder analysis, ConOps) are your primary deliverables. **Sprint 2:** Return to Section 1.6 (competitive benchmarking) to refine your market analysis for PDR. **Later sprints:** Reference the stakeholder map when making design trade-offs.

1.1 Defining the Problem

1.1.1 Market Analysis and Competitive Positioning

Before defining your problem, understand what already exists. Conduct a **competitive search** appropriate to your project type:

- **Commercial product:** Search Amazon, patent databases (Google Patents), and review sites. Document 3–5 competing products with price, features, and limitations. Your problem statement must articulate what gap remains.
- **Research/instrumentation:** Search Google Scholar and IEEE Xplore for prior solutions. Document the state of the art and identify the specific capability gap.
- **Process improvement:** Benchmark current performance metrics against industry standards or competitor processes.

The competitive analysis feeds directly into your problem statement’s “WHY IT MATTERS” element; you cannot justify a project without knowing what already exists.

1.1.2 Customer Discovery and the Entrepreneurial Mindset

Engineering value creation begins with deep customer understanding. The KEEN framework emphasizes three C’s: **Curiosity** (investigate the problem space broadly), **Connections** (link technical knowledge to stakeholder needs and market realities), and **Creating Value** (design solutions that deliver measurable benefit).

Customer discovery is the practice of testing assumptions about the problem *before* committing to a solution. Talk to at least 5 potential users. Ask open-ended questions: “Walk me through the last time this problem occurred.” “What did you try? Why didn’t it work?” Document answers; they become the foundation of your ConOps.

Synthesis: After interviews, group raw needs into clusters: (1) critical needs (mentioned by most personas, high impact) become SHALL requirements; (2) important needs (mentioned

by some, moderate impact) become SHOULD goals; (3) nice-to-have needs (mentioned rarely) become COULD features. This maps directly to MoSCoW prioritization.

Patent Landscape Search

Search Google Patents and USPTO for 2–3 relevant utility patents. Document: patent number, title, assignee, key claims, and a freedom-to-operate assessment (does your concept infringe any active claims?). This is not a legal opinion; it is an engineering awareness exercise that reveals prior art and protected design approaches.

Standards and Regulatory Landscape

Identify applicable standards early: UL/CSA for electrical safety, FCC Part 15 for radiated emissions, IP ratings for environmental protection, food-contact regulations if applicable. Standards become non-functional requirements. Discovering a regulatory requirement at CDR (instead of PDR) causes expensive redesign.

AI-Assisted Customer Persona Development

AI tools can help you *prepare for* real stakeholder interviews, not replace them. Before meeting real users, describe the problem domain and ask the AI to role-play as different stakeholder types: the primary user, the maintainer, the regulator, the adjacent-system owner. Conduct the same interview questions you would use with real stakeholders. The purpose is to **sharpen your questions and expose blind spots**, so that your limited time with real people is more productive.

AI Personas Are Practice, Not Data

AI-generated personas are hypothesis generators, not validated needs. An AI “greenhouse operator” has never actually lost a crop to frost. Treat AI responses as plausible questions to ask, not as answers to rely on. Your SDRD requirements must trace to real stakeholder input (interviews, observation, published user studies), not to AI conversations. Reviewers will ask where your needs came from; “I asked ChatGPT” is not a defensible answer.

Document each persona interview with: persona description, key quotes, identified needs (stated and unstated), and implications for requirements. Label AI-generated content explicitly in your engineering notebook so you can distinguish it from validated data.

The Mom Test^a

^aR. Fitzpatrick, *The Mom Test*, Robfitz, 2013.

Never ask “Would you use this?”; people say yes to be polite. Instead ask about *past behavior*: “When did you last face this problem? What did you do?” Past actions predict future behavior far better than hypothetical opinions.

1.1.3 Problem vs. Solution Thinking

Engineers instinctively jump to solutions. The discipline of problem definition requires staying in the **problem space** (what is wrong, who is affected, what the consequences are) before entering the **solution space** (technologies, components, architectures).

Critical test: Can someone read your statement and propose a fundamentally different solution? If yes, you are in the problem space. If the statement implies only one approach, you have prematurely entered the solution space.

Solution Masquerading as a Problem

Bad: “We need to build an Arduino-based temperature monitoring system with a solar panel.” This specifies technology before the problem is understood. **Good:** “Greenhouse operators cannot maintain optimal growing temperatures in unheated hoop houses during Colorado Front Range winters, resulting in \$12,000/season crop losses.” This is solution-agnostic.

1.1.4 The Five-Element Framework

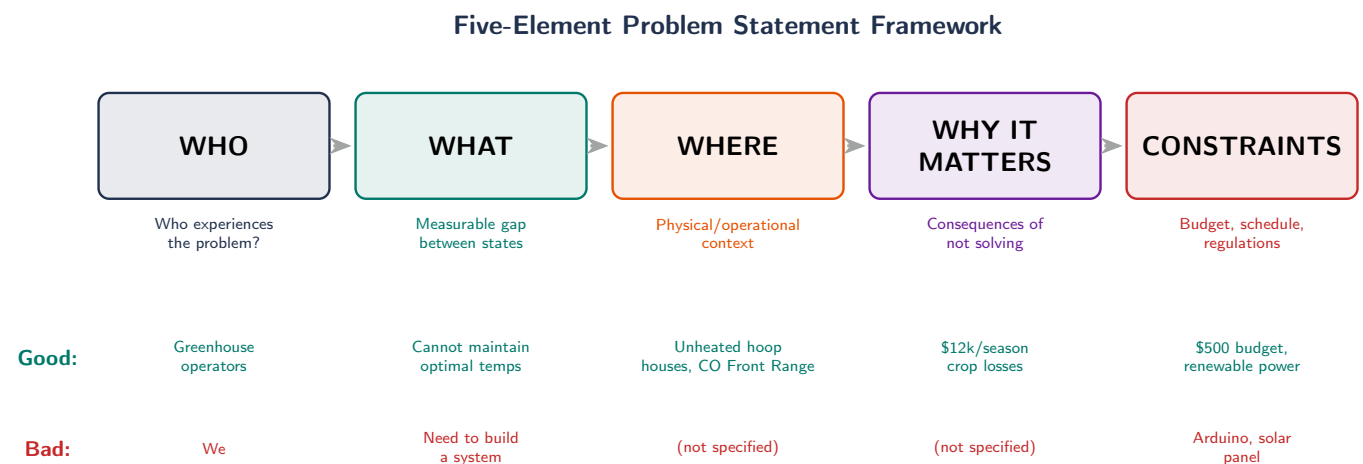


Figure 1.1: Five-element problem statement framework with good and bad examples.

1. **WHO:** Who experiences the problem? Be specific about roles and context.
2. **WHAT:** What is the measurable gap between current and desired states?
3. **WHERE:** In what physical, operational, or environmental context?
4. **WHY IT MATTERS:** What are the consequences of not solving it? (Cost, safety, health.)
5. **CONSTRAINTS:** What boundaries limit the solution? (Budget, schedule, technology, regulations.)

Constraints directly become non-functional requirements. The “why it matters” element justifies the project’s existence.

1.1.5 Scope and Boundaries

Scope defines what is inside and outside the project. Without it, scope creep expands the project indefinitely. The **system boundary** distinguishes what you build from what the environment provides. Capture this in a **context diagram**: your system as a single block with all external interfaces (power, data, user, mechanical, environmental) labeled.

1.2 Stakeholder Engagement

1.2.1 Identifying Stakeholders

A stakeholder is anyone who affects, is affected by, or has an interest in the project. **Missing a stakeholder = missing requirements = integration surprises.**



Figure 1.2: Power/interest grid with engagement strategies and nine-category identification checklist.

Nine categories to check: primary users, beneficiaries, sponsor/funder, maintainers, regulators, integrators (adjacent systems), adversaries, trainers, and disposal handlers. Students consistently miss maintainers, regulators, and adjacent-system owners.

1.2.2 The Power/Interest Grid

Classify each stakeholder by their power (ability to influence the project) and interest (how much they care). Four quadrants with distinct strategies: **Manage closely** (high power, high interest, key decision makers), **Keep satisfied** (high power, low interest), **Keep informed** (low power, high interest, valuable input), **Monitor** (low power, low interest).

1.2.3 Needs Elicitation Techniques

Use **2–3 methods**; single-method elicitation misses needs. Strongest combination for engineering projects: **interviews** (capture stated needs) + **observation** (capture unstated needs) + **prototyping** (validate assumptions).

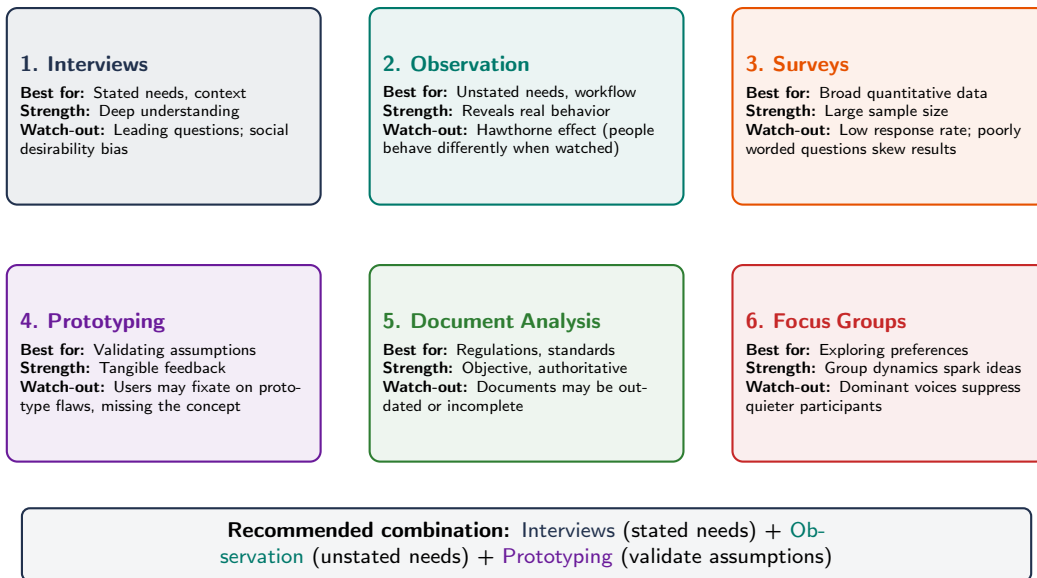


Figure 1.3: Six elicitation methods with best-use cases, strengths, and watch-outs.

The Best Question in Engineering

End every stakeholder interview with: “What should I have asked that I didn’t?” This single question catches more missing requirements than any structured technique.

1.2.4 Managing Conflicting Needs

Conflicts are normal. Prioritize using **MoSCoW**: Must (becomes **SHALL**), Should (becomes **SHOULD**), Could (if budget allows), Won’t (explicitly excluded). When conflicts are irreconcilable, the sponsor or primary user has final authority, but the decision must be documented.

1.2.5 Needs vs. Wants

A **need** is something without which the system fails its mission, it becomes a **SHALL** requirement. A **want** enhances value but the system works without it, it becomes a **SHOULD** goal pursued only if resources allow. Separate them early.

1.3 From Problem to ConOps

1.3.1 Concept of Operations

The ConOps describes how the system will be used from the stakeholder’s perspective, written in **plain language** (no engineering jargon). Contents: system mission, current situation, operational scenarios (normal, degraded, emergency, maintenance), user roles, environmental conditions, external interfaces, and performance expectations.

The ConOps is the document the stakeholder reviews and approves **before requirements are written**. Misunderstandings caught here cost minutes; caught in requirements, they cost weeks.

1.3.2 Use Cases and Operational Scenarios

A **use case** describes a specific user-system interaction to achieve a goal. An **operational scenario** extends the use case into a realistic narrative with timing, weather, user state, and system state. Scenarios reveal requirements that abstract analysis misses.

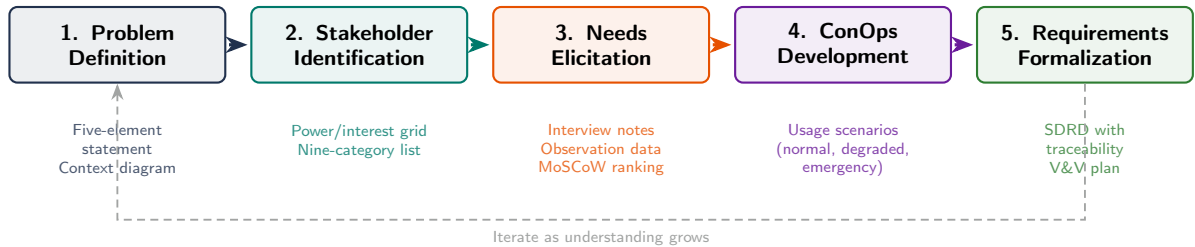


Figure 1.4: Five-stage needs-capture pipeline from problem to formal requirements with deliverables.

GS GreenShield: ConOps to Requirements Mapping

System: GreenShield. Automated Greenhouse Frost Protection System.

ConOps excerpt: “When nighttime temperature drops below 3°C, the system activates supplemental heating and alerts the operator via SMS. In degraded mode (cellular failure), the system continues autonomous heating control using local logic.”

Requirements derived:

- SYS-REQ-001: The system shall monitor air temperature continuously at ≥ 1 Hz.
- SYS-REQ-002: The system shall activate the heater when temperature $\leq$ user-defined setpoint (default 3°C).
- SYS-REQ-003: The system shall send an SMS alert within 60 s of heater activation.
- SYS-REQ-004: The system shall continue autonomous heating control during communication failure.

Every ConOps sentence should map to at least one requirement. The cost-of-change curve (Figure 3.1) shows why catching errors here, in the requirements phase, is far cheaper than catching them during integration. If it does not, either the ConOps is too vague or requirements are missing.

Industry SE Parallels

- **INCOSE SEH 4th ed.:** Sec. 4.1 (Stakeholder Needs and Requirements Definition), Sec. 4.2 (System Requirements Definition).
- **NASA SP-2016-6105 Rev 2:** Ch. 4 (System Design Processes), Sec. 4.1 (Stakeholder Expectations Definition).
- **IEEE 29148:** Standard for requirements engineering processes.

In industry, the ConOps is often a formal deliverable reviewed at a System Requirements Review (SRR), which precedes the Preliminary Design Review. The MoSCoW prioritization method maps to what INCOSE calls “needs prioritization” within stakeholder requirements definition.

1.4 Artifact Handoff: What Chapter 1 Produces

Artifact	Description	Forward Reference
Problem Statement	Five-element statement (who/what/where/why/constraints)	Ch. 2 (p. 12)
Stakeholder Map	Power/interest grid with engagement strategies	Ch. 2, Ch. 4 (validation)
Concept of Operations	Plain-language usage scenarios (normal, degraded, emergency)	Ch. 2, Ch. 6 (O&M)
Competitive Analysis	3–5 competing solutions with gap identification	Ch. 4 (concept generation)
Context Diagram	System boundary with external interfaces labeled	Ch. 2 (interface requirements)

1.5 Stakeholder Analysis Techniques

1.5.1 Stakeholder Identification

Begin by casting a wide net: who is affected by this product? Beyond the obvious end user, consider: the person who purchases it (may differ from the user), the person who installs and maintains it, the person who disposes of it, regulatory bodies that must approve it, adjacent systems that interact with it, and communities affected by its production and use.

For the GreenShield project, stakeholders typically include: the primary user (the person benefiting from the product's function), the course instructor (who evaluates it against educational objectives), the lab technician (who must store and maintain the prototype), future student teams (who may build on your work), and the environment (affected by material choices and energy consumption).

1.5.2 Stakeholder Interviews

Structured interviews are the most reliable method for understanding stakeholder needs. Prepare 5–8 open-ended questions before the interview:

“What is the biggest frustration you experience with [current process]?”

“Walk me through a typical day where this problem arises.”

“If you could change one thing about [existing solution], what would it be?”

“What would a perfect solution look like to you?”

“How much would you be willing to pay for that solution?”

Record the responses verbatim; do not paraphrase during the interview. Identify themes across multiple interviews: if three out of five stakeholders mention the same pain point, it is likely a real and significant need.

1.5.3 Needs vs. Wants vs. Constraints

Needs: requirements that must be met for the product to be useful. “The device must measure temperature.”

Wants: features that are desired but not essential. “It would be nice if it displayed the data on a phone app.”

Constraints: limitations imposed by physics, regulations, budget, or the environment. “It must run on 12 V battery power because there is no AC outlet at the installation site.”

Distinguishing these three categories during stakeholder analysis prevents over-engineering (spending resources on wants at the expense of needs) and scope creep (adding features that were never required).

1.6 Problem Statement Formulation

A well-written problem statement is the foundation of the entire project. It must be:

Specific: “Students in MEGN 300 lose 20 minutes per lab session troubleshooting DAQ wiring because the current breadboard setup has unreliable connections.” Not: “The lab setup is bad.”

Measurable: include quantified impact. “The current process wastes \$50,000/year in rework costs” or “15% of student groups fail to complete the lab in the allotted time.”

Solution-agnostic: describe the problem, not a solution. “The building’s HVAC system cannot maintain comfortable temperature in Room 201” (problem). Not: “Room 201 needs a space heater” (solution).

1.7 Competitive Benchmarking

Before designing a new product, study what already exists. For each competing product, document:

Features: what does it do? What are its specifications?

Price: what does it cost (purchase, installation, operation)?

Strengths: where does it excel?

Weaknesses: where does it fall short? (These are your opportunities.)

Market position: who buys it? Why?

Organize this information in a competitive comparison matrix. The matrix reveals the gap your product must fill: the unmet need at the intersection of user desire and competitor weakness.

1.8 Practice Questions

1. Rewrite this solution-as-problem into a proper five-element problem statement: “We need to design a solar-powered weather station using a Raspberry Pi.”
2. List all stakeholders for a MEGN 301 design project that designs an automated pet feeder for elderly users. Use the nine-category checklist.
3. You interview a greenhouse operator who says “I need a temperature display.” Through observation, you notice she checks the display, then walks to the heater and turns it on manually. What is her *actual* need? How does this change the requirements?
4. Classify these stakeholders on a power/interest grid for a campus parking management system: (a) university president, (b) daily commuter student, (c) campus police, (d) city traffic department, (e) disability services office.
5. Write a 1-paragraph ConOps for an automated frost protection system for a small greenhouse, including at least two operational scenarios (normal and degraded mode).

6. **Market analysis:** existing pet feeders on Amazon range \$30–\$150. Strengths: app connectivity, portion control. Weaknesses: require Wi-Fi (elderly users may lack technical skill), small hoppers (2–3 day capacity), no medication dispensing. Gap: a system designed specifically for elderly users with large-print controls, cellular connectivity (no Wi-Fi setup), and caregiver alerting.
7. **Customer discovery approach:** interview 3–5 elderly pet owners and 2–3 caregivers. Key questions: “Show me how you feed your pet today.” “What happens when you travel or are hospitalized?” “How comfortable are you with smartphone apps?” Expected insights: manual feeding is a daily routine that provides companionship (do not eliminate it entirely); missed feedings happen during health crises, not routine days; caregivers need alerts, not control.
8. A sponsor wants the system under \$500, the user wants a touchscreen display, and the maintainer wants modular components. These conflict. How do you resolve this using the MoSCoW framework?

1.8.1 Answers

1. “Small-farm operators [WHO] lack access to real-time, site-specific weather data [WHAT] at remote agricultural locations without grid power [WHERE], leading to crop management decisions based on regional forecasts that miss micro-climate variations by up to 5 ° C [WHY], within a \$300 budget using only renewable power sources [CONSTRAINTS].”
2. Primary users: elderly pet owner (operator). Beneficiaries: the pet; family members who worry remotely. Sponsor: the pet owner or their family. Maintainers: family member or home-care aide who refills food and troubleshoots. Regulators: food-safety standards for pet food storage; electrical safety (UL). Integrators: home Wi-Fi network; smartphone (if app-connected). Adversaries: the pet (may try to defeat the mechanism). Trainers: whoever teaches the elderly user. Disposal: recycling or trash service.
3. Her actual need is automated frost protection, not a temperature display. The display is her current workaround for a lack of automation. The real requirement is: “The system shall activate the supplemental heater when temperature drops below a user-defined setpoint.” The display becomes a secondary want, not the primary need. This changes the architecture from a passive monitor to an active control system.
4. (a) University president: high power, low interest → Keep Satisfied. (b) Commuter student: low power, high interest → Keep Informed. (c) Campus police: high power, high interest → Manage Closely (they enforce parking rules). (d) City traffic department: moderate power, low interest → Keep Satisfied (jurisdiction over surrounding roads). (e) Disability services: moderate power, high interest → Manage Closely (ADA compliance is non-negotiable).
5. “The frost protection system monitors greenhouse air temperature continuously. During normal operation, when temperature drops below the user-defined threshold (default 3 ° C), the system automatically activates the supplemental heater and sends an alert to the operator’s phone. The operator can acknowledge the alert, adjust the setpoint, or override the system remotely. In degraded mode (communication failure), the system continues autonomous control using local logic, activating the heater based on its last-known setpoint and logging events locally until communication is restored. The operator discovers the communication failure on their next manual check or when they notice the absence of the daily status report.”

6. MoSCoW: **Must**: system cost under \$500 (sponsor constraint; non-negotiable). Modular components for field replacement (maintainer need; essential for long-term usability). **Should**: visual status display (user need; but a low-cost LCD or LED indicators at \$5 may satisfy the need without a \$50 touchscreen). **Could**: touchscreen interface (user want; only if budget allows after must-haves). **Won't**: full-color graphical touchscreen in v1.0. Resolution: the display is a legitimate need but the *touchscreen* is a want. Propose a 2-line LCD (\$3) for v1.0 with a touchscreen upgrade path for v2.0 if the sponsor increases budget.

Chapter 2

Value Proposition & Triple Bottom Line

When to Read This Chapter

Sprint 1: Read Sections 2.1–2.3 (four-dimension framework, feasibility, desirability). These feed directly into your problem statement justification. **Sprint 2:** Read Section 2.5 (competitive analysis) before PDR. **Sprint 3:** Return to Section 2.4 (viability, cost analysis) when building your power budget and BOM.

2.1 Why Value Proposition Matters in Engineering Design

A technically brilliant product that nobody wants, nobody can afford, or nobody can sustain is a failure. Before investing hundreds of engineering hours in requirements, design, and fabrication, you must answer a fundamental question: **is this product worth building?** The value proposition framework (Figure 2.1) provides a structured method for answering that question by examining four dimensions: feasibility, desirability, viability, and sustainability. Together, these dimensions ensure that a product concept is not just technically possible, but genuinely valuable to its intended users and stakeholders.

In MEGN 301, your team’s GreenShield project must satisfy all four dimensions. A common failure mode in student projects is building something that works (feasible) but that nobody actually needs (not desirable), costs more than the benefit it provides (not viable), or creates environmental harm that outweighs its utility (not sustainable). The value proposition analysis forces you to confront these questions early, when the cost of pivoting is measured in whiteboard sessions rather than scrapped prototypes.

2.2 The Four-Dimension Value Framework

2.2.1 Feasibility: Can We Build It?

Feasibility asks whether the proposed solution is technically achievable within the constraints of the project. It encompasses:

Technical feasibility: do the enabling technologies exist? Can the required performance be achieved with available materials, components, and processes? For the GreenShield project, this means: can the sensors, actuators, and control algorithms deliver the specified performance within the physical constraints (size, weight, power)?

Resource feasibility: does the team have the skills, tools, budget, and time to execute the design? A concept requiring custom ASIC fabrication is technically feasible in industry but infeasible in a one-semester course. Assess honestly: what capabilities does your team have, and what gaps need to be filled through learning, collaboration, or scope reduction?

Manufacturing feasibility: can the product be fabricated with available processes? In MEGN 301, this typically means: 3D printing (FDM, SLA), laser cutting, manual machining, off-

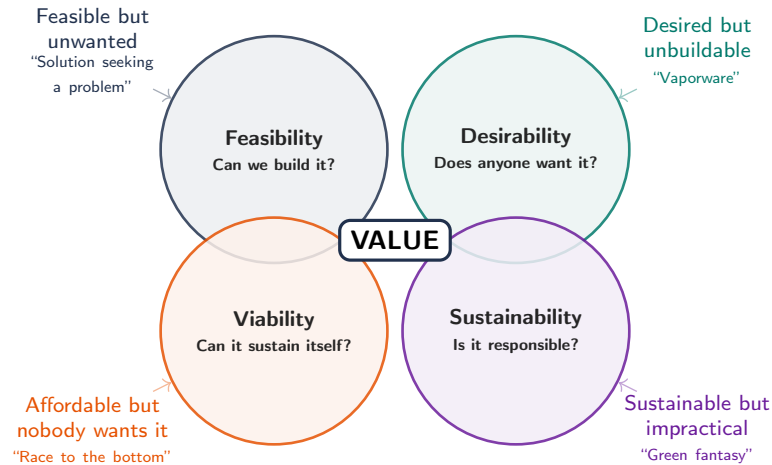


Figure 2.1: The four-dimension value framework. A product must satisfy all four dimensions to create genuine value. Meeting only one or two dimensions produces the characteristic failure modes shown at each corner. The center represents the intersection where feasibility, desirability, viability, and sustainability all align.

the-shelf electronics, and hand soldering. Designs requiring injection molding tooling (Chapter 17, p. 172), CNC 5-axis machining, or custom PCB fabrication with <6 mil traces may be infeasible within the project timeline.

Integration feasibility: can the subsystems be assembled into a working system? This is the most commonly underestimated dimension. Individual subsystems may work perfectly on the bench but fail when integrated due to mechanical interference, electromagnetic compatibility issues, thermal interactions, or software timing conflicts.

The Feasibility Trap

Students often confuse “theoretically possible” with “feasible within our constraints.” A solar-powered autonomous boat is theoretically possible, but designing and building one in 15 weeks with a \$500 budget and no prior experience in waterproofing, marine electronics, or solar power systems is not feasible. Scope your feasibility assessment to your actual resources, not to what a well-funded company could achieve.

A concrete MEGN 301 example: designing a custom 4-layer PCB with impedance-controlled traces and 0201 components is technically possible, but for a first prototype, a 2-layer board with 0805 passives and a verified ESP32 DevKit module is the feasible choice. The custom board might perform 5% better; the DevKit ships tomorrow and works on the first try.

2.2.2 Desirability: Does Anyone Want It?

Desirability asks whether the proposed solution addresses a genuine need, want, or pain point for its intended users. It encompasses:

User need validation: have you spoken to actual users (not just imagined their needs)? What problem do they currently face? How do they currently solve it (or work around it)? What would make their life measurably better?

User experience: is the product intuitive to use? Does it fit into the user’s existing workflow or environment? A technically superior product with a confusing interface or an inconvenient form factor will not be adopted.

Emotional and aesthetic appeal: beyond functional performance, does the product look and feel appropriate for its context? Industrial equipment should convey ruggedness and reliability. Consumer products should be attractive and approachable. Medical devices should communicate precision and cleanliness.

Differentiation: how is this solution better than existing alternatives? If the answer is “it isn’t, really,” the product lacks a compelling value proposition regardless of its technical merit.

In MEGN 301, desirability is assessed through stakeholder interviews, user surveys, competitive benchmarking, and design reviews. The Preliminary Design Review (PDR) should include a clear articulation of who the user is, what problem they face, and why your solution is better than the alternatives.

2.2.3 Viability: Can It Sustain Itself Economically?

Viability asks whether the product can be produced and sold at a price that covers costs and generates value. It encompasses:

Cost analysis: what does it cost to produce one unit? Include materials (BOM cost), labor (assembly time $\times$ labor rate), overhead (equipment, facilities, testing), and amortized tooling (mold costs, fixtures). For MEGN 301 prototypes, the relevant question is: does the prototype cost fit within the project budget?

Revenue model: how does the product generate value? For commercial products: unit sales, subscriptions, service contracts, consumables. For internal tools: labor savings, quality improvement, safety enhancement. For the GreenShield project: quantify the value delivered in concrete terms (hours saved, energy reduced, safety incidents prevented).

Market sizing: how many units could realistically be sold? A brilliant niche product with 50 potential customers worldwide may not justify the development investment.

Competitive pricing: what do comparable products cost? Your product must deliver comparable or superior value at a competitive price point, or deliver significantly more value at a premium price.

2.2.4 Sustainability: What Are the Long-Term Consequences?

Sustainability asks whether the product’s benefits outweigh its environmental and social costs over its complete life cycle, from raw material extraction through manufacturing, use, and end-of-life disposal. It encompasses:

Environmental impact: what materials are used, and what are their extraction and processing impacts? How much energy does manufacturing consume? What emissions (CO₂, VOCs, particulates) does the product generate during use? Can the product be recycled, refurbished, or safely disposed of at end of life?

Social impact: does the product improve or degrade quality of life for its users and the communities affected by its production and disposal? Does the supply chain involve ethical labor practices? Are materials sourced from conflict-free regions?

Longevity and durability: a product that lasts 10 years and is repairable is more sustainable than one that lasts 2 years and must be discarded. Design for durability, repairability, and upgradability reduces the life-cycle environmental footprint per unit of service delivered.

Circular economy alignment: can the product be designed for disassembly, material recovery, and component reuse? Can a take-back program return end-of-life products to the manufacturer for refurbishment or recycling?

2.3 The Triple Bottom Line

The Triple Bottom Line (TBL) framework, introduced by John Elkington in 1994, evaluates organizational and product performance across three dimensions:

2.3.1 People (Social)

The social dimension encompasses the product's impact on all human stakeholders: users, workers, communities, and society at large. Key questions: does the product improve safety, health, or quality of life? Does it create or eliminate jobs? Are workers in the supply chain treated fairly and paid equitably? Does the product respect cultural values and community standards? Is it accessible to people with disabilities?

For the GreenShield project, the social dimension includes: user safety (the product must not create hazards), ergonomics (the product must be comfortable and intuitive to use), and community impact (the product should not generate noise, pollution, or other nuisances for neighbors).

2.3.2 Planet (Environmental)

The environmental dimension encompasses the product's impact on natural systems across its entire life cycle. Key questions: what is the carbon footprint of production and use? What resources (water, energy, rare materials) are consumed? What waste and emissions are generated? Can the product be recycled or biodegraded at end of life?

Life Cycle Assessment (LCA) is the formal methodology for quantifying environmental impacts. A simplified LCA for MEGN 301 projects should consider: (1) materials – mass and type (metals, plastics, electronics, batteries), (2) manufacturing energy (3D printing, machining, soldering), (3) use-phase energy consumption (if powered), (4) transportation (shipping weight and distance), and (5) end-of-life fate (landfill, recycling, hazardous waste).

Quick Environmental Impact Estimation

For a rough LCA in MEGN 301: (1) Sum the mass of each material category. (2) Multiply by the embodied energy factor: aluminum ≈ 170 MJ/kg, steel ≈ 25 MJ/kg, ABS plastic ≈ 95 MJ/kg, copper (PCB) ≈ 60 MJ/kg, lithium battery ≈ 400 MJ/kg. (3) Sum to get total embodied energy. (4) Add use-phase energy ($P_{\text{watts}} \times t_{\text{hours}} \times 0.0036$ MJ/Wh $\times$ lifetime years). (5) Compare to the energy saved or the environmental benefit delivered by the product.

2.3.3 Profit (Economic)

The economic dimension ensures the product is financially sustainable. A product that loses money on every unit sold, no matter how socially and environmentally beneficial, cannot persist. The economic dimension includes: unit manufacturing cost, expected selling price, gross margin ($\geq 40\%$ for hardware products to cover overhead, distribution, and warranty), development cost recovery (break-even volume), and total cost of ownership for the user (purchase price + operating costs + maintenance + disposal).

2.4 Applying TBL to the GreenShield Project

Every GreenShield project should include a TBL assessment in the Preliminary Design Review:

People: who benefits? Who is affected? What safety considerations exist? How does the product improve the user's daily experience?

Planet: what is the estimated embodied energy? What is the use-phase energy consumption? What materials require special disposal (batteries, electronics)? Can the product be disassembled for recycling?

Profit: what is the estimated BOM cost? What would a realistic selling price be? How many units would need to be sold to recover the development investment? What is the user's return on investment (cost savings or productivity improvement)?

The TBL analysis does not require precise numbers at the concept stage. Order-of-magnitude estimates are sufficient to identify show-stoppers (e.g., a product that costs \$500 to build but competes with \$20 alternatives, or a product whose battery contains 5 kg of lithium that cannot be recycled).

GS GreenShield Template: Value Proposition Summary

Include in your PDR package:

- **User:** Who is the primary user? What is their unmet need?
- **Solution:** What does our product do, in one sentence?
- **Feasibility:** What is the highest-risk technical element? How will we mitigate it?
- **Desirability:** What evidence do we have that users want this? (Interview quotes, survey data, competitive gap.)
- **Viability:** Estimated BOM cost vs. target price. Estimated market size.
- **Sustainability:** Estimated embodied energy. End-of-life plan. Key environmental risks.
- **TBL Summary:** One sentence each for People, Planet, Profit impact.

2.5 Value Proposition Canvas

The Value Proposition Canvas (Osterwalder & Pigneur) is a visual tool that maps the product's features to the customer's needs:

Customer Profile (right side): Jobs (what the customer is trying to accomplish), Pains (obstacles, risks, frustrations), Gains (desired outcomes, benefits, aspirations).

Value Map (left side): Products & Services (what you offer), Pain Relievers (how your product addresses each pain), Gain Creators (how your product delivers each gain).

A strong value proposition achieves **fit**: every significant customer pain is addressed by a pain reliever, and every important gain is delivered by a gain creator. Unaddressed pains represent risks; undelivered gains represent missed opportunities.

2.6 Stakeholder Value Mapping

Different stakeholders derive different value from the same product. A comprehensive value proposition addresses each stakeholder's perspective:

End user: functional value (does it solve my problem?), experiential value (is it pleasant to use?), and economic value (does it save me money or time?).

Buyer (may be different from user): economic value (ROI, payback period), risk reduction (warranty, reliability, vendor stability), and compliance value (meets regulatory requirements, company standards).

Installer/maintainer: ease of installation, ease of maintenance, availability of spare parts, quality of documentation.

Society: environmental impact, safety, job creation/displacement, resource consumption.

For the GreenShield project, map each stakeholder's needs to specific product features and quantify the value delivered where possible.

2.7 Business Model Canvas Integration

The Value Proposition Canvas fits within the larger Business Model Canvas (BMC), which describes the complete business logic:

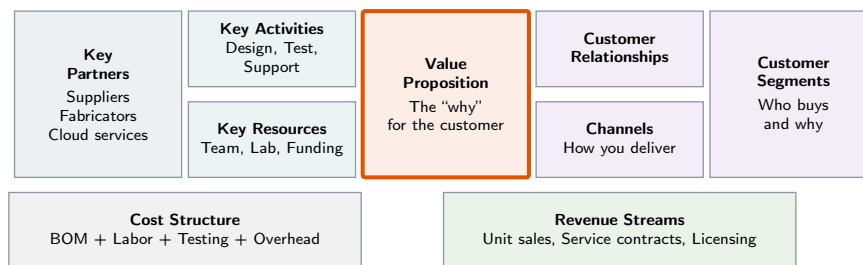


Figure 2.2: Simplified Business Model Canvas. The Value Proposition (highlighted) sits at the center, connecting what the team builds (left: partners, activities, resources, costs) to who benefits (right: customers, relationships, channels, revenue). In MEGN 301, the left side is your engineering process; the right side is your stakeholder analysis.

Key Partners: who supplies critical components, expertise, or distribution? For a GreenShield product: PCB fabricators, sensor suppliers, cloud service providers.

Key Activities: what must the team do to deliver the value proposition? Design, fabrication, testing, user support.

Key Resources: what assets are needed? Engineering talent, lab equipment, funding, IP.

Cost Structure: what are the major costs? BOM, labor, testing equipment, certification.

Revenue Streams: how does the product generate income? Unit sales, service contracts, data subscriptions, licensing.

While MEGN 301 is not a business course, understanding the BMC (Figure 2.2) ensures your technical design fits within a viable business context. A product that cannot be profitably produced, distributed, and supported will not survive regardless of its technical merit.

2.8 Competitive Analysis Framework

2.8.1 Direct Competitors

Products that solve the same problem in the same way. Compare feature-by-feature: performance, price, size, weight, reliability, user experience. Identify where your product is superior (the competitive advantage) and where it lags (the competitive disadvantage that must be accepted or mitigated).

2.8.2 Indirect Competitors

Products that solve the same problem in a different way, or products that make the problem less important. For a portable water purifier: indirect competitors include bottled water, municipal treatment upgrades, and UV sterilization pens. Understanding indirect competitors reveals the true competitive landscape and the user's switching cost.

2.8.3 Non-Consumption

The most powerful competitor is often “doing nothing.” If the user is currently tolerating the problem without any solution, your product must overcome the inertia of non-consumption. The value proposition must be compelling enough that the user is willing to change their behavior, invest money, and learn a new system.

2.9 Quantifying Value: The Business Case

For B2B (business-to-business) products, the value proposition must be quantified financially:

Cost savings: “Our sensor reduces calibration time from 2 hours to 15 minutes. At \$75/hour for a technician with 50 sensors calibrated annually: $\text{savings} = 50 \times 1.75 \times \$75 = \$6,563/\text{year}$.”

Revenue increase: “Our quality monitoring system reduces defect rates from 2% to 0.5%, recovering \$15,000/year in scrapped product.”

Risk reduction: “Our safety interlock prevents the failure mode that caused \$200,000 in damage last year. Even at 10% probability: $\text{expected savings} = \$20,000/\text{year}$.”

Payback period: the time for cumulative savings to exceed the product cost. For B2B products: <2 years is attractive, <1 year is compelling, >3 years is difficult to justify.

For B2C (business-to-consumer) products, value is often experiential rather than financial, and quantification is harder. Use: willingness-to-pay surveys, comparable product pricing, and customer satisfaction metrics.

2.10 Practice Questions

1. Your team proposes a solar-powered phone charging station for campus bus stops. Evaluate feasibility, desirability, viability, and sustainability. Identify the highest-risk dimension and propose a mitigation strategy.
2. A GreenShield team's product costs \$120 in materials and takes 8 hours to assemble. The product saves the user approximately \$15/month in energy costs. Calculate the payback period. Is this viable for a consumer market? For an industrial market?
3. Conduct a simplified TBL assessment for a battery-powered portable fan. List two considerations for each dimension (People, Planet, Profit).
4. Explain why a product can be feasible and desirable but not viable. Give a specific example from consumer electronics.
5. Your product uses a 5000 mAh lithium-polymer battery. Research the environmental impact of lithium mining and battery disposal. Propose a design change that would improve the product's sustainability score.

2.10.1 Answers

1. Feasibility: solar panel + charge controller + USB outlets is proven technology; risk is weather variability and vandalism. Desirability: students waiting at bus stops often have low phone battery; validated by informal survey. Viability: \$200 BOM, no revenue model unless campus funds it; viable as a university sustainability project, not as a commercial product. Sustainability: solar energy is renewable; battery and electronics require recycling; aluminum enclosure is recyclable. Highest risk: viability (no clear revenue); mitigation: frame as a grant-funded campus improvement, not a commercial product.
2. Payback = $\$120/\$15 = 8$ months. For consumer: borderline viable (consumers prefer <6-month payback for energy products). For industrial: excellent (industrial buyers accept 1–3 year payback). Viability depends on the target market, not just the technology.
3. People: (+) provides comfort in hot environments, (–) battery creates disposal hazard. Planet: (+) reduces need for large HVAC, (–) battery and motor require mining and energy-intensive manufacturing. Profit: (+) low BOM (\$15–30), (–) intense competition from established brands drives margins below 20%.
4. A product can be technically buildable (feasible) and users may want it (desirable), but if it costs more to produce than customers will pay, it is not viable. Example: modular smartphones (Project Ara) were technically feasible and highly desired by tech enthusiasts, but the modular design added \$100+ to manufacturing cost, making the price uncompetitive with integrated designs.
5. LiPo batteries contain lithium (from brine extraction in Chile/Argentina, high water consumption), cobalt (from DRC, ethical concerns), and electrolytes (flammable, toxic). Design change: reduce battery capacity to 2500mAh (sufficient for the application) and design the enclosure for easy battery replacement and return. Add a QR code linking to a battery recycling drop-off locator. Switch from cobalt-based to lithium iron phosphate (LFP) chemistry, which is cobalt-free and more recyclable.

Chapter 3

Writing Requirements & Defining Architecture

When to Read This Chapter

Sprint 1: Read Sections 3.1–3.3 (SDRD, requirements writing, quality checklist). These are your primary Sprint 1 deliverables. **Sprint 2:** Read Sections 3.4–3.6 (functional decomposition, trade studies, FMEA) before PDR. **Sprint 3:** Return to Section 3.5 (interface definition) when writing ICDs for CDR.

Requirements are the single most important deliverable in the design process. Getting requirements right costs minutes; fixing wrong requirements costs weeks. The cost-of-change curve illustrates why.

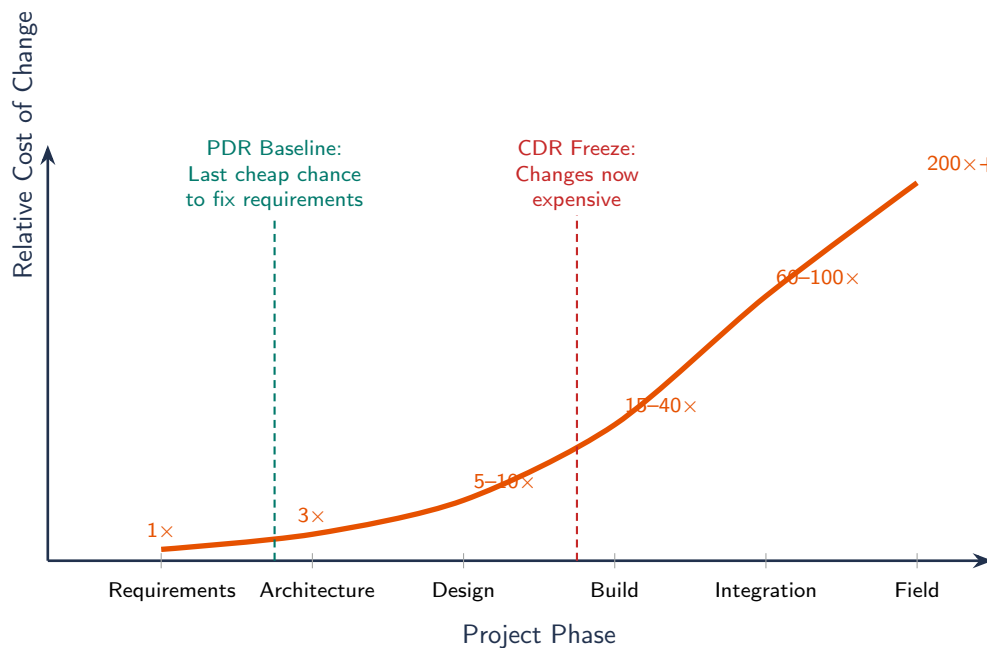


Figure 3.1: Cost of change increases exponentially with project phase. A requirement error caught during requirements writing costs 1x; the same error caught during integration costs 60–100x. This is why the quality checklist below matters.

3.1 Writing Requirements

3.1.1 The System Design Requirements Document (SDRD)

The SDRD is the single authoritative document that captures all system-level requirements, their rationale, traceability, and verification methods. Standard structure:

1. **Introduction:** system overview, scope, applicable documents.
2. **System Description:** ConOps summary, system boundary, external interfaces.
3. **Requirements:** organized by functional, performance, and non-functional categories. Each includes: unique ID, shall-statement, rationale, parent traceability, verification method (T/A/I/D).
4. **Traceability Matrix:** forward (need $\rightarrow$ requirement) and backward (requirement $\rightarrow$ need).
5. **Verification Plan Summary:** method assignments and planned test phases.

The SDRD is a living document; baselined at PDR and updated under change control through CDR and FDR.

3.1.2 What Is a Requirement?

A requirement is a formal statement of a capability or constraint the system must satisfy. It uses the word **shall** to indicate mandatory compliance. “The system *shall* measure ambient temperature” is a requirement. “Should” and “will” are non-binding.

A well-formed requirement has: (1) a unique ID (SYS-REQ-042) for traceability, (2) a single shall-statement with one testable condition, (3) measurable acceptance criteria with units and tolerance, (4) a rationale explaining why it exists, and (5) a verification method (Test, Analysis, Inspection, or Demonstration).

Common Failures

Vague: “shall be fast.” **Compound:** “shall measure AND display.” **Untestable:** “shall be user-friendly.” **Gold-plated:** “shall use an Arduino” (specifies implementation, not need). Each failure produces downstream rework.

3.1.3 Requirement Types

- **Functional:** What the system shall *do*. Actions: sense, compute, transmit, actuate, display. “The system shall measure ambient temperature.”
- **Performance:** How *well* it does it. Quantitative criteria. “...with accuracy of $\pm 0.5^\circ \text{C}$ over -20 to $+60^\circ \text{C}$ at $\geq 1 \text{ Hz}$.” A functional requirement without a paired performance requirement is incomplete.
- **Non-Functional (NFR):** Constraints on *how* it is built. Environmental (“operate at -20 to $+60^\circ \text{C}$ ”), regulatory (“comply with FCC Part 15”), material (“RoHS-compliant”), reliability (“ ≥ 1000 hrs MTBF”), interface (“CAN 2.0B”).

3.1.4 Requirements Hierarchy and Traceability

Requirements decompose top-down: stakeholder needs $\rightarrow$ system requirements $\rightarrow$ subsystem requirements $\rightarrow$ component specifications. Each level adds detail. Every child requirement traces to a parent (backward traceability). Every parent is decomposed into children (forward traceability). An orphan requirement has no justification; an undecomposed requirement has no implementation.

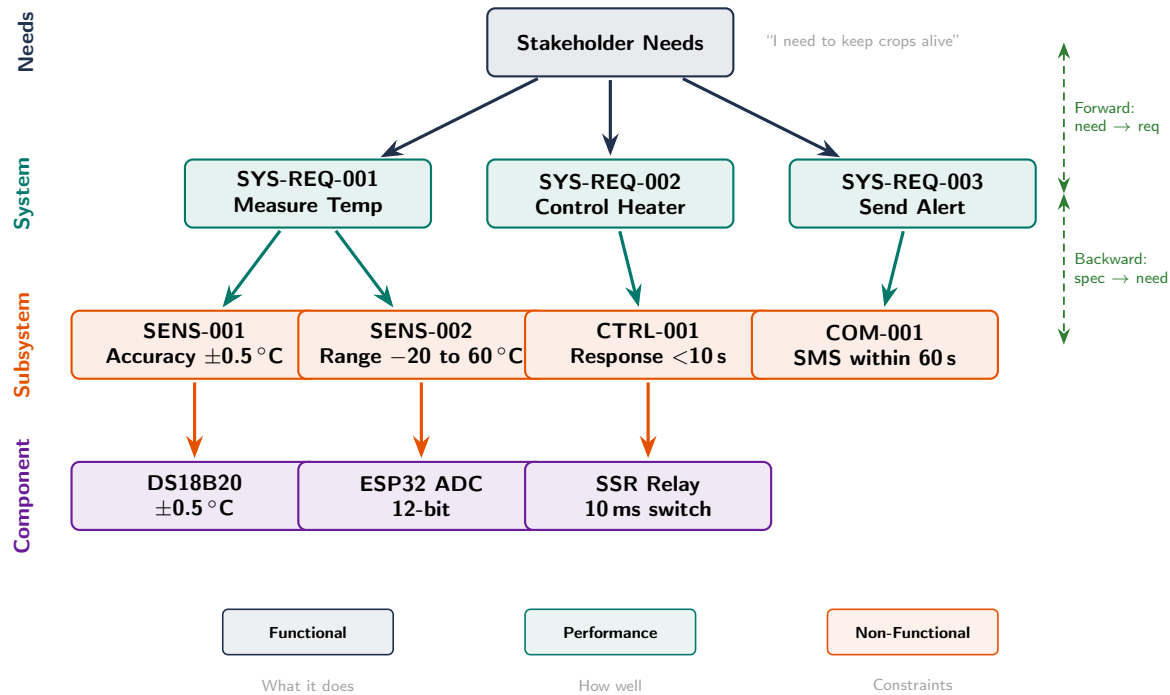


Figure 3.2: Requirements hierarchy from stakeholder needs through system, subsystem, and component levels with traceability and three requirement types.

3.1.5 The Requirements Quality Checklist

Before baselining any requirement, verify it passes all seven tests:

1. **Necessary:** does it trace to a stakeholder need? (No gold-plating.)
2. **Unambiguous:** can it be interpreted only one way?
3. **Complete:** does it include all conditions (range, tolerance, units)?
4. **Consistent:** does it conflict with any other requirement?
5. **Verifiable:** can you write a test procedure for it right now?
6. **Traceable:** does it have a unique ID and parent link?
7. **Implementation-free:** does it specify WHAT, not HOW?

A requirement that fails any test must be rewritten before PDR.

3.1.6 Writing Good Requirements: Five Rules

1. **One shall per requirement:** compound requirements are untestable.
2. **Quantitative criteria:** numbers, units, tolerance. "Fast" is not a requirement.
3. **Testable as written:** if you cannot envision a test procedure, rewrite.
4. **Implementation-free:** state WHAT, not HOW. " ≥ 12 analog inputs" not "use Arduino Mega."
5. **Include rationale:** WHY does this requirement exist? Prevents zombie requirements.

3.2 Defining Architecture

3.2.1 Functional Decomposition

Define WHAT the system must do (functions) before deciding HOW (physical components). Break the mission into primary functions, sub-functions, and leaf functions. Each leaf maps to a physical component. Use verb-noun naming: Sense Temperature, Process Data, Provide Power.

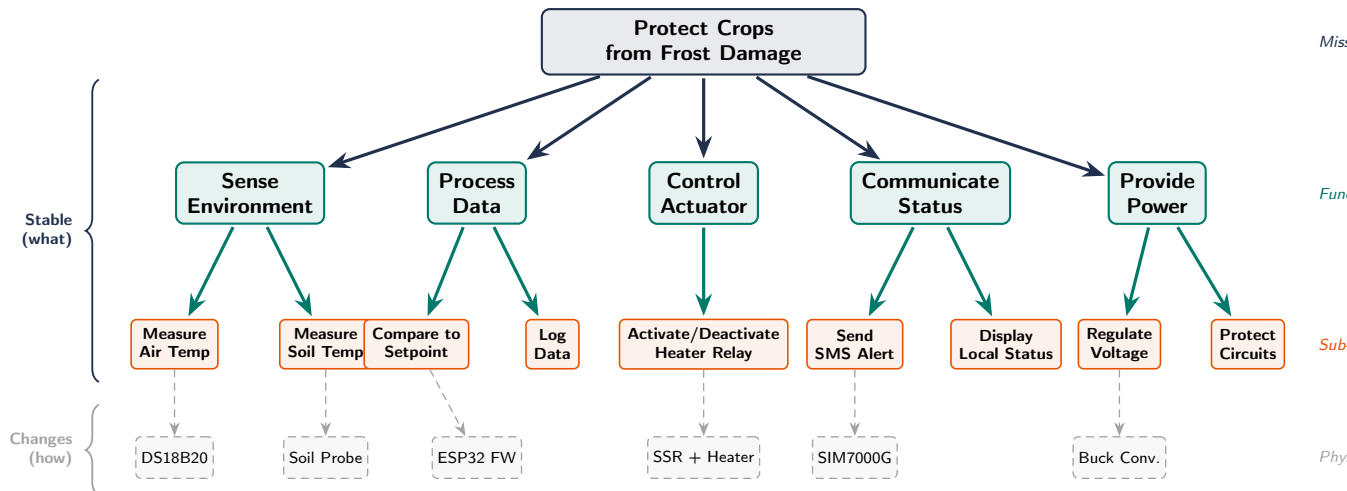


Figure 3.3: Functional decomposition: functions defined before physical allocation. Functions are stable; implementations change.

3.2.2 From Functions to Architecture Blocks

Functional decomposition produces a tree of functions (verb-noun). Architecture allocation maps each leaf function to a physical module. This mapping is one-to-many (one function may require multiple modules) and many-to-one (one module may implement multiple functions). Document this in an **allocation matrix**: rows = functions, columns = physical modules, cells = primary/supporting role.

The allocation matrix reveals integration complexity: modules that implement many functions are high-risk single points of failure.

3.2.3 Interface Definition

Once functions map to modules, define interfaces between them with Interface Control Documents (ICDs): what crosses the boundary (signals, power, data), in what format (protocols, voltage levels), with what constraints (bandwidth, latency).

Start Your FMEA During Architecture: Not at FDR

Failure Mode and Effects Analysis (FMEA) is most valuable *before* hardware is ordered. As you allocate functions to physical modules (this chapter), immediately ask: “What happens if this module fails?” Chapter 3 (p. 20) provides the full FMEA framework and a worked GreenShield example. Performing FMEA during architecture (Chapters 2–3) enables you to

design out critical failure modes. Waiting until O&M (Chapter 7, p. 66) turns FMEA into an autopsy on a failed project.

3.2.4 Architecture Patterns

Centralized: one controller. Simple but single point of failure. **Distributed:** multiple controllers on a bus (CAN, I²C). More robust, enables parallel development. **Hierarchical:** layered control. Choose based on reliability, complexity, and team structure.

3.2.5 Requirements–Architecture Iteration

Requirements and architecture co-evolve. Initial requirements drive a candidate architecture. That architecture reveals implementation constraints (weight, power, thermal) that feed back as new requirements or modified performance targets. This iteration is normal and healthy, but must be **documented**. Each iteration produces a revised SDRD version with change rationale.

Convergence test: when two consecutive iterations change fewer than 5% of requirements, the design is stable enough to baseline at PDR.

3.3 Environmental and Regulatory Constraints

Every system operates in an environment that imposes constraints. Systematically address:

- **Temperature:** operating range, storage range, thermal shock.
- **Moisture/IP:** splash (IPX4), rain (IPX5), submersion (IPX7/8).
- **Vibration/shock:** transport, operational, handling drop height.
- **EMC:** FCC Part 15 Class B (residential), IEC 61000 (EU).
- **Safety:** UL/CSA/CE marking, OSHA, university lab safety policies.
- **Materials:** RoHS compliance, California Prop 65, food-contact (FDA 21 CFR).

3.4 Accessibility and Universal Design

Accessibility is a non-functional requirement category that is easy to overlook and expensive to retrofit. Universal Design, a framework developed by Ronald Mace at NC State University, holds that products and environments should be usable by *all people*, to the greatest extent possible, without the need for adaptation or specialized design. The seven principles of Universal Design provide a practical checklist for evaluating your design:

1. **Equitable Use:** the design is useful and marketable to people with diverse abilities. No user group is segregated or stigmatized by the design.
2. **Flexibility in Use:** the design accommodates a wide range of individual preferences and abilities, including right- or left-handed use and adaptable pacing.
3. **Simple and Intuitive Use:** the design is easy to understand regardless of the user's experience, knowledge, language skills, or concentration level.
4. **Perceptible Information:** the design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities. Use redundant modes (visual + audible + tactile).
5. **Tolerance for Error:** the design minimizes hazards and adverse consequences of accidental or unintended actions. Fail-safe features and confirmation steps reduce risk.

6. **Low Physical Effort:** the design can be used efficiently and comfortably with minimum fatigue. Controls require reasonable force and allow a neutral body position.
7. **Size and Space for Approach and Use:** appropriate size and space is provided for approach, reach, manipulation, and use regardless of the user's body size, posture, or mobility aid.

3.4.1 Regulatory Requirements

For products with **physical interfaces** (controls, handles, displays, enclosures), the Americans with Disabilities Act (ADA) establishes minimum accessibility standards. ADA compliance is not optional for products deployed in public spaces, educational institutions, or workplaces. Key considerations include: reach ranges, operating force limits, tactile indicators, and clearances for wheelchair access.

For products with **digital interfaces** (screens, apps, web dashboards), the Web Content Accessibility Guidelines (WCAG 2.1) define three conformance levels (A, AA, AAA). Level AA is the most common compliance target and covers: text alternatives for non-text content, keyboard navigability, sufficient color contrast ($\geq 4.5:1$ for normal text), and screen reader compatibility.

Accessibility Quick-Check

Before PDR, ask these three questions about your design:

- **Limited mobility:** Can the product be operated with one hand? Can all controls be reached from a seated position? Do any controls require more than 5 lbf of force?
- **Limited vision:** Are all critical indicators available in at least two sensory modes (e.g., visual + audible)? Is text ≥ 12 pt? Does the display maintain $\geq 4.5:1$ contrast ratio?
- **Limited hearing:** Are all audible alerts accompanied by a visual indicator? Are instructions available in text form rather than audio only?

If the answer to any question is “no,” write a requirement to address it. Accessibility requirements are testable: “The system shall be operable with one hand and require no more than 5 lbf of actuation force on any control” is a well-formed SHALL statement.

3.5 Trade Studies and Decision Making

3.5.1 The Trade Study Process

1. Define the decision and evaluation criteria (derived from requirements). Weight criteria (sum to 1.0).
2. Identify 3–5 viable candidates. No straw-men.
3. Rate each option against each criterion using data, not preference.
4. Score (raw $\times$ weight), sum, select. Check sensitivity: if ± 0.05 weight change flips the winner, investigate further.
5. Document everything: matrix, data sources, weight rationale, dissenting opinions.

3.5.2 Pugh Matrix vs. Weighted Decision Matrix

Pugh (screening): rate options as +, S, – against a datum. Fast. No weights. Eliminates clearly inferior options. **Weighted matrix (selection):** numerical ratings $\times$ weights. Rigorous, quantitative, traceable. Best practice: Pugh to screen $\rightarrow$ weighted matrix to select.

Weighted Decision Matrix: Motor Selection

Criterion	Weight	Stepper	DC Gear	Servo
Torque	0.25	4 (1.00)	5 (1.25)	3 (0.75)
Speed	0.15	2 (0.30)	4 (0.60)	5 (0.75)
Cost	0.20	3 (0.60)	5 (1.00)	2 (0.40)
Precision	0.20	5 (1.00)	2 (0.40)	4 (0.80)
Power draw	0.10	2 (0.20)	4 (0.40)	3 (0.30)
Ease of control	0.10	3 (0.30)	4 (0.40)	4 (0.40)
Total	1.00	3.40	4.05	3.40

Rating scale: 1 = poor, 5 = excellent.
Score = Rating × Weight.
Check sensitivity: vary weights ±0.05.

Winner

Figure 3.4: Weighted decision matrix example: motor selection with six criteria, three options.

3.5.3 Worked Example: MCU Trade Study

Decision: Select an MCU platform for a battery-powered environmental sensor node.

Criterion (Weight)	ESP32	STM32L4	ATmega328P
Power consumption (0.25)	3 (0.75)	5 (1.25)	4 (1.00)
Analog channels (0.20)	5 (1.00)	4 (0.80)	4 (0.80)
Cost (0.20)	5 (1.00)	3 (0.60)	5 (1.00)
Flash/RAM (0.15)	5 (0.75)	5 (0.75)	2 (0.30)
Ecosystem (0.10)	5 (0.50)	3 (0.30)	5 (0.50)
Wi-Fi/BLE (0.10)	5 (0.50)	1 (0.10)	1 (0.10)
Total	4.50	3.80	3.70

Table 3.1: MCU trade study for battery-powered sensor node.

Sensitivity check: If “Power consumption” weight increases from 0.25 to 0.30 (taking 0.05 from Wi-Fi), STM32L4 total rises to 4.05 and ESP32 drops to 4.40; winner unchanged. Selection is robust.

Decision: ESP32. Despite higher power draw, the built-in Wi-Fi eliminates a separate module (\$8 saved, one fewer interface, one fewer failure mode). For battery life, use deep sleep aggressively (Section 12.4).

3.6 Requirements-to-Architecture Flow

The five stages produce specific deliverables: ConOps, requirements document, functional tree, trade study reports, and architecture document (block diagrams + BOM + ICDs). V&V results feed backward through the chain when failures occur.

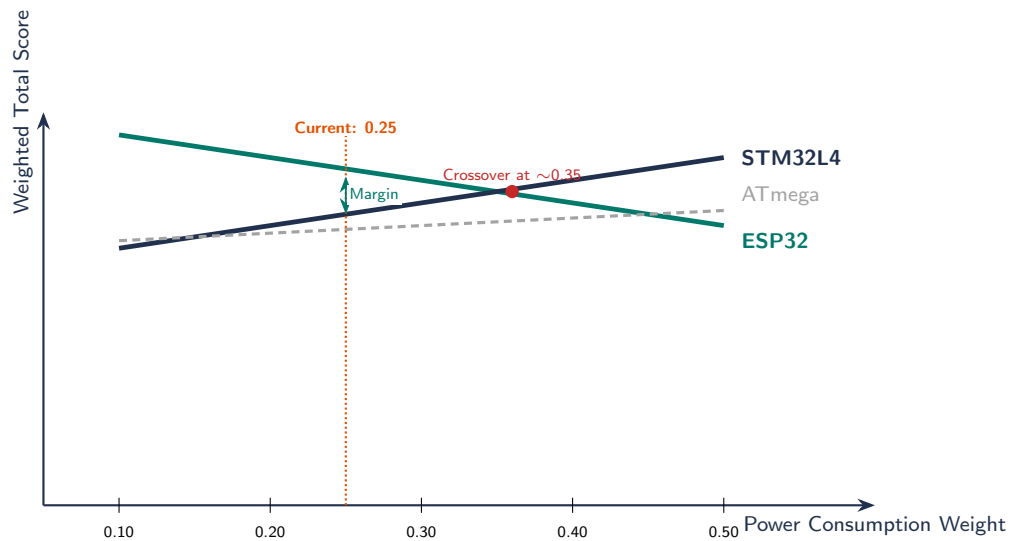


Figure 3.5: Sensitivity analysis for the MCU trade study. The ESP32 remains the winner until the “Power consumption” weight exceeds ~ 0.35 (40% above the assigned value). This large margin to crossover confirms the selection is robust. When the margin is small (< 0.05), prototype both finalists.

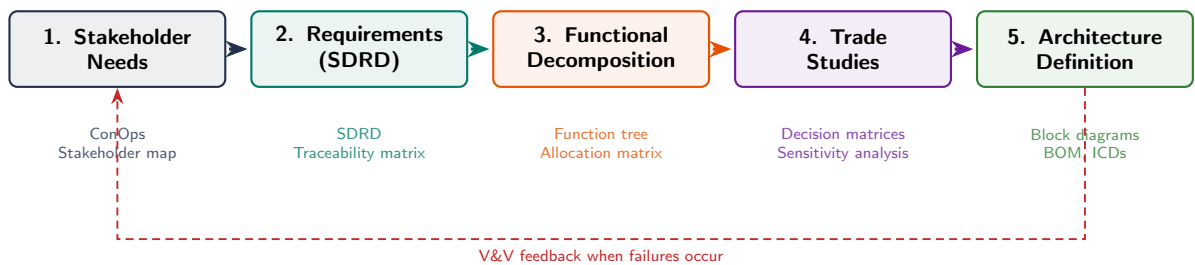


Figure 3.6: Five-stage design flow: stakeholder needs $\rightarrow$ requirements $\rightarrow$ functional decomposition $\rightarrow$ trade studies $\rightarrow$ architecture.

GS GreenShield: Functional Decomposition

Top-level function: Protect greenhouse crops from frost damage.

Decomposition:

- **Sense Environment** → Measure Air Temp, Measure Soil Temp, Detect Heater Status
- **Process Data** → Compare to Setpoint, Apply Hysteresis, Log Data
- **Control Actuator** → Activate/Deactivate Heater Relay
- **Communicate Status** → Send SMS Alert, Display Local Status, Receive Remote Commands
- **Provide Power** → Regulate Supply Voltage, Manage Battery, Protect Circuits

Each leaf function maps to a physical component. “Measure Air Temp” → DS18B20 sensor. “Regulate Supply Voltage” → 5 V buck converter.

Worked Example. GreenShield Architecture: Grouping functions into subsystems yields: Sensor subsystem (DS18B20 + soil probe), Controller subsystem (ESP32), Actuator subsystem (relay + heater), Communications subsystem (SIM7000G LTE-M module), and Power subsystem (12 V adapter + 5 V buck + 3.3 V LDO).

Industry SE Parallels

- **INCOSE SEH 4th ed.:** Sec. 4.2 (System Requirements Definition), Sec. 4.3 (Architecture Definition).
- **NASA SP-2016-6105 Rev 2:** Ch. 4.2 (Technical Requirements Definition), Ch. 4.3 (Logical Decomposition).
- **IEEE 1233:** Guide for developing system requirements specifications.

3.7 Artifact Handoff: What Chapter 3 Produces

Artifact	Description	Forward Reference
SDRD	System requirements with traceability and V&V assignments	Ch. 3–6, PDR baseline
Functional Decomposition	Function tree from mission to leaf functions	Ch. 3 (concept generation)
Allocation Matrix	Functions mapped to physical modules	Ch. 5 (integration)
Trade Study Reports	Weighted decision matrices with sensitivity analysis	Ch. 3, PDR

3.8 Writing Good Requirements: Advanced Techniques

3.8.1 The SHALL/SHOULD/MAY Hierarchy

SHALL: mandatory requirement. The system must satisfy it. Non-compliance is a deficiency that must be resolved. “The system shall operate on 12 V DC $\pm 10\%$.”

SHOULD: desired but not mandatory. Non-compliance is noted but does not block acceptance. “The enclosure should be splash-resistant to IP44.”

MAY: optional. Implemented if time and budget permit. “The system may include a wireless data logging mode.”

Every requirement must use one of these three verbs. Avoid “will,” “can,” “is,” and other ambiguous verbs that obscure whether the requirement is mandatory.

3.8.2 Requirement Attributes

Each requirement in the SDRD should have the following attributes:

ID: unique identifier (SYS-001, MECH-003, SW-012). The prefix indicates the domain; the number is sequential.

Text: the “shall” statement, precisely worded.

Rationale: why this requirement exists. Traces to a stakeholder need, a safety standard, a physics constraint, or a design decision.

Verification method: how compliance will be demonstrated (Test, Analysis, Inspection, or Demonstration).

Priority: critical (must have for MVP), important (should have), or desirable (nice to have).

Parent: the higher-level requirement from which this one is derived. System requirements derive from stakeholder needs; subsystem requirements derive from system requirements.

3.8.3 Deriving Subsystem Requirements

System-level requirements must be decomposed into subsystem requirements that individual team members can design to:

System: “The system shall maintain temperature at $50 \pm 1^\circ\text{C}$.”

Derived sensor: “The temperature sensor shall have accuracy $\leq \pm 0.3^\circ\text{C}$ over $20\text{--}80^\circ\text{C}$.”

Derived controller: “The PID controller shall reduce steady-state error to $< 0.5^\circ\text{C}$.”

Derived actuator: “The heater shall provide $\geq 50\text{ W}$ of thermal power.”

Derived power: “The power supply shall deliver $\geq 5\text{ A}$ at 12 V .”

Each derived requirement traces to the parent and, through the parent, to the stakeholder need. This traceability chain is essential for justifying every design decision and for verifying that the complete system meets user expectations.

3.9 Architecture: Making the Right Decisions

3.9.1 Functional vs. Physical Architecture

The **functional architecture** describes what the system does, decomposed into functions (sense, process, actuate, communicate, power). It is independent of how those functions are implemented.

The **physical architecture** maps functions to physical components. “Sense temperature” → Type K thermocouple + INA128 amplifier + ESP32 ADC. “Actuate heater” → SSR + 200 W ceramic heater. The physical architecture makes the design concrete and testable.

3.9.2 Interface Control Documents (ICD)

An ICD defines the physical, electrical, and data interfaces between two subsystems. For each interface:

Mechanical: mating features, bolt pattern, gasket requirements, alignment tolerances.

Electrical: signal names, voltage levels, current requirements, connector type and pinout, cable length and shielding.

Data: protocol (I2C, SPI, UART, Wi-Fi), data format, message rate, error handling.

ICDs are contracts between subsystem teams. Once agreed upon, either side can design and build their subsystem independently, confident that they will connect correctly at integration. Without ICDs, integration is chaos.

3.9.3 Architecture Tradeoff Analysis

When multiple architectures are feasible, use a tradeoff matrix (Pugh matrix) to evaluate them systematically:

- (1) List the evaluation criteria (cost, complexity, reliability, performance, schedule risk).
- (2) Weight each criterion by importance (1–5 scale).
- (3) Score each architecture against each criterion (1–5 scale).
- (4) Multiply weights $\times$ scores and sum.
- (5) The highest total score is the recommended architecture, but examine the scores for any criterion where the recommended option is significantly worse than an alternative (potential deal-breaker).

Document the tradeoff analysis in the PDR package. The review board wants to see that the architecture was chosen through engineering analysis, not by default or by the loudest team member's preference.

3.10 Practice Questions

1. Rewrite this bad requirement as a well-formed shall-statement: "The system should be fast and accurate."
2. A system requirement states "shall measure temperature to $\pm 0.5^\circ \text{C}$." Decompose this into three subsystem requirements for the sensor, ADC, and display subsystems.
3. Your project has a requirement: "shall use an Arduino Mega 2560." Is this a valid requirement? Why or why not? Rewrite it properly.
4. Explain the difference between forward and backward traceability. Why do you need both?
5. Create a weighted decision matrix comparing three MCU options for a battery-powered sensor node. Define at least five criteria with weights.
6. Why should you perform functional decomposition BEFORE selecting physical components?
7. A trade study produces scores of 3.80 vs. 3.75. What should you do before accepting the winner?

3.10.1 Answers

1. Two requirements: "SYS-REQ-001: The system shall respond to user input within 200 ms of activation." "SYS-REQ-002: The system shall measure ambient temperature with accuracy of $\pm 0.5^\circ \text{C}$."
2. (a) "The sensor shall output a signal proportional to temperature with linearity $\leq \pm 0.2^\circ \text{C}$ over -20 to $+60^\circ \text{C}$." (b) "The ADC shall digitize the sensor signal with resolution $\leq 0.1^\circ \text{C}$ and total error $\leq \pm 0.15^\circ \text{C}$." (c) "The display shall present the temperature value with update rate $\geq 1 \text{ Hz}$ and display resolution of 0.1°C ." Note: $\sqrt{0.2^2 + 0.15^2} + \text{display} < 0.5^\circ \text{C RSS}$.

3. Not valid; it specifies implementation (HOW) rather than need (WHAT). Rewrite: “SYS-REQ-XXX: The controller shall provide ≥ 12 analog input channels, ≥ 4 serial interfaces, and ≥ 256 KB flash memory. Rationale: sensor count and communication needs.”
4. Forward: parent $\rightarrow$ children. Checks completeness, every requirement is decomposed. Backward: child $\rightarrow$ parent. Checks justification, every spec traces to a need. Without forward, requirements go unimplemented. Without backward, unnecessary work gets done.
5. Criteria might include: power consumption (0.25), analog channels (0.20), cost (0.20), flash/RAM (0.15), ecosystem/library support (0.10), package size (0.10). Options: STM32L4, ESP32, ATmega328P.
6. Functions are stable across implementations. Decomposing functionally separates the problem space from the solution space, enabling objective trade studies. Jumping to hardware locks the design before alternatives are explored.
7. Sensitivity analysis: vary each weight by ± 0.05 and check if the winner changes. A 0.05 margin is very tight. Investigate the criteria where the options diverge most. Consider prototyping both finalists.

Chapter 4

Concept Generation & Early Design

When to Read This Chapter

Sprint 2: Read the full chapter. Sections on morphological charts, decision matrices, and trade studies are your primary PDR deliverables. **Sprint 3:** Return to the prototyping and BOM sections when transitioning to detailed design and procurement. **Related:** builds on Chapter 3 requirements (p. 20); feeds into PDR (Chapter 21, p. 218); prototype types connect to the integration strategy in Chapter 6 (p. 55).

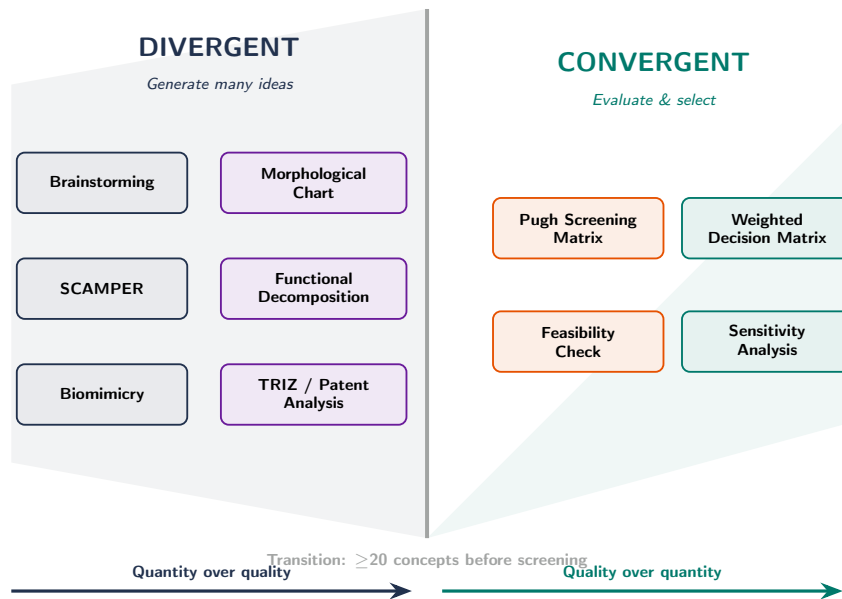


Figure 4.1: Concept generation techniques: divergent to convergent.

4.1 Reverse Engineering

Reverse engineering is a structured process of disassembling an existing product to understand how it works. It builds engineering intuition and directly informs concept generation.

4.1.1 The Reverse Engineering Process

1. **Document the intact product:** photograph from all angles, measure overall dimensions, weigh it, record all labels and markings.

Morphological Chart: 5 Functions × 3 Options = 243 Combinations

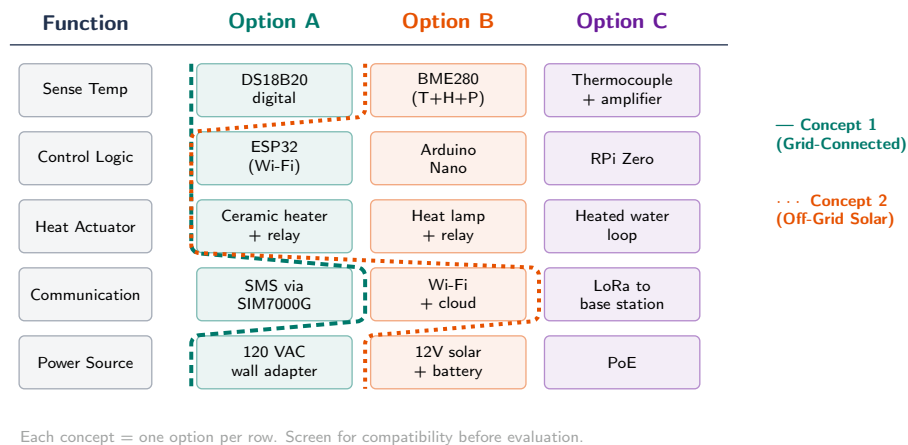


Figure 4.2: Morphological chart with systematic concept combinations.

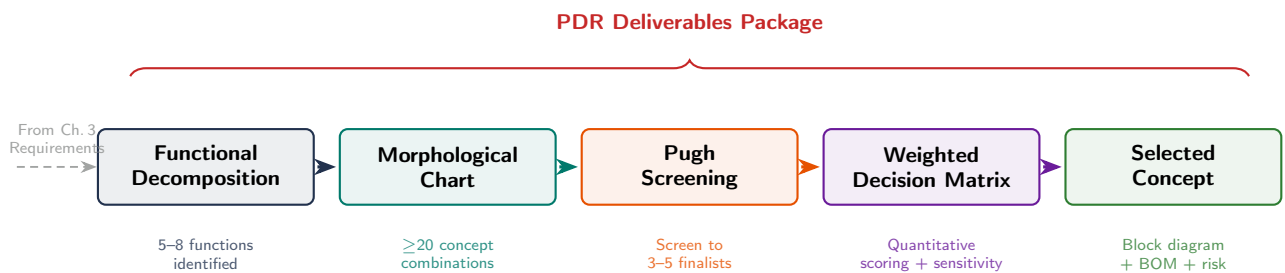


Figure 4.3: Concept-to-design pipeline with PDR deliverables.

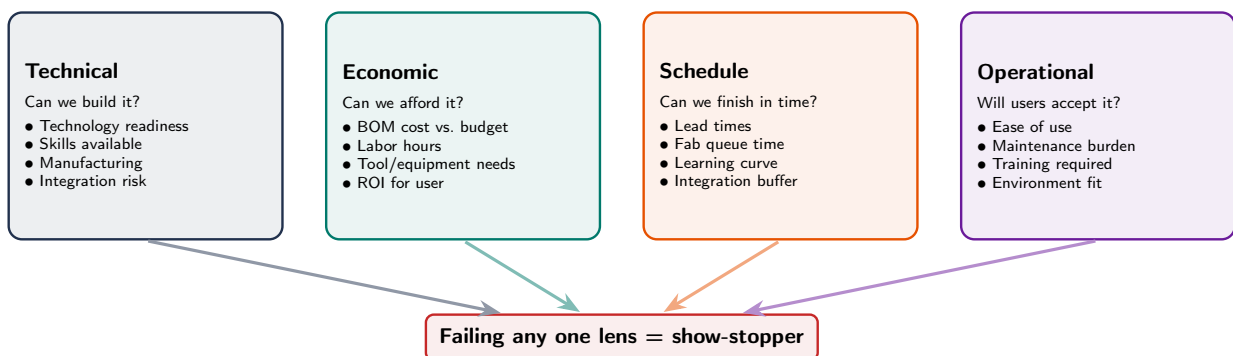


Figure 4.4: Four-lens feasibility assessment.

2. **Disassemble systematically:** remove fasteners in order, photograph each step, bag and label hardware. Use the right tools; forcing components causes damage that obscures design intent.
3. **Measure and catalog:** calipers for every component. Record materials (magnet test for ferrous metals, density test for plastics). Identify standard parts (bearings, fasteners, seals) by markings.
4. **Create the BOM:** part number, description, quantity, material, source, estimated cost.
5. **Create an exploded view:** CAD or carefully arranged photograph with callout labels.
6. **Functional analysis:** map each component to a function. Which are structural? Functional? Cosmetic? Note any clever design solutions.

4.1.2 What Reverse Engineering Reveals

Assembly sequence (what goes in first must come out last. DFA insight). Material choices and why they were made. Manufacturing processes (injection molding witness lines, machining marks, weld joints). Standard vs. custom parts ratio. Design-for-cost decisions. Failure modes on used products (wear patterns).

4.1.3 Reverse Engineering Outputs

The analysis produces artifacts that feed directly into design:

- **Component inventory** with identified materials and manufacturing methods; informs DFM decisions (Injection Molding chapter).
- **Interface inventory** of all accessible mounting points, electrical connections, and mechanical features; basis for ICDs (Chapter 2).
- **Constraint list** documenting what your design cannot change; feeds requirements as non-functional constraints.
- **Design insights** about proven approaches, failure modes, and opportunities; informs concept generation and FMEA.

Non-Destructive Constraint

In MEGN 301, you must return the COTS product undamaged at the end of each class period. No drilling, cutting, gluing, or permanent modifications. Your add-on system must interface through existing features: existing screw holes, connectors, friction fits, clamps, or custom brackets. This mirrors real-world accessory development where your product must work with an existing platform without voiding its warranty.

4.1.4 From Reverse Engineering to Concept Generation

The reverse engineering analysis directly feeds concept generation:

1. The interface inventory defines the physical envelope and connection options for your morphological chart.
2. The constraint list eliminates infeasible concepts before time is wasted developing them.
3. Design insights from the existing product provide proven approaches to build upon.
4. The component inventory reveals standard parts that you can reuse or specify.

4.2 Generating Concepts

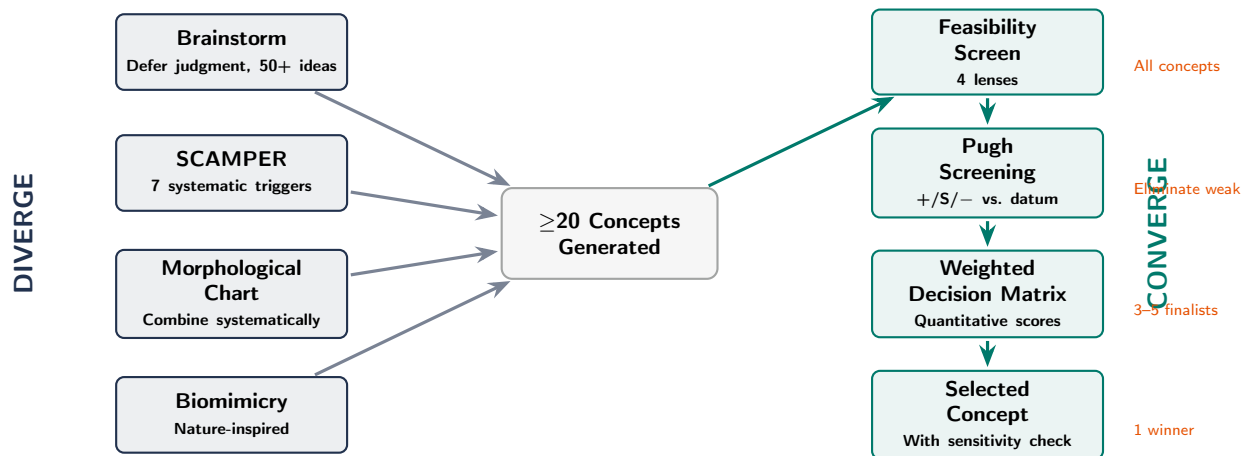


Figure 4.5: The concept generation process: from functional decomposition through systematic evaluation.

4.2.1 Diverge Then Converge

The concept generation process (Figure 4.5) has two distinct phases. **Divergent** thinking generates as many ideas as possible without evaluation (brainstorming, SCAMPER, morphological charts, and bio-inspired design). **Convergent** thinking systematically evaluates and selects (Pugh screening matrices and weighted decision matrices). Generate at least 20 concepts before screening any of them.

4.2.2 Brainstorming Rules

Defer judgment (no criticism during ideation). Go for quantity (target 50+ ideas in 30 minutes). Build on others' ideas ("yes, and..."). Encourage wild ideas (they contain seeds of insight). Sketch, don't just talk.

4.2.3 SCAMPER

Seven systematic triggers: Substitute, Combine, Adapt, Modify/Magnify, Put to other use, Eliminate, Reverse. Apply each to existing concepts or competitor products to generate variations.

4.2.4 Morphological Charts

List functions as rows (from functional decomposition), brainstorm 3–5 implementation options per function as columns. A complete concept = one option from each row. A 5×4 chart produces 1,024 combinations, systematically exploring the design space.

4.2.5 Bio-Inspired Design

Extract functional principles from biological systems. Velcro from burdock burrs. Sharkskin for drag reduction. Use AskNature.org to search by function. TRIZ provides 40 inventive principles for resolving design contradictions.

4.2.6 Stage 1: Pugh Screening Matrix

Rate each concept against a datum (baseline concept) as better (+), same (S), or worse (-). No weights, no numbers; just qualitative comparison. Sum +'s and -'s. Eliminate concepts with many -'s. The survivors proceed to the weighted matrix.

Rules: at least 3 concepts plus the datum. Criteria from requirements only. If two concepts score similarly, do not eliminate either.

4.2.7 Stage 2: Weighted Decision Matrix

Rate surviving concepts 1–5 against weighted criteria (weights sum to 1.0). Calculate weighted score per option. Check sensitivity (see Figure 3.5 for a graphical example): vary each weight ± 0.05 ; if the winner changes, the selection is fragile.

Common traps: Straw-man alternatives (including clearly inferior options to make the favorite “win”; every candidate must be legitimate). Anchoring on a favorite (assign ratings independently before group discussion). Criteria not from requirements (if a criterion does not trace to a requirement, remove it). Post-hoc weight adjustment (lock weights before scoring).

Stage 1: Pugh Screening: rate concepts as +/S/- against a datum. Fast, no weights. Eliminates clearly inferior options. **Stage 2: Weighted Decision Matrix:** rate finalists 1–5 against weighted criteria. Quantitative, traceable. Sensitivity check: if ± 0.05 weight change flips the winner, investigate further.

4.3 Risk Assessment

Risk management begins during concept selection, not after design. A **risk** is any event or condition that could prevent the system from meeting requirements. Every concept carries risks; the question is which are acceptable and which require mitigation.

4.3.1 The Risk Matrix

Risks are characterized by **likelihood** (1–5) and **consequence** (1–5). The product $L \times C$ gives the Risk Priority Number (RPN).

Score	Likelihood	Consequence
1	Very unlikely (<5%)	Negligible (cosmetic, workaround exists)
2	Unlikely (5–20%)	Minor (schedule slip <1 week, minor rework)
3	Possible (20–50%)	Moderate (schedule slip 1–2 weeks, partial re-design)
4	Likely (50–80%)	Major (missed milestone, significant redesign)
5	Almost certain (>80%)	Catastrophic (project failure, safety hazard)

Table 4.1: Risk likelihood and consequence scales.

Action thresholds: RPN 1–4 = monitor, 5–9 = mitigate, 10–15 = mitigate aggressively, 16–25 = redesign or eliminate. Maintain a risk register from PDR through FDR, updating status at each review.

4.3.2 Risk Mitigation Strategies

Four options (in order of preference): **Avoid** (change the design to eliminate the risk), **Mitigate** (reduce likelihood or consequence), **Transfer** (shift risk to a vendor, insurer, or partner), **Accept** (document and monitor; only for low-RPN items).

GS GreenShield: Risk Register at PDR

Risk	L	C	RPN	Mitigation
Solar panel insufficient in winter	4	4	16	Battery margin + Dec test
LoRa blocked by terrain	3	3	9	Site survey + repeater
Relay fails closed (heater on)	2	5	10	Thermal fuse backup
Thermocouple accuracy inadequate	3	3	9	In-situ calibration
MCU brown-out during Wi-Fi TX	4	3	12	470 μ F decoupling

Red-zone: solar panel (16) and MCU brown-out (12) require active mitigation before CDR.

4.4 Failure Modes and Effects Analysis (FMEA)

FMEA systematically identifies potential failure modes, their effects, and their causes *during design*; before hardware exists.

4.4.1 Conducting a Design-Phase FMEA

For each component or function:

1. List every way it can fail (**failure modes**).
2. Determine the effect of each failure on the system (**effects**).
3. Identify root causes.
4. Rate Severity (S), Occurrence (O), and Detection (D) on 1–10 scales.
5. Calculate $RPN = S \times O \times D$. Higher RPN = higher priority for corrective action.
6. Design corrective actions for the top-5 RPN items.

Component	Failure Mode	Effect	S	O	D	RPN
Temp sensor	Reads high	Heater never activates; crop loss	8	3	4	96
Relay	Stuck closed	Heater runs continuously; fire risk	9	2	6	108
MCU	Brown-out reset	System offline during freeze event	7	4	5	140
cellular module	No signal	Alert not sent; operator unaware	5	3	3	45
Power supply	Fails open	Complete system loss	8	2	7	112

Table 4.2: Example design-phase FMEA for GreenShield frost protection system.

The MCU brown-out (RPN 140) is the highest-priority item; address it with decoupling capacitors, a voltage supervisor, and separate power routing (see Chapter 12).

4.4.2 Prototype Types in Detail

Proof of Concept (PoC): validates the riskiest technical assumption. Build the minimum hardware/software to answer one question: “Does the physics/algorithm/mechanism work?” Example: a breadboard circuit with just the sensor and MCU to verify temperature accuracy before designing the full system. Build time: 1–3 days. Quality: ugly is fine.

Form/Fit prototype: verifies physical compatibility. 3D-print the enclosure, mount real components, check cable routing and assembly sequence. No electronics need to function; this is a geometry check. Build time: 2–4 days. **Always print at 1:1 scale and test-fit real components.**

Looks-Like/Works-Like: a near-complete system that demonstrates both form and function. This is your CDR deliverable. It should be close to the final design but may use breadboard wiring or temporary mounting. Build time: 1–3 weeks.

Minimum Viable Product (MVP): the simplest version that delivers value to the stakeholder. It meets the MUST requirements but may defer SHOULD and COULD features. The MVP is what you demonstrate at FDR.

Build the Highest-Risk PoC First

Identify the single technical element most likely to fail. Build a PoC for that element before anything else. If it fails, you learn early and can pivot. If it works, confidence increases and the rest of the design proceeds on solid ground.

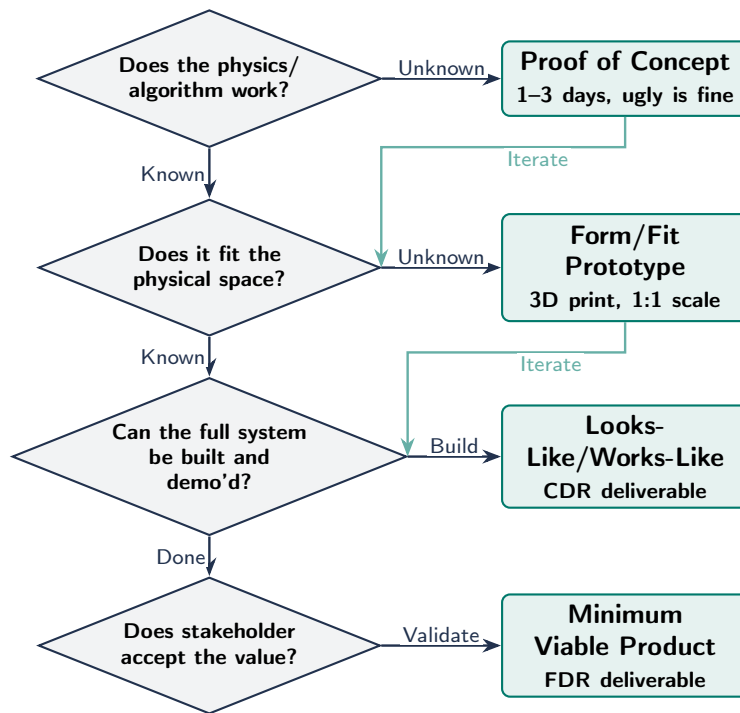


Figure 4.6: Prototype type decision flowchart. Start at the top: address unknowns with the simplest prototype type, iterate, and progress downward. Each stage answers a different question.

4.5 Prototyping and Feasibility

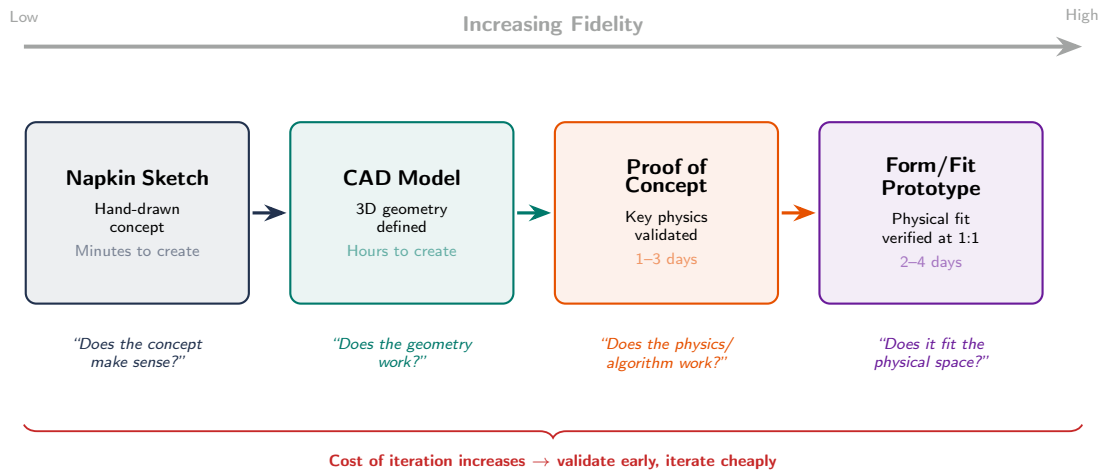


Figure 4.7: From sketch to prototype: progressive fidelity in the concept-to-hardware pipeline.

Four prototype types (see the progression in Figure 4.7 and the decision flowchart in Figure 4.6): Proof of Concept (does the physics work?), Form/Fit (does it fit the space?), Looks-Like/Works-Like (can the team build the full system?), and Minimum Viable Product (does the stakeholder accept it?). Build the PoC for the highest-risk element FIRST.

4.5.1 Feasibility Assessment

Every concept must pass four tests: **Technical** (can we build it?), **Economic** (can we afford it?), **Schedule** (can we finish in time?), **Operational** (will users accept it?). Failing any one is a show-stopper.

4.6 Design Documentation

Complete design documentation enables someone else to build, test, and maintain your system without your presence.

4.6.1 Bill of Materials (BOM)

Required columns: item number, part number, description, quantity, material, vendor, vendor part number, unit cost, lead time, status (ordered/received/installed). Organize hierarchically: Level 0 = system, Level 1 = subsystems, Level 2 = assemblies, Level 3 = parts. **Every part must appear in the BOM.**

4.6.2 Wiring Diagrams and Schematics

Schematics show electrical connections using standard symbols; the “source code” of hardware. **Wiring diagrams** show physical routing with connector pin assignments and wire colors/gauges. Both are required: schematics for design intent, wiring diagrams for assembly.

4.6.3 Software Architecture and Flowcharts

Document firmware with: (1) state-machine diagram showing all states and transitions, (2) data flow diagram (sensor → processing → actuator), (3) pin mapping table (GPIO → function → component), (4) communication protocol definitions (message formats, baud rates, addresses).

4.6.4 CAD Models and Engineering Drawings

CAD models define geometry. Engineering drawings communicate that geometry with dimensions, tolerances, surface finishes, and notes. Every custom part requires a drawing. Standard parts reference vendor part numbers; no drawing needed.

Plan B Components

For every critical component, identify a Plan B; an alternative from a different vendor meeting the same requirements. Lead times slip, parts go out of stock, and vendors make mistakes. A Plan B list saves weeks of panic.

Free Electronics Supplies

Before ordering, check the MEGN 301 electronics inventory in the ME shop. Common components (resistors, capacitors, LEDs, wire, headers, breadboards) are available at no cost. Check the shared inventory spreadsheet first.

The Spring Break Shipping Trap

Orders placed the week before spring break often arrive *during* break when no one is on campus. Plan so critical parts arrive at least one week before any closure. International suppliers (AliExpress, LCSC): allow 3–4 weeks. McMaster-Carr ships next-day to Golden.

GS GreenShield: Morphological Chart

Function	Option A	Option B	Option C
Sense Temp	DS18B20 digital	BME280 (T/H/P)	Thermocouple + amp
Control Logic	ESP32 (Wi-Fi)	Arduino Nano	RPi Zero
Heat Actuator	Ceramic heater + relay	Heat lamp + relay	Heated water loop
Communication	SMS via SIM7000G	Wi-Fi + cloud	LoRa to base station
Power Source	120 VAC wall adapter	12V solar + battery	PoE
Concept 1 (Grid-Connected): DS18B20 + ESP32 + Ceramic heater + Wi-Fi + Wall adapter. Lowest cost, simplest, requires AC power.			
Concept 2 (Off-Grid Solar): BME280 + ESP32 + Ceramic heater + LoRa + Solar+battery. Grid-independent, higher cost, battery sizing critical.			
Concept 3 (Industrial): Thermocouple + RPi + Heated water loop + SMS + PoE. Highest capability and cost.			

GS GreenShield: Risk Assessment for Concept 2 (Off-Grid)

Technical risks: (1) Solar panel sizing for Colorado winters, December insolation is 3.5 peak sun hours vs. 6.5 in July. Battery must sustain 3 consecutive cloudy days. (2) LoRa range through polycarbonate glazing, 3–6 dB attenuation at 915 MHz.

Schedule risk: Solar charge controller + BMS integration adds 2 weeks. If the team is behind at CDR, this is a schedule-buster.

Economic risk: Solar (\$25) + charge controller (\$15) + LiPo (\$20) + LoRa (\$12 ea) = \$84 for power/comms vs. \$8 for Concept 1.

Verdict: High-risk for 16-week schedule. Recommend Concept 1 with upgrade path.

Industry SE Parallels

- **INCOSE SEH 4th ed.:** Sec. 4.3 (Architecture Definition), Sec. 4.4 (Design Definition).
- **NASA SP-2016-6105 Rev 2:** Ch. 4.4 (Design Solution Definition), Ch. 6.6 (Decision Analysis).
- **IEEE 1220:** Standard for systems engineering process management.

4.7 Artifact Handoff: What Chapter 4 Produces

Artifact	Description	Forward Reference
Morphological Chart	Functions × implementation options	PDR documentation
Trade Study / Decision Matrix	Weighted evaluation of top concepts	PDR deliverable
Selected Concept Package	Block diagram, preliminary BOM, risk assessment	Ch. 4, CDR
Design Documentation	BOM, schematics, CAD, software architecture	Ch. 5 (integration), FDR

4.8 Structured Creativity Methods

4.8.1 Morphological Analysis

Morphological analysis systematically generates concepts by combining solutions for each function:

- (1) List the key functions the product must perform (sense, actuate, power, communicate, enclose).
- (2) For each function, brainstorm 3–5 alternative solutions (for “sense temperature”: thermocouple, RTD, thermistor, IR sensor, bimetallic strip).
- (3) Arrange in a matrix with functions as rows and solutions as columns.
- (4) Generate concepts by selecting one solution from each row and combining them. A 5-function matrix with 4 alternatives per function yields $4^5 = 1024$ theoretical combinations.
- (5) Screen the combinations for compatibility (not all combinations are physically or economically feasible) and select the 5–10 most promising for detailed evaluation.

4.8.2 SCAMPER Method

A checklist for modifying existing designs: **S**ubstitute (different material, component, process), **C**ombine (merge two functions into one component), **A**dapt (borrow a solution from another domain), **M**odify (change size, shape, color, texture), **P**ut to other use (use the product for a different application), **E**liminate (remove a feature or component), **R**everse/Rearrange (change the order, orientation, or hierarchy).

4.8.3 Biomimicry

Nature has solved many engineering problems through evolution. Look to biological systems for inspiration: gecko feet for adhesion without glue, termite mounds for passive ventilation, shark skin for drag reduction, pinecones for humidity-responsive actuation, and bone structures for lightweight strength. In MEGN 301, biomimicry is particularly relevant for thermal management (how do desert animals stay cool?), fluid handling (how do trees pump water 100 m upward?), and structural design (how do shells achieve high strength-to-weight ratios?).

4.9 Concept Selection: Beyond the Pugh Matrix

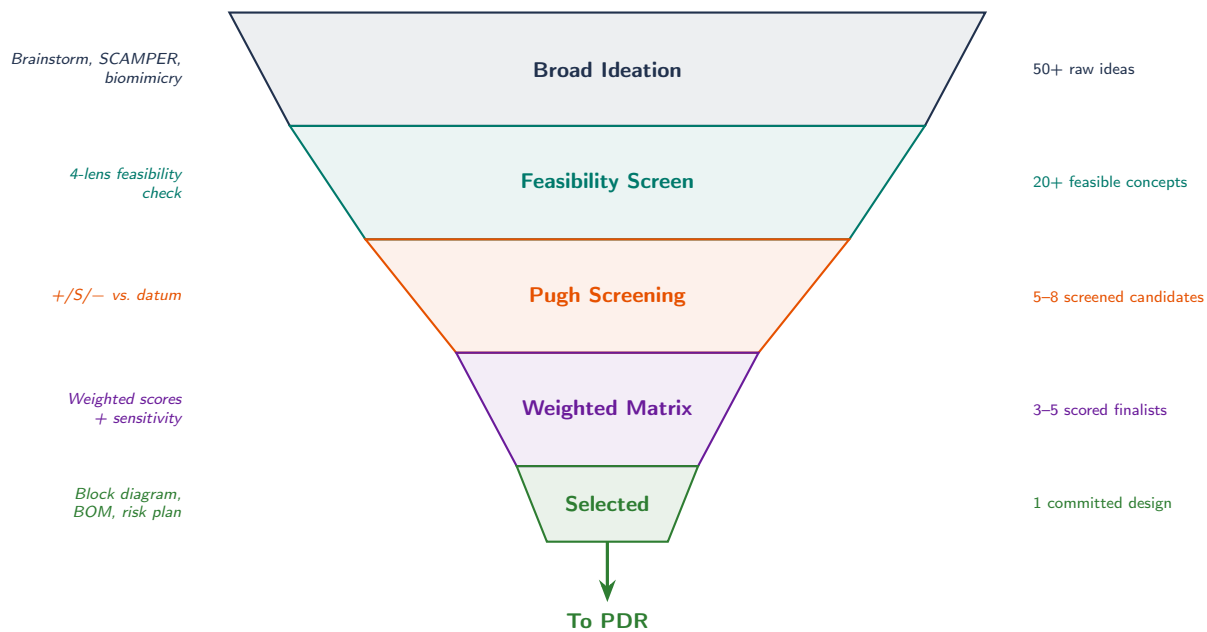


Figure 4.8: Concept selection funnel: from broad ideation through screening to a single committed design.

4.9.1 Weighted Decision Matrix

The concept selection funnel (Figure 4.8) narrows candidates systematically. An enhanced Pugh matrix where criteria are weighted by importance:

- (1) Define 6–10 evaluation criteria (performance, cost, complexity, reliability, manufacturability, safety, sustainability).
- (2) Assign weights (total = 100%). Typically: performance 25%, cost 20%, manufacturability 15%, reliability 15%, safety 10%, sustainability 10%, aesthetics 5%.
- (3) Score each concept against each criterion on a 1–5 scale.
- (4) Compute weighted scores and rank.
- (5) Perform sensitivity analysis: vary the weights by $\pm 20\%$ and check whether the ranking changes. If it does, the decision is sensitive to subjective weighting and requires careful justification.

4.9.2 Concept Testing

Before committing to a concept, build a quick proof-of-concept prototype to validate the highest-risk element. This is *not* the final prototype; it is a focused experiment to answer one question: “does this approach actually work?”

For a thermal control system: the highest risk might be “can we maintain $\pm 1^\circ\text{C}$ with a 200 W heater and PID control?” Build a minimal system (heater, thermocouple, Arduino, relay) and test. The result either validates the concept (proceed) or invalidates it (pivot), saving weeks of detailed design effort.

4.10 Design for the GreenShield Competition

The GreenShield project evaluation criteria guide concept selection:

Innovation (25%): novel application of engineering principles, creative use of materials or processes, unique approach to the problem.

Technical execution (30%): quality of the requirements, design, fabrication, and testing. Meets all specified requirements.

Sustainability (20%): environmental impact assessment, design for disassembly, material selection, energy efficiency.

Communication (15%): quality of documentation, presentations, and demonstrations.

Teamwork (10%): evidence of effective collaboration, equitable contribution, and conflict resolution.

Concepts should be selected not only for technical merit but for alignment with these evaluation criteria. A technically simple but highly sustainable and well-documented project may score higher than a technically ambitious project that fails to meet half its requirements.

4.11 Failure Modes and Effects Analysis (FMEA) During Design

FMEA is not a post-mortem performed after the system fails. It is a **design tool** that prevents failures by forcing the team to systematically ask “what can go wrong?” during the concept and early design phase, when changes are cheap.

4.11.1 When to Perform FMEA

Perform FMEA *before* CDR, ideally during concept selection (this chapter) or architecture definition (Chapter 3). Waiting until Operations & Maintenance (Chapter 7) transforms FMEA from a prevention tool into an autopsy. The cost of discovering a failure mode scales exponentially with project phase: \$1 at concept, \$10 at CDR, \$100 at integration, \$1000 at FDR.

4.11.2 The FMEA Process

For each component or function in the design:

(1) **Failure mode**: how can it fail? Be specific. “Motor fails” is too vague. “Motor stalls due to overcurrent” or “motor shaft coupling slips under load” are actionable.

(2) **Effect**: what happens to the system if this failure occurs? “System overheats because heater stays ON” or “pump stops, fluid stagnates.”

(3) **Cause**: what could trigger this failure? “Insufficient motor torque for the load” or “set screw loosened by vibration.”

- (4) **Severity (S)**: how bad is the effect? 1 (negligible) to 10 (safety hazard, injury possible).
 (5) **Occurrence (O)**: how likely is the cause? 1 (extremely unlikely) to 10 (almost certain).
 (6) **Detection (D)**: how likely is the failure to be detected *before* it causes harm? 1 (certain detection) to 10 (no detection mechanism).
 (7) **Risk Priority Number**: $RPN = S \times O \times D$. Maximum $RPN = 1000$. Prioritize mitigation actions for the highest RPNs.

4.11.3 FMEA for GreenShield Projects

At minimum, analyze the top 10 failure modes covering: power system failures (short circuit, reverse polarity, battery overdischarge), sensor failures (open circuit, drift, saturation), actuator failures (stall, runaway, mechanical failure), communication failures (Wi-Fi dropout, I2C bus hang), and software failures (watchdog timeout, state machine deadlock).

GS GreenShield: FMEA Template

Component	Failure Mode	Effect	S	O	D	RPN
Heater SSR	Fails ON (shorted)	Uncontrolled heating, fire risk	9	2	4	72
ESP32	Firmware crash	All outputs freeze at last state	7	4	3	84
LiPo battery	Over-discharge	Battery damage, swelling	8	3	2	48
Motor coupling	Set screw loosens	Motor spins but load doesn't move	4	5	6	120
Thermocouple	Wire breaks (open)	Reading jumps to max or NaN	6	3	2	36

The motor coupling ($RPN = 120$) is the highest priority. Mitigation: use Loctite 242 on the set screw and add a flat on the shaft. The ESP32 crash ($RPN = 84$) is second: mitigation is the hardware watchdog timer and fail-safe relay (Chapter 15 in the 300 notes, Chapter 12 in this document).

4.12 Practice Questions

- Create a morphological chart for a campus recycling station with at least 4 functions and 3 options each. Trace two complete concept paths.
- Your team brainstormed only 5 concepts in 20 minutes. What went wrong? What facilitation techniques would you use to increase output?
- A trade study produces scores of 3.82 vs 3.79 between two motor options. What do you do?
- Your proof-of-concept test shows the sensor accuracy is 1.2°C against a 0.5°C requirement. What are your options?
- You are reverse-engineering a consumer drone. During disassembly you find a small metal weight glued inside one motor arm. What is its likely purpose, and what does this tell you about the manufacturer's design process?

6. Your team's BOM lists "motor" with no part number, no vendor, and no specifications. Why is this a problem, and what minimum information must every BOM entry contain?

4.12.1 Answers

1. Example morphological chart for campus recycling station:

Function	Option A	Option B	Option C
Identify material	RFID tag	Camera + ML	Weight sensor
Sort mechanism	Rotating drum	Conveyor + diverter	Gravity chute
User interface	Touchscreen	Button panel	Voice commands
Compaction	Hydraulic press	Screw auger	Gravity (none)

Concept 1: RFID + Conveyor + Touchscreen + Hydraulic = high accuracy, high cost, complex.

Concept 2: Camera + Gravity chute + Button + None = moderate accuracy, low cost, simple.

2. Too few concepts: (a) Criticism crept into brainstorming (evaluation during ideation kills creativity). (b) Team anchored on a favorite concept. (c) No systematic technique used. Fixes: enforce "no criticism" rule with a timer, use SCAMPER to force variations, require sketches not just words, set a minimum of 20 ideas before stopping.
3. A 0.03 margin is within noise. Sensitivity analysis: vary each weight by ± 0.05 . If the winner changes, the selection is not robust. Options: prototype both finalists, gather better data on the most differentiating criterion, or choose based on a secondary factor (availability, team familiarity).
4. Options: (a) Improve the sensor (better calibration, thermal isolation, averaging algorithm). (b) Change the sensor (switch to RTD from thermocouple, higher accuracy, higher cost). (c) Relax the requirement (stakeholder may accept $\pm 1.0^\circ\text{C}$, verify with stakeholder before changing). (d) Add a correction factor (calibrate against a reference and apply polynomial correction in firmware). Always investigate the requirement validity before redesigning hardware.
5. The weight is likely for vibration balancing. The motor arm, with motors and propellers, has a center of gravity that is not perfectly symmetric. The manufacturer added a counterweight to tune the balance and reduce vibration. This tells you the manufacturer performs dynamic balancing as a post-assembly step; a quality control measure that indicates vibration was a problem in earlier designs.
6. "Motor" without specifications is useless for procurement, testing, or replacement. A vendor cannot fill this order. Minimum BOM entry: manufacturer (e.g., Pololu), part number (e.g., 4843), description ("12V 100RPM DC gearmotor"), key specs (voltage, no-load speed, stall torque, rated current), quantity, unit cost, vendor, and lead time.

Chapter 5

Verification, Validation & Test Plans

When to Read This Chapter

Sprint 1: Read Sections 5.1–5.3 (V&V concepts, TAID methods, test procedure template) while writing your SDRD. Draft test plans in parallel with requirements. **Sprint 3:** Return to the VCRM section when updating for CDR. **Sprints 5–7:** Use the boundary testing and statistical sections during V&V execution. **Related:** When tests fail, see Chapter 20 (Debugging, p. 210). For test execution details, see Chapter 8 (p. 75).

5.1 Verification vs. Validation

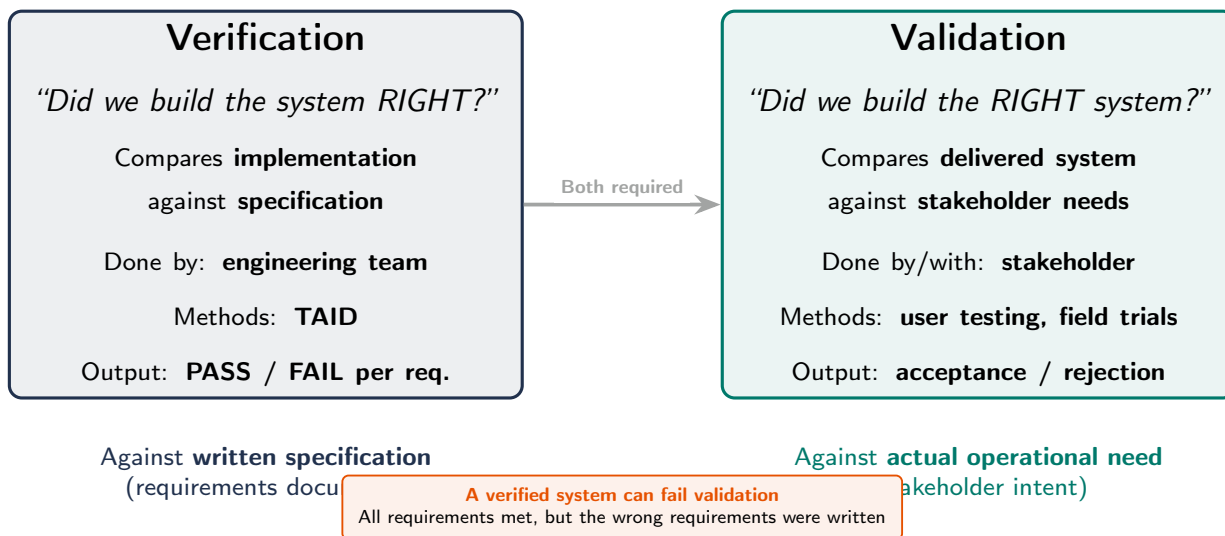


Figure 5.1: Verification (meets requirements) vs. validation (meets needs); two different questions.

Verification asks: "Did we build the system RIGHT?" It compares the implemented system against the written specification. Done by the engineering team using Test, Analysis, Inspection, or Demonstration (TAID). Each requirement receives an individual PASS/FAIL verdict.

Validation asks: "Did we build the RIGHT system?" It compares the delivered system against the stakeholder's actual operational need. Done with the stakeholder through user testing, field trials, or acceptance testing.

A Verified System Can Fail Validation

All requirements met perfectly, but the wrong requirements were written. This is why requirements quality matters; verification confirms specs; validation confirms the problem is actually solved. You need both.

5.2 Four Verification Methods: TAID

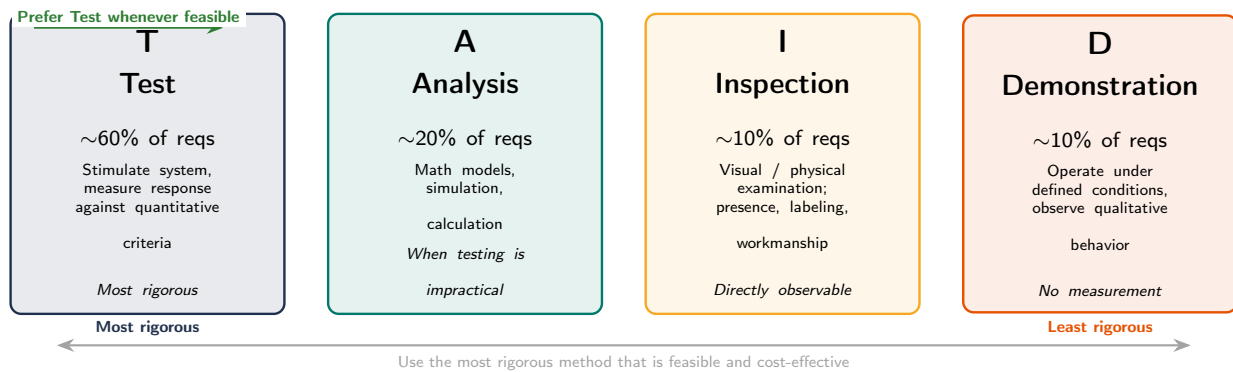


Figure 5.2: Test, Analysis, Inspection, and Demonstration with usage percentages and selection guidance.

- **Test (T, ~60% of reqs)**: Stimulate the system and measure the response against quantitative criteria. Most rigorous. Preferred whenever feasible.
- **Analysis (A, ~20%)**: Mathematical models, simulation, or calculation. Used when testing is impractical (FEA, thermal sim, worst-case analysis). Must document assumptions.
- **Inspection (I, ~10%)**: Visual/physical examination; presence, labeling, workmanship, material certificates.
- **Demonstration (D, ~10%)**: Operate under defined conditions and observe qualitative behavior. No measurement. Used when quantitative testing is not needed.

Assign the Method When You Write the Requirement

If you cannot identify a verification method, the requirement is untestable and must be rewritten. The method check is the single best quality gate for requirements.

5.3 Writing Test Plans

5.3.1 What Makes a Good Test Plan

The gold standard: another engineer with no project knowledge can reproduce your results exactly. Five qualities: **Traceable** (maps to requirement IDs), **Reproducible** (specific enough for independent replication), **Binary** (quantitative pass/fail, no judgment calls), **Complete** (all sections filled), **Executed as written** (no ad-hoc deviations).

5.3.2 Test Procedure Structure

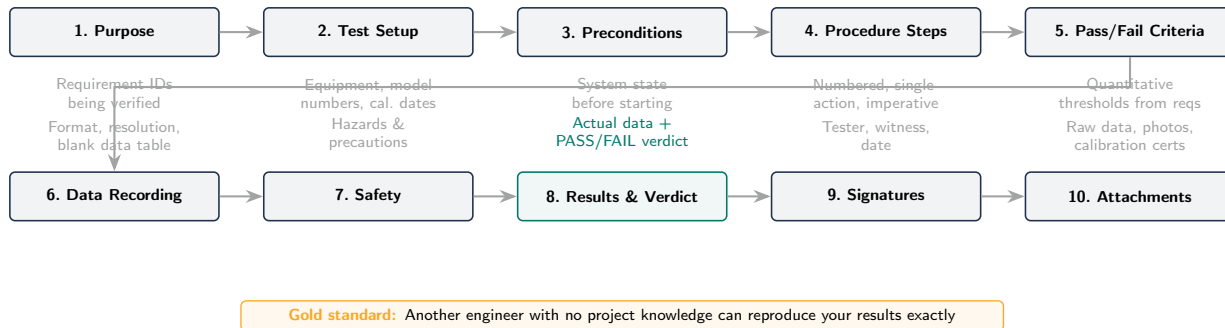


Figure 5.3: Complete test procedure template with all 10 required sections.

Every test procedure contains: (1) Purpose; requirement IDs verified; (2) Test setup; equipment with model numbers and calibration; (3) Preconditions; system state before starting; (4) Procedure steps; numbered, single-action, imperative voice, with expected results; (5) Pass/fail criteria; quantitative thresholds from requirements; (6) Data recording; format, resolution, blank table; (7) Safety precautions; (8) Results and verdict; (9) Signatures; (10) Attachments.

5.3.3 Writing Good Procedure Steps

Each step is a single action with an expected result. Use specific values (“Set reference bath to $25.0 \pm 0.1^\circ \text{C}$ ”), not vague instructions (“check temperature”). Include wait times for stabilization. Imperative voice: “Set...” “Record...” “Verify...”

5.4 Handling Results

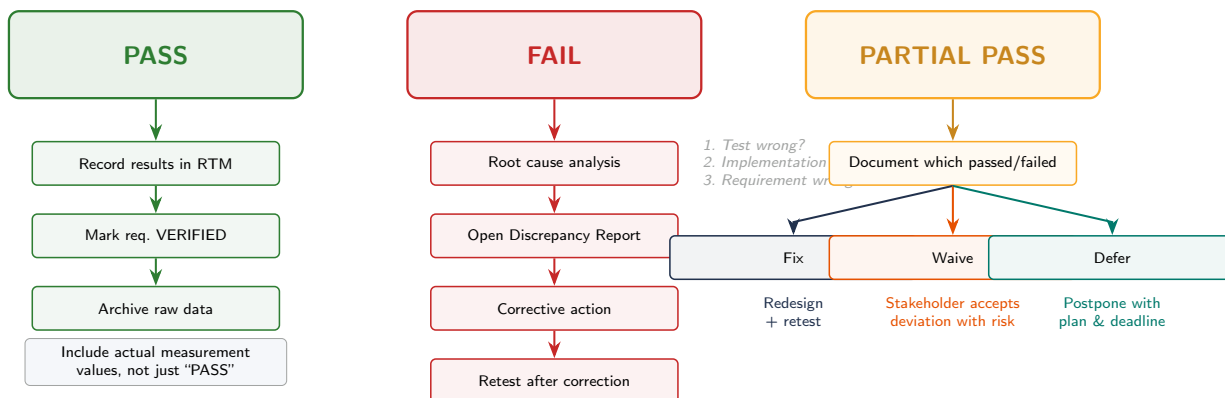


Figure 5.4: Three verification outcomes: PASS, FAIL, PARTIAL PASS with disposition workflows.

5.4.1 PASS

All criteria met. Record results in RTM, mark requirement VERIFIED, archive raw data. Include actual measurement values, not just “PASS.”

5.4.2 FAIL

One or more criteria not met. Root cause analysis in order: (1) Is the test procedure wrong? (2) Is the implementation wrong? (3) Is the requirement wrong? Open a Discrepancy Report (DR), assign corrective action, track to resolution, retest after correction.

5.4.3 PARTIAL PASS

Some criteria met, some not. Document which passed and which failed with data. Three resolution paths: **Fix** (redesign + retest), **Waive** (stakeholder formally accepts the deviation with documented risk), or **Defer** (postpone with a plan and deadline). A waiver transfers risk to the stakeholder.

5.4.4 Root Cause Analysis

Three possible root causes, investigated in order: (1) the test is wrong (procedure error, calibration issue); (2) the implementation is wrong (most common); (3) the requirement is wrong (overly conservative or based on incorrect assumptions). Changing a requirement requires formal change control. Never silently.

5.5 The VCRM

The Verification Cross-Reference Matrix links every requirement to its method, procedure, and status. Maintain from day one. At FDR, it tells the complete story. Example columns: Req ID, Text, Type, Method, Test Proc ID, Status, Notes.

5.5.1 VCRM Evolution Over Time

The VCRM is not a static document; it evolves at each project milestone:

- **At PDR:** all requirements listed, methods assigned, status = “Planned” for all.
- **At CDR:** some subsystem-level tests executed, status updated. Analysis-method requirements may show results from design calculations.
- **At FDR:** all requirements have a final status. The VCRM is the single-page executive summary of your entire V&V campaign.

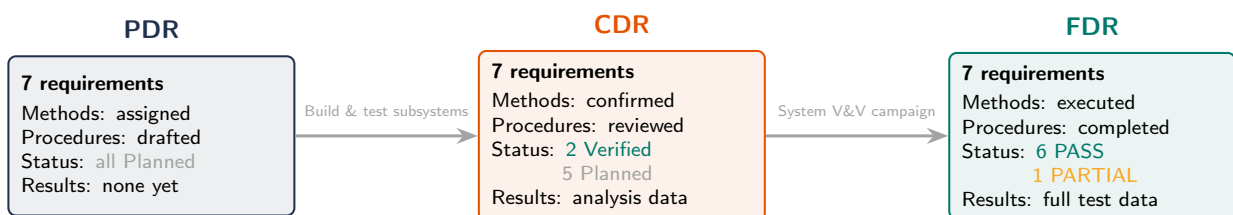


Figure 5.5: VCRM evolution across milestones. At PDR, all requirements are planned. At CDR, subsystem-level analysis and early tests fill in results. At FDR, every requirement has a final verdict with supporting data.

5.5.2 End-to-End Traceability Example

Stakeholder need: “I need to know when the greenhouse is too cold.” → **System requirement:** SYS-REQ-003: “The system shall send an SMS alert within 60 s of heater activation.” → **Verification method:** Test. → **Test procedure:** TP-003: Trigger heater activation, start stopwatch, verify SMS received within 60 s, record actual time for 3 trials. → **Result:** Trial 1: 12 s, Trial 2: 14 s, Trial 3: 11 s. Mean = 12.3 s. Verdict: PASS ($12.3 < 60$). → **VCRM entry:** SYS-REQ-003 | Test | TP-003 | PASS | 12.3 s mean.

This chain, need → requirement → method → procedure → result → VCRM, is what “requirements-driven design” means in practice.

5.6 V&V and the V-Model

The V-Model maps development phases (left side) to verification phases (right side): stakeholder needs ↔ validation, system requirements ↔ system testing, subsystem requirements ↔ integration testing, component specs ↔ unit testing. Implementation is at the bottom. Test planning happens on the LEFT side; design tests while writing requirements.

5.7 Statistical Confidence and Test Rigor

5.7.1 Developmental Testing vs. Production Qualification

How many times must you run a test? For **developmental testing** (MEGN 301): 3 repetitions minimum per test point. Report mean and range. For **production qualification** (industry): sample sizes from statistical requirements; typically $n \geq 30$ for parametric statistics, or MIL-STD-1916 sampling plans.

5.7.2 Practical Implications for Student Projects

Run each test 3 times, report all three values plus the mean, flag any result where the spread exceeds 20% of the mean. Report measurement uncertainty: if your instrument resolution is 0.1°C and your requirement is $\pm 0.5^\circ\text{C}$, you have adequate measurement capability. If your instrument resolution is 0.5°C , you cannot meaningfully verify a $\pm 0.5^\circ\text{C}$ requirement.

5.7.3 The Test Environment Matters

Test conditions that differ from operational conditions produce invalid results. Document and control: ambient temperature, humidity, power supply source (bench supply vs. battery vs. wall adapter; each behaves differently), electromagnetic environment (lab bench vs. field), and mechanical mounting (loose on bench vs. installed in enclosure). Any deviation from the intended operating environment must be noted in the test report with an assessment of impact.

5.7.4 Measurement Uncertainty

Every measurement has uncertainty. Your instrument’s resolution, accuracy, and calibration status all contribute. Rule of thumb: the measurement system should be $\geq 4\times$ more accurate than the requirement tolerance. If your requirement is $\pm 0.5^\circ\text{C}$, your measurement system must be accurate to $\pm 0.125^\circ\text{C}$ or better. Document measurement uncertainty in every test report.

5.8 Edge Case and Boundary Testing

Requirements specify ranges (“0 to 50 °C,” “0 to 100% RH”). Test at: the **nominal** operating point, **both boundary values**, and at least one point **outside** the boundary to verify graceful degradation (not catastrophic failure).

The Room-Temperature Trap

Testing only at room temperature ($\sim 22^\circ\text{C}$) is the most common V&V shortcut. If your requirement says 0–50 °C, you *must* test at 0 °C and 50 °C. Use the ME department environmental chamber or an ice bath + heat gun. A system that works at 22 °C but fails at 0 °C does not meet the requirement.

5.9 Minimum Viable V&V for MEGN 301

The minimum V&V package for student projects:

1. A VCRM with every requirement assigned a method and status.
2. A written test procedure (all 10 sections) for every T-method requirement.
3. Test results with actual measured data (not just “pass”).
4. For FAIL or PARTIAL PASS: root cause analysis and corrective action.
5. Peer review: another team executes at least one procedure. If they cannot reproduce results, the procedure is inadequate.

5.10 External Team Testing

Having another team execute your test procedures verifies reproducibility and uncovers ambiguities. **Preparing:** provide the procedure, equipment list, and safety briefing. Do NOT explain verbally; the document must stand alone. Observe without intervening (unless safety). Collect written feedback on every unclear step.

GS GreenShield: VCRM Fragment at Three Milestones

Req ID	Requirement	Method	At PDR	At CDR
SYS-REQ-001	Temp meas. at ≥ 1 Hz	T	Planned	In Progress
SYS-REQ-002	Heater at $\leq$ setpoint	T	Planned	PASS
SYS-REQ-003	SMS within 60 s	T	Planned	Planned
SYS-REQ-004	Autonomous mode on failure	D	Planned	Planned
SYS-REQ-005	Accuracy $\pm 0.5^\circ\text{C}$	T	Planned	PARTIAL

At **FDR**: all rows should show PASS, FAIL with Discrepancy Report, or waived with stakeholder approval. SYS-REQ-005 PARTIAL PASS (0.6 °C error at 0 °C boundary) requires resolution.

Industry SE Parallels

- **INCOSE SEH 4th ed.:** Sec. 4.10 (Verification), Sec. 4.11 (Validation).
- **NASA SP-2016-6105 Rev 2:** Ch. 5.3 (Verification), Ch. 5.4 (Validation).
- **IEEE 1012:** Standard for System, Software, and Hardware V&V.

5.11 Artifact Handoff: What Chapter 5 Produces

Artifact	Description	Forward Reference
VCRM	Cross-reference matrix linking reqs to test status	CDR, FDR deliverables
Test Procedures	Reproducible procedures for every T-method requirement	Ch. 5 (integration test)
Test Reports	Results with data, verdict, and any DRs	FDR documentation

5.12 Test Method Selection: T-A-I-D

Every requirement must be verified by one of four methods:

Test (T): direct measurement of the parameter under controlled conditions. Most rigorous; provides quantitative evidence. Use for: performance requirements (accuracy, speed, power consumption), safety requirements (overcurrent protection, thermal shutdown), and environmental requirements (temperature range, vibration).

Analysis (A): mathematical or computational evaluation. Use when testing is impractical (structural stress analysis before building, thermal simulation of extreme conditions, reliability prediction from component data). Analysis is only as good as the model; document all assumptions.

Inspection (I): visual or dimensional examination. Use for: physical characteristics (weight, dimensions, color, marking), material identification (marked per specification), and workmanship (solder quality, wire routing, labeling).

Demonstration (D): qualitative exercise of the system's functions without quantitative measurement. Use for: user interface verification (the display shows the correct information), operational procedures (the system starts up correctly), and safety demonstrations (the emergency stop works).

Selection guideline: use Test whenever feasible. Use Analysis when testing is impossible or prohibitively expensive. Use Inspection for attributes that can be directly observed. Use Demonstration for qualitative or pass/fail functions.

5.13 Designing for Testability

Design decisions made early in the project determine whether the system can be effectively tested:

Test points: include physical access points for measuring voltages, currents, and signals at key locations in the circuit. A test header (pin row connected to critical signals) allows an oscilloscope to be quickly attached during debugging and formal testing.

Calibration access: if the system requires calibration (sensor offset, gain adjustment, PID tuning), provide a software interface (serial command, button sequence, or configuration file) for entering calibration parameters without modifying firmware.

Diagnostic modes: firmware that can output raw sensor values, intermediate calculations, and actuator commands to a serial port enables rapid diagnosis of integration problems. Include a “debug” mode that bypasses the normal user interface and shows engineering-level data.

Stimulus injection: the ability to inject known test signals (simulated sensor readings, forced actuator commands) enables testing individual subsystems without the full system connected. In firmware, a “simulation” mode that replaces real sensor data with predefined test values is invaluable.

5.14 Writing Test Procedures

A test procedure must be detailed enough that someone who did not design the system can execute the test and obtain the same result. Each procedure includes:

Objective: what requirement is being verified, and what will be measured.

Equipment: list every instrument by type and calibration status (e.g., “Fluke 87V DMM, Cal. due 2026-06-15”).

Setup: how to configure the system and instruments, including specific settings (DAQ range, sample rate, trigger mode).

Procedure: numbered steps. Each step is a single action: “Set the power supply to 12.0 V, current limit 2.0 A.” “Record the temperature sensor reading in Table 1.” “Wait 60 s for the system to reach steady state.”

Data recording: a pre-formatted table or form for recording results, including spaces for the tester’s name, date, environmental conditions, and equipment serial numbers.

Acceptance criteria: the quantitative pass/fail threshold. “PASS if the measured temperature is within $\pm 1.0^{\circ}\text{C}$ of the reference at all 5 setpoints. FAIL if any setpoint deviation exceeds 1.0°C .”

Safety: any hazards and required precautions. “High voltage present when cover is removed. Do not touch exposed terminals.”

5.15 Practice Questions

1. Explain how a system can pass all verification tests but fail validation. Give a concrete example.
2. Classify each requirement by verification method (T/A/I/D): (a) “Shall withstand 50 N load without permanent deformation.” (b) “Shall display the company logo on the front panel.” (c) “Shall complete a power-on sequence within 10 seconds.” (d) “Shall have a thermal margin of $\geq 20^{\circ}\text{C}$ at max power under worst-case ambient.”
3. Write a complete test procedure (all 9 sections) for verifying: “SYS-REQ-007: The system shall measure ambient temperature with accuracy of $\pm 0.5^{\circ}\text{C}$ over 0 to 50°C .”
4. A test result shows 0.6°C error against a requirement of $\pm 0.5^{\circ}\text{C}$. What is the verdict? Describe the three possible root causes and the investigation sequence.
5. Your VCRM shows 18 PASS, 2 FAIL, and 1 PARTIAL PASS out of 21 requirements at FDR. What do you present to the stakeholder?

5.15.1 Answers

1. A greenhouse monitoring system meets all temperature accuracy, data rate, and communication requirements (verified). But the stakeholder discovers the display is unreadable in direct sunlight; a need that was never captured as a requirement. The system was built right (verification), but it was not the right system (validation failed). This is a requirements-capture failure.
2. (a) Test; apply 50 N load and measure deformation. (b) Inspection; visually confirm logo presence and placement. (c) Demonstration; power on and time the sequence. (d) Analysis; thermal simulation under worst-case conditions.

3. Purpose: verifies SYS-REQ-007. Setup: NIST-traceable temperature bath ($\pm 0.05^\circ\text{C}$), calibrated reference thermometer, DUT powered and warmed 30 min. Preconditions: DUT firmware v2.1, ambient $23 \pm 3^\circ\text{C}$. Steps: (1) Set bath to 0.0°C , wait 10 min, record DUT and reference. (2) Repeat at 10, 20, 30, 40, 50°C . Pass/fail: $|\text{DUT} - \text{reference}| \leq 0.5^\circ\text{C}$ at every point. Data: table with columns for setpoint, reference, DUT, error. Safety: bath uses water, standard lab PPE. Signatures: tester, witness, date.
4. Verdict: FAIL ($0.6 > 0.5$). Investigation: (1) Check test: was the reference bath stable? Was the reference thermometer calibrated? Repeat with verified equipment. (2) Check implementation: sensor calibration coefficients, ADC linearity, thermal coupling. (3) Check requirement: was $\pm 0.5^\circ\text{C}$ justified? If the application can tolerate $\pm 1.0^\circ\text{C}$, a requirements change (with stakeholder approval) may be appropriate. Open a DR regardless.
5. Present the VCRM showing all 21 requirements with status. For the 18 PASS items, show evidence summaries. For the 2 FAIL items, present root cause analysis, corrective actions, and retest plan with timeline. For the PARTIAL PASS, present which criteria passed, which failed, risk assessment, and recommendation (fix, waive, or defer). The stakeholder decides on the partial pass and accepts or rejects the open items.

5.15.2 Answers (continued)

6. **External testing value:** When your team writes and executes its own test procedures, you unconsciously fill in gaps with tacit knowledge (“oh, you need to wait 30 seconds after power-on before reading the sensor”, but the procedure does not say that). An external team has no tacit knowledge. Every ambiguity, missing step, and unclear criterion is exposed. The external team’s feedback is a direct measure of your documentation quality.
7. **Boundary testing strategy:** For a requirement “ $0\text{--}50^\circ\text{C} \pm 0.5^\circ\text{C}$ ”: test at 0°C (boundary), 25°C (nominal), and 50°C (boundary). Also test at -5°C and 55°C to verify graceful degradation (system should report out-of-range, not crash or give false readings). For humidity requirement “ $0\text{--}100\%$ RH”: test at 0% (desiccant chamber), 50% (ambient), and 95% (humidity chamber; 100% causes condensation that is a separate failure mode). Report all data points, not just pass/fail.

Chapter 6

Subsystem Integration

When to Read This Chapter

Sprint 3: Skim the integration sequence and N^2 diagram sections to inform your CDR interface definitions. **Sprint 4:** Read the full chapter before starting subsystem demonstrations. **Sprints 5–6:** Return to the debugging and configuration management sections during system integration.

Integration Time Estimate

Integration always takes longer than expected. Plan for integration to consume 30–50% of total project time. If your Gantt chart shows one week for integration, double it. The most common schedule failure in student projects is underestimating integration effort. See Chapter 20 (p. 210) for the systematic debugging methodology you will need when integration reveals faults.

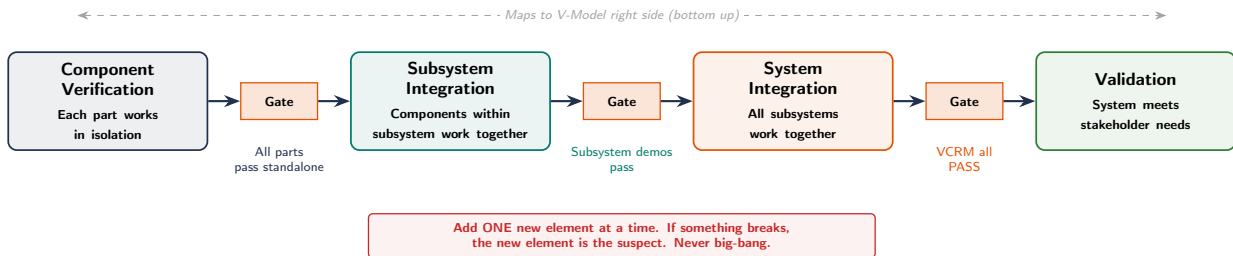


Figure 6.1: Bottom-up integration with test gates.

6.1 Integration Strategy

6.1.1 Bottom-Up Integration

Verify components individually → integrate into subsystems → test each subsystem → combine into system → system test. At each stage, add ONE new element and test. If something breaks, the new element or its interface is the suspect. **Never big-bang** (assemble everything at once).

6.1.2 Integration Stages and the V-Model

The V-Model maps integration from bottom-up: **Component verification** (each part works alone), **Subsystem integration** (components within a subsystem work together), **System integration** (subsystems work together), **System verification** (system meets all requirements),

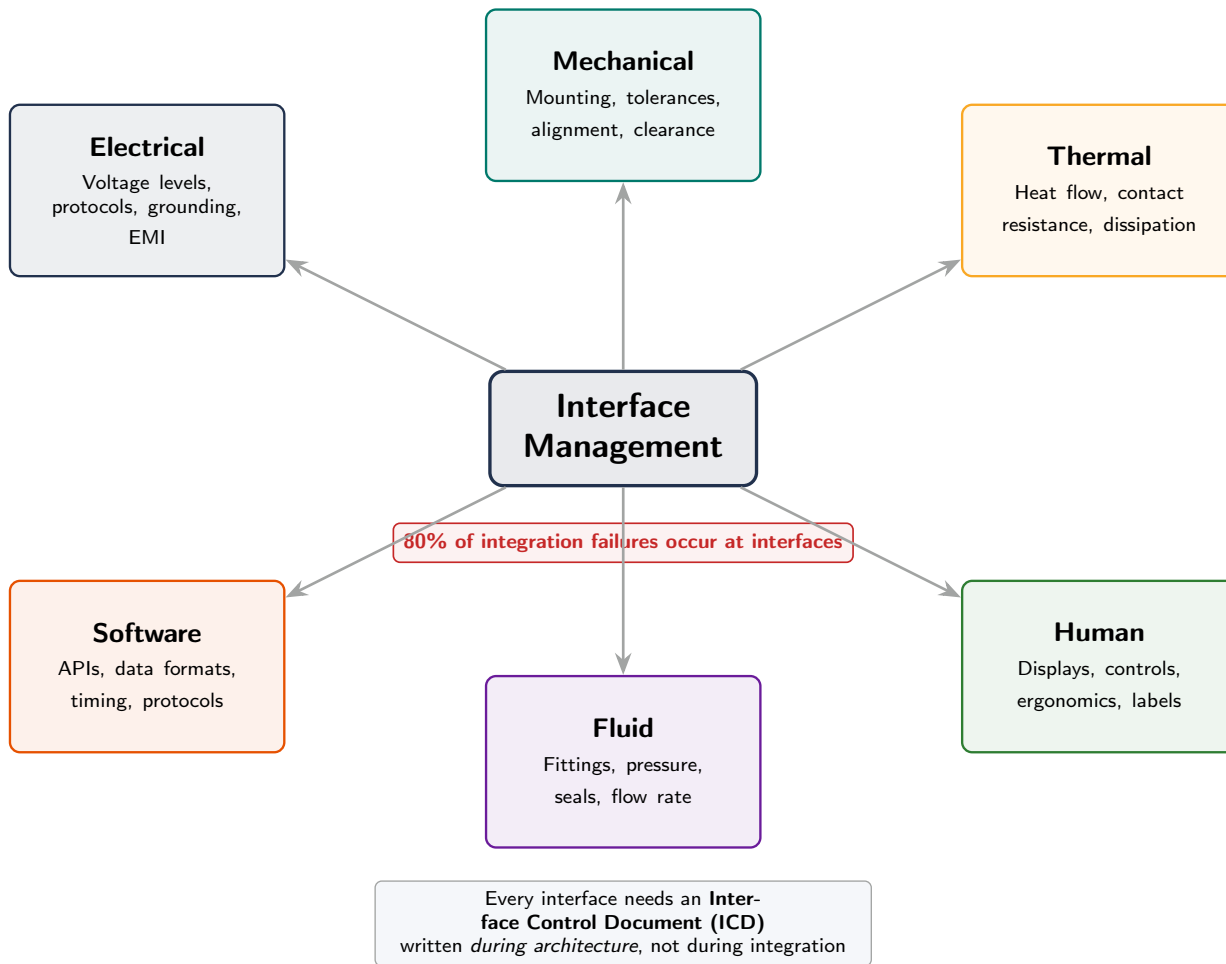


Figure 6.2: Interface types and common failure modes.

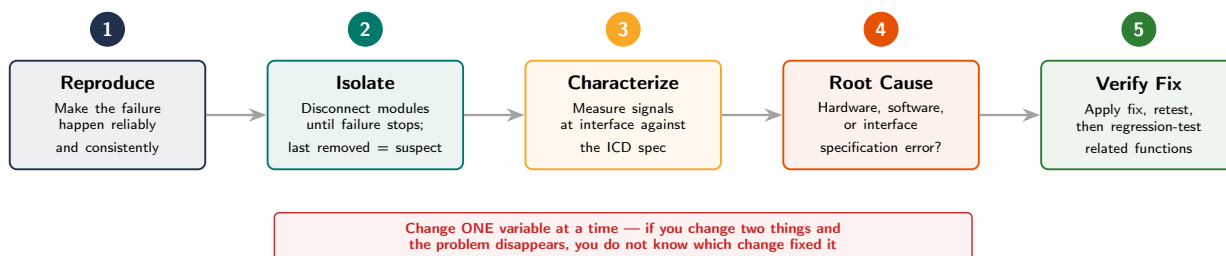


Figure 6.3: Five-step systematic debugging process.

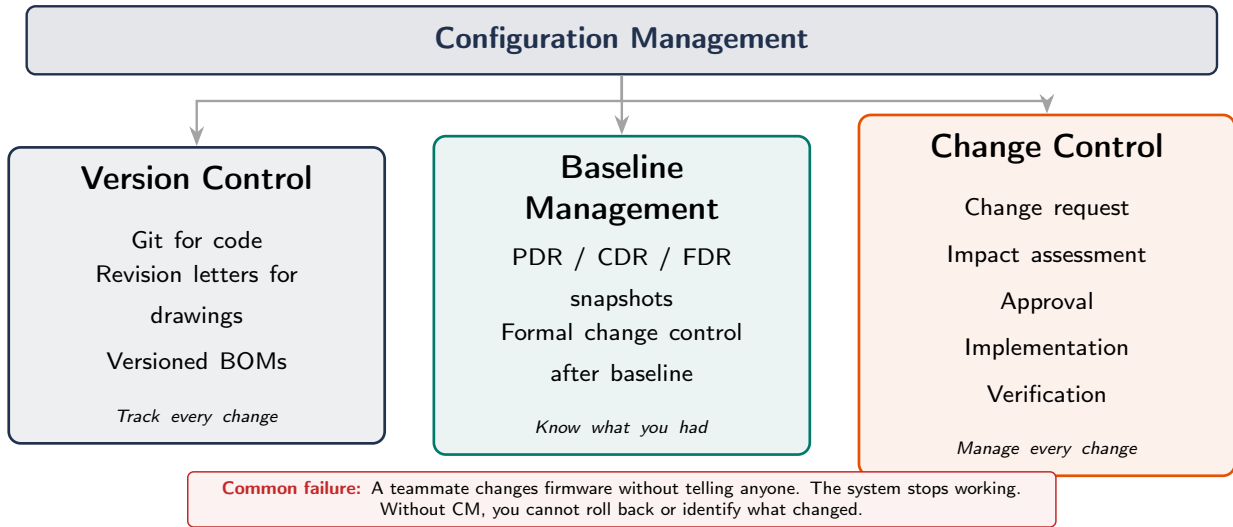


Figure 6.4: Configuration management pillars.

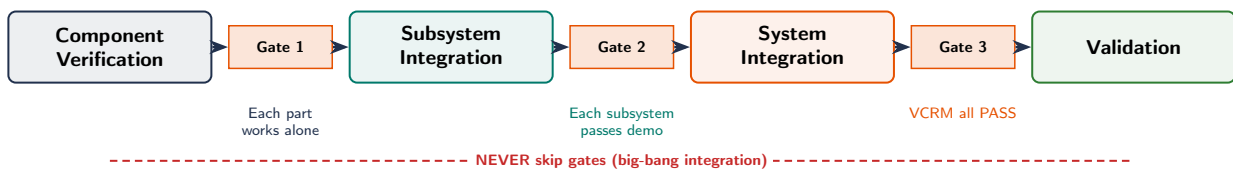


Figure 6.5: Bottom-up integration sequence with test gates. Each gate requires all tests at the current level to pass before advancing. Skipping gates (big-bang integration) multiplies debugging time by 5–10×.

Validation (system meets stakeholder needs). Each stage has specific entry and exit criteria. Never advance with open failures.

6.1.3 Integration Sequence Planning

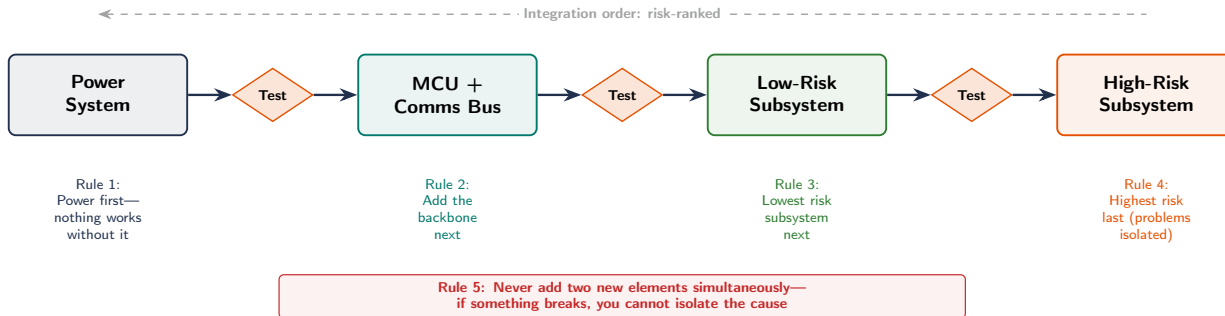


Figure 6.6: Integration sequence plan: power-first, risk-ordered subsystem build-up with test gates.

The order of integration matters (Figure 6.6). Rules: (1) Integrate power first; nothing works without power. (2) Add the lowest-risk subsystem next. (3) Add the highest-risk subsystem last (so problems are isolated to one new element). (4) Never add two new elements simultaneously (if something breaks, you cannot isolate the cause).

6.1.4 Test Gates

Each integration stage has entry and exit criteria. Entry: all components from the previous stage verified. Exit: all tests at the current stage PASS. Never advance with open failures.

6.1.5 Critical Subsystem Demonstrations

Before full system integration, each subsystem must pass a standalone demonstration. Three steps: (1) **Plan**; define pass/fail criteria, (2) **Execute**; run with data recording (video for design review), (3) **Document**; 1-page summary with data, photos, verdict. This is a CDR deliverable.

6.1.6 The N² (N-Squared) Diagram

The N² diagram maps interfaces between subsystems. Modules on the diagonal; cell (i, j) = what module i sends to module j . Dense rows/columns = high integration risk.

GS GreenShield: N ² Diagram (5×5)					
	Sensor	MCU	Relay	Comms	PSU
Sensor	—	Temp (I2C)			
MCU	Config	—	GPIO	UART TX	
Relay		Status	—		
Comms		UART RX		—	
PSU	3.3V	3.3V	12V	3.3V	;

PSU interfaces with everything. MCU receives from everything. Both are high-risk integration points. The I2C bus requires defined pull-up resistor ownership.

GS GreenShield: I²C Pull-Up Failure

The sensor team assumed the MCU board had pull-ups. The MCU team assumed the sensor breakout had them. Neither did. I²C floated, communication was intermittent, 8 hours of debugging. **Fix:** every ICD must state which side provides pull-ups, termination resistors, and bias voltages.

Case Study: Diagnosing Brown-Out Blues

A greenhouse controller reset randomly on relay activation. Scope showed 800 mV dip on 3.3 V rail (below 2.7 V brown-out threshold). Root cause: relay coil inrush through shared power trace. Fix: 470 μ F cap on MCU rail + separate relay power trace. Regression: 100 relay cycles, zero resets.

Industry SE Parallels

- **INCOSSE SEH 4th ed.:** §4.6 Implementation, §4.7 Integration.
- **NASA SP-2016-6105 Rev 2:** Ch. 5.1 Implementation, Ch. 5.2 Integration.

6.2 Artifact Handoff: What Chapter 6 Produces

Artifact	Description	Used In
Integration Plan	Build sequence with test gates	CDR
N ² Diagram	Interface matrix	ICD verification
Integration Test Results	Subsystem and system-level data	FDR

6.3 Interface Management

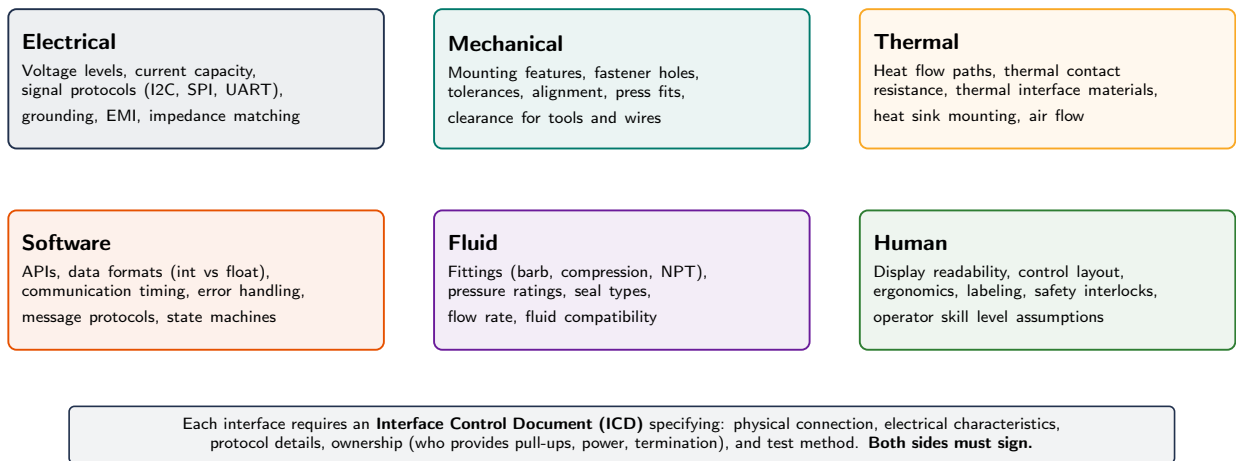


Figure 6.7: The six interface types: electrical, mechanical, thermal, software, fluid, and human.

80% of integration failures occur at interfaces (Figure 6.7). Six types: electrical (voltage, protocols, grounding), mechanical (mounting, tolerances, alignment), thermal (heat flow, contact

resistance), software (APIs, data formats, timing), fluid (fittings, pressure, seals), and human (displays, controls, ergonomics).

Write Interface Control Documents (ICDs) DURING architecture, not during integration. The ICD defines what crosses each boundary, in what format, with what constraints. Every interface needs a single owner.

6.3.1 Interface Control Documents (ICDs)

An ICD defines what crosses each boundary, in what format, with what constraints. Minimum ICD content for each interface:

1. **Interface name and number** (e.g., ICD-003: MCU ↔ Relay Driver).
2. **Physical connection**: connector type, pin assignments, wire gauge/color.
3. **Electrical characteristics**: voltage levels, current draw, impedance, timing.
4. **Protocol**: I2C address, baud rate, message format, error handling.
5. **Ownership**: which side provides power, pull-ups, termination.
6. **Test method**: how you verify the interface works (oscilloscope measurement, loopback test).

Every ICD must be signed by the owners of both sides of the interface. Unsigned ICDs are the #1 root cause of integration failures.

6.4 Debugging

Five-step process: (1) Reproduce the failure reliably. (2) Isolate by disconnecting modules until the failure stops. (3) Characterize by measuring signals at the interface against the ICD. (4) Root-cause: hardware, software, or interface spec? (5) Verify the fix, then regression-test. **Change ONE variable at a time.**

6.5 Configuration Management

Three pillars: **Version control** (Git for code, revision letters for drawings, versioned BOMs), **Baseline management** (PDR/CDR/FDR snapshots with formal change control after baseline), **Change control** (change request → impact assessment → approval → implementation → verification).

6.5.1 Critical Subsystem Demonstrations

Before full integration, each subsystem should pass a standalone demonstration. Three steps: (1) **Plan**: define what the subsystem demonstrates, test equipment needed, pass/fail criteria. (2) **Execute**: run the demonstration with data recording; video for design review. (3) **Document**: 1-page summary with data, photos, verdict; a CDR deliverable.

6.6 The Integration Challenge

Subsystem integration is where student projects most commonly fail. Each subsystem may work perfectly on the bench in isolation, but when connected together, unexpected interactions emerge: electrical noise from the motor corrupts sensor readings, the mechanical frame flexes under load and misaligns optical sensors, the firmware timing that worked with one sensor fails when three

sensors share the same bus, and the power supply that was adequate for bench testing sags under full system load.

The root cause is almost always insufficient interface definition during design. If the interfaces between subsystems are not precisely specified (voltage levels, timing, mechanical tolerances, thermal limits, EMI emissions), integration becomes a process of discovering and fixing problems rather than a planned assembly.

6.7 Integration Order Strategy

Bottom-up integration: build and test the lowest-level components first, then assemble them into subsystems, then integrate subsystems into the full system. Advantages: problems are isolated early. Disadvantages: the full system is not tested until the end.

Top-down integration: start with the system-level controller and add subsystems one at a time, using stubs (simulated inputs/outputs) for subsystems not yet available. Advantages: the system architecture is validated early. Disadvantages: stubs may not represent real subsystem behavior.

Recommended for MEGN 301: a hybrid approach. First, integrate the “backbone” (power supply + MCU + communication bus). Then add subsystems one at a time, testing each integration point before adding the next. If a problem appears, it is in the most recently added subsystem or its interface.

6.8 Interface Verification

Before connecting two subsystems, verify each interface independently:

Electrical interfaces: measure the voltage, current, impedance, and timing at each connection point. Verify that the driving subsystem can supply what the receiving subsystem requires. Check for voltage level compatibility (3.3 V logic vs. 5 V logic), current capacity (the source can provide the load’s peak current), and signal integrity (rise time, ringing, overshoot within limits).

Mechanical interfaces: verify that mating features align within tolerance. Check that fastener holes are concentric, that press-fit dimensions match, and that there is clearance for wire routing and assembly tools. A 0.5 mm misalignment that is invisible on the CAD screen becomes a show-stopper during assembly.

Software interfaces: verify that data formats, communication protocols, timing, and error handling are consistent between subsystems. A common failure: one subsystem sends data as a 16-bit signed integer while the other expects a 32-bit float.

6.9 Integration Testing Protocol

After each subsystem is connected, execute a focused integration test:

(1) **Smoke test:** power on and verify no component overheats, no magic smoke escapes, and no fuses blow. Seriously: this catches reversed power connections, short circuits, and overcurrent conditions before they cause damage.

(2) **Communication test:** verify that the newly connected subsystem can exchange data with the controller. Send a known command; verify the expected response.

(3) **Functional test:** exercise the subsystem through its full operating range while monitoring all other subsystems for interference. Does the motor’s EMI corrupt the temperature sensor? Does the pump’s vibration affect the accelerometer?

(4) **Document:** record the test results, any anomalies, and the system configuration (firmware version, wiring, component values).

6.10 Debugging During Integration

When something does not work after integration, resist the urge to change multiple things simultaneously. Instead:

Isolate: disconnect the newly added subsystem. Does the problem persist? If yes: the problem was pre-existing (regression). If no: the problem is in the new subsystem or its interface.

Bisect: for the new subsystem, test each interface signal independently. Is the power supply correct? Is the communication working? Is the sensor reading plausible? Is the actuator responding?

Instrument: add measurement points (oscilloscope probes, serial print statements, LED indicators) to observe internal signals. The problem is almost always at an interface, and measuring the interface signals identifies which side is at fault.

6.10.1 Integration Engineer's Toolkit

Effective integration requires the right measurement tools at the bench. Know what each one does and when to reach for it:

Multimeter: first-reach tool. Measures DC/AC voltage, current, resistance, and continuity. Use for: verifying power rails, checking fuses, confirming wiring continuity. Limitation: cannot capture transients or timing.

Oscilloscope: essential for any time-domain problem. Use for: verifying PWM duty cycle and frequency, capturing motor inrush transients, checking I2C/SPI/UART waveforms, identifying noise and ringing on signal lines, measuring rise times. Learn to use triggering (edge, pulse width) and cursors for timing measurements.

Logic analyzer: multi-channel digital capture with protocol decode. Use for: debugging SPI/I2C/UART communication (shows exact bytes sent/received), verifying timing between multiple digital signals, capturing intermittent glitches that an oscilloscope misses. The Saleae Logic 8 (\$400) or its clones (\$15) are standard for student projects.

Bench power supply: not just a power source but a diagnostic tool. Use the current limit as a fault detector: set the limit to the expected current draw. If the circuit trips the limit immediately, there is a short. If it draws zero current, there is an open. The current readout tells you whether the system is behaving normally before you connect any other instruments.

Thermal camera or IR thermometer: identifies hot spots that indicate overcurrent, poor thermal design, or component failure. A component running 20 °C hotter than its neighbors warrants investigation. The FLIR ONE phone attachment (\$200) or a \$25 IR thermometer gun are adequate for student projects.

The Integration Time Estimate

Students consistently underestimate integration time. A common estimate: “we have 5 subsystems, each takes 1 day to build, so we need 5 days.” Reality: building takes 5 days, but integration and debugging takes 5–15 more days. Budget at least 30% of the total project time for integration and testing. If you have 15 weeks, reserve weeks 10–15 for integration, testing, and FDR preparation.

6.11 Interface Testing

Before connecting two subsystems, verify each interface independently using loopback testing:

Serial (UART) loopback: connect TX to RX on the same device. Send a known string; verify it is received correctly. This confirms the baud rate, data format, and driver are correct before connecting to the other device.

I2C bus scan: use the Arduino “I2C Scanner” sketch to verify that each I2C device on the bus responds at the expected address. Missing devices indicate wiring errors or address conflicts.

SPI echo: send a known byte sequence to the SPI slave and read back the response. Verify the clock polarity, phase, and bit order match the slave’s requirements.

Power rail verification: before connecting any subsystem to the power bus, measure the voltage and verify it is within the subsystem’s input range. A 5 V subsystem connected to a 12 V rail is instantly destroyed.

6.12 Integration Sequence for a Typical GreenShield Project

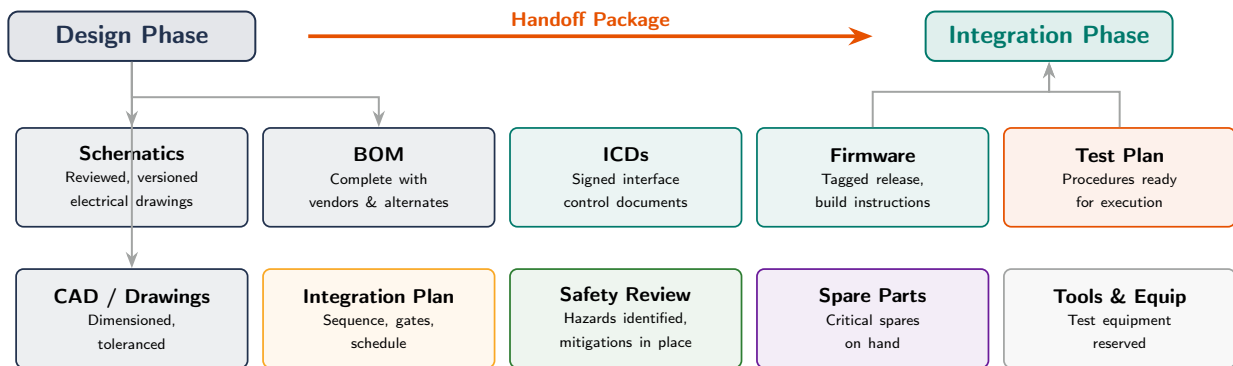


Figure 6.8: Integration handoff package: documentation artifacts that transfer from design to integration.

The integration handoff package (Figure 6.8) ensures documentation transfers cleanly from design to build.

Week 10: **Power system integration.** Connect the power supply to the distribution board. Verify all voltage rails. Connect the MCU and verify it boots. This is the “backbone” that everything else depends on.

Week 11: **Sensor integration.** Connect each sensor one at a time. For each: verify the raw ADC reading is plausible, apply the calibration equation, verify the engineering-unit output matches a reference measurement. If a sensor produces garbage data: check wiring, check the analog input mode (differential vs. RSE), check the ADC channel assignment in firmware.

Week 12: **Actuator integration.** Connect each actuator one at a time. For each: command a known output (50% PWM, fixed valve position), verify the physical response (motor spins, heater heats, valve opens). If an actuator does not respond: check the drive circuit (MOSFET gate voltage, relay coil current), check the firmware output pin assignment, verify the power supply can handle the load.

Week 13: **Control loop closure.** Connect the sensor feedback to the controller. Start with conservative gains (K_p at 20% of expected value, $K_i = 0$, $K_d = 0$). Verify the loop responds in the correct direction (negative feedback: increasing temperature reduces heater power). If the

loop runs away (positive feedback): the sensor polarity or the control sign is inverted. Gradually increase gains while monitoring stability.

Week 14: **System-level testing.** Execute the formal test plan. Document results. Address non-conformances.

Week 15: **FDR preparation.** Compile test results, finalize documentation, rehearse the presentation.

6.13 Practice Questions

1. Your system works with subsystems A and B connected, but fails when you add subsystem C. Describe your debugging approach step by step.
2. List all interfaces for a system with a sensor module, controller board, motor driver, and power supply. Which interfaces are highest risk?
3. Your teammate changes the firmware without telling anyone and the system stops working. What configuration management failure occurred?
4. Write entry and exit criteria for the subsystem integration stage of a greenhouse monitoring system.
5. Draw an N² diagram for a system with four modules: temperature sensor, ESP32 MCU, relay driver, and OLED display. Identify the highest-risk interface and explain why.
6. Your team skipped critical subsystem demonstrations at CDR. During integration, the motor driver outputs 5 V logic but the MCU expects 3.3 V. What process failure caused this, and how would a proper ICD have prevented it?

6.13.1 Answers

1. Debugging approach: (1) Reproduce: confirm failure occurs with A+B+C but not A+B. (2) Isolate: is it C itself or the C-to-A/C-to-B interface? Disconnect C from B, connect only C-to-A. Does it work? Then C-to-B interface is the suspect. (3) Characterize: measure signals at the C-to-B interface against the ICD. Check voltage levels, timing, protocol. (4) Root cause: is the interface spec wrong, or is C's output wrong? (5) Fix the specific issue, retest, then add B back.
2. Interfaces: Sensor↔Controller (I2C data + power), Controller↔Motor Driver (PWM signal + power), Motor Driver↔PSU (high-current power), Controller↔PSU (logic power), Sensor↔PSU (sensor power). Highest risk: Controller↔Motor Driver (PWM noise coupling, ground bounce from motor current spikes) and Motor Driver↔PSU (inrush current, ground loops).
3. Configuration management failure: no change control process. The teammate made an uncontrolled change. Fix: establish version control (Git for firmware), require peer review before merge, tag releases, and never deploy untested code to the integration hardware.
4. Entry criteria: all subsystem tests passed, all ICDs signed, all subsystem firmware at tagged release, test equipment available. Exit criteria: all subsystem-to-subsystem interface tests pass, system powers up and executes basic sequence, no brown-out or reset events in 30-minute soak test.

5. N² diagram: MCU column is densest (receives from all three others). MCU↔Relay Driver interface is highest risk because it crosses voltage domains (3.3 V logic to 12 V relay coil) and the relay switching causes noise on shared power lines.
6. The 5 V logic on a 3.3 V MCU input would have been caught by an ICD that specifies: “Motor driver PWM input: 3.3 V logic, active high, 1 kHz.” The ICD would have flagged the voltage incompatibility during the CDR documentation review, before any hardware was built.

Chapter 7

Operations & Maintenance

When to Read This Chapter

Sprint 3: Skim the lifecycle cost and serviceability sections; let them influence your CDR design decisions (connectorized modules, access panels, diagnostic LEDs). **Sprint 7:** Read the full chapter when preparing O&M deliverables for FDR. Write the quick-start guide, maintenance manual, and spare parts list. **Related:** FMEA from Chapter 3 (p.20) feeds directly into maintenance planning. EOL planning continues in Chapter 9 (p.82). FDR deliverables in Chapter 23 (p.230).

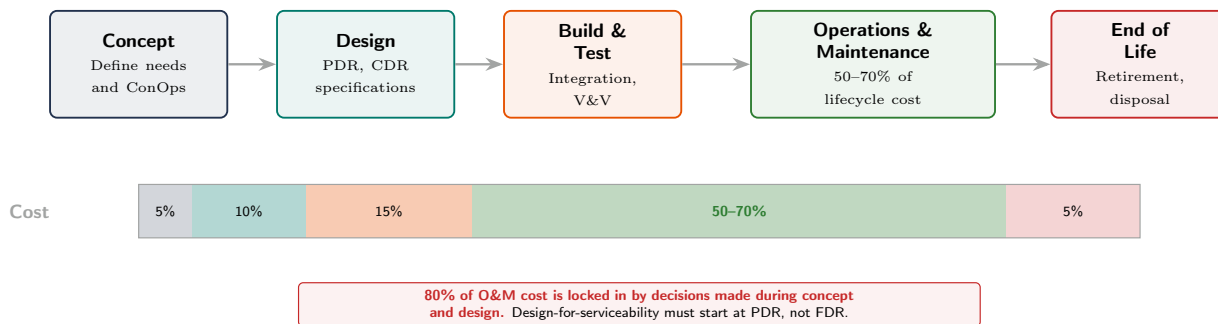


Figure 7.1: System lifecycle with cost distribution.

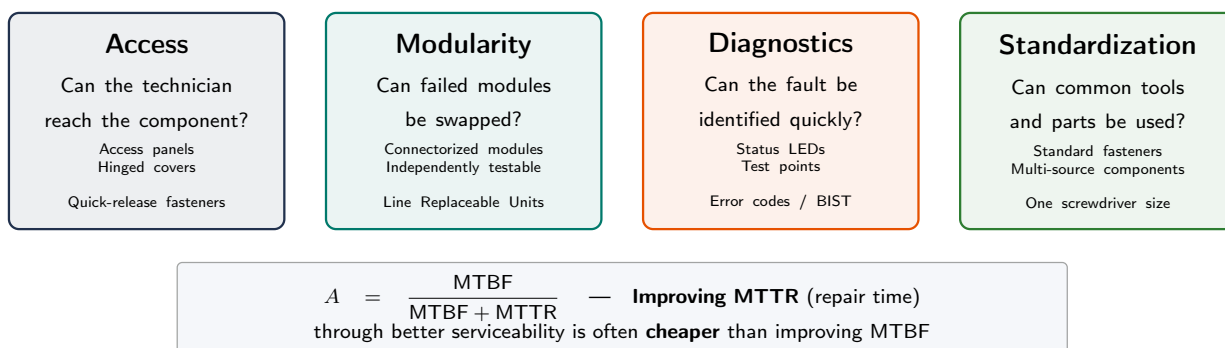


Figure 7.2: Four serviceability principles.

7.1 Lifecycle Thinking

Operations and maintenance accounts for 50–70% of total lifecycle cost, far more than development. 80% of that cost is locked in by decisions (see Figure 7.5) made during concept and

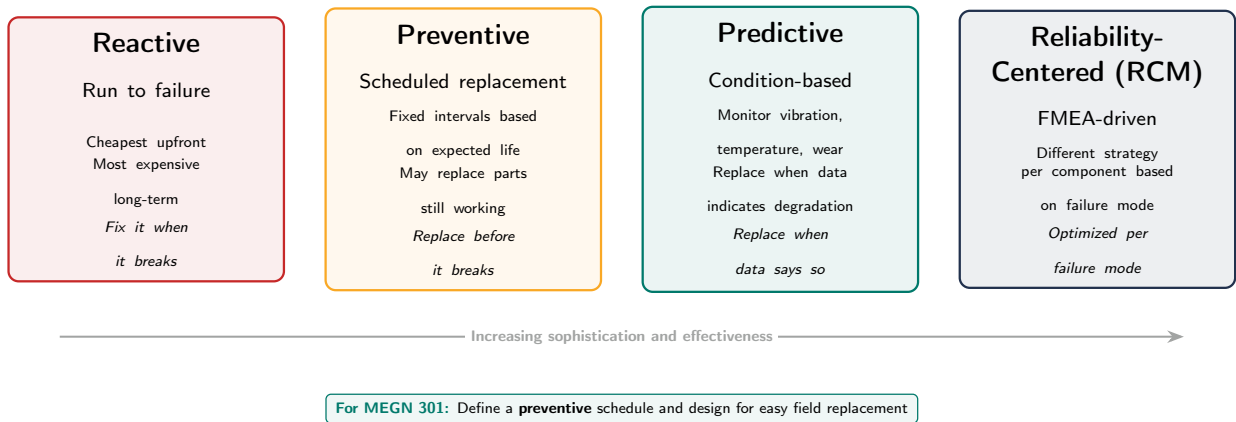


Figure 7.3: Maintenance strategies from reactive to RCM.

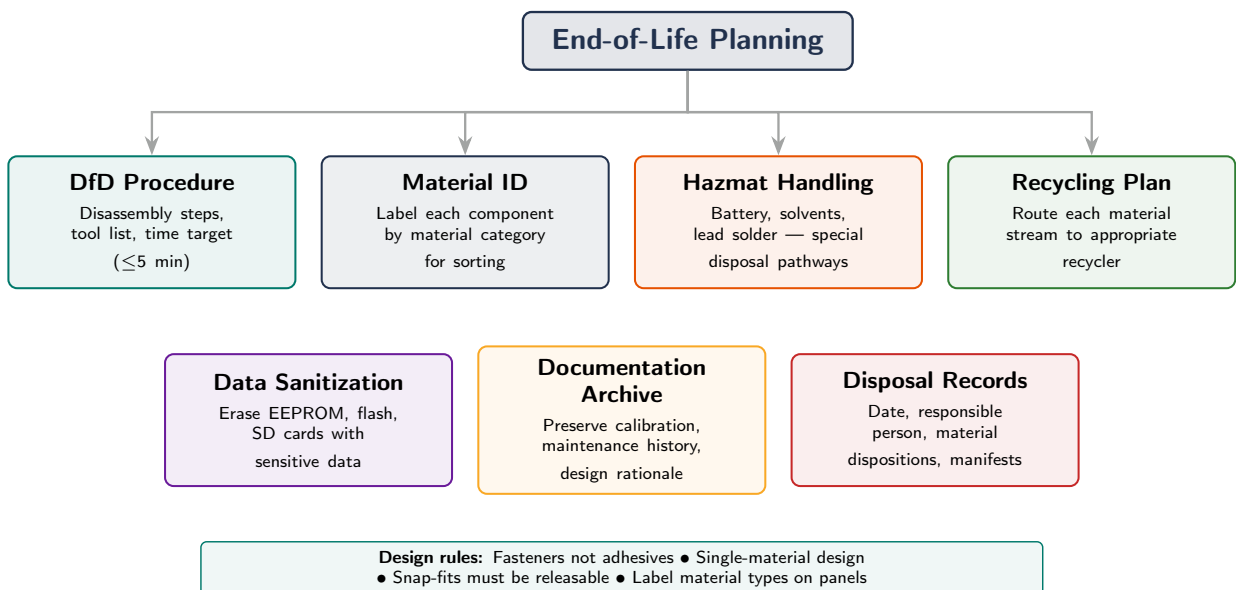


Figure 7.4: End-of-life planning deliverables.

preliminary design. This means design-for-serviceability, maintainability, and disposability must be considered DURING design.

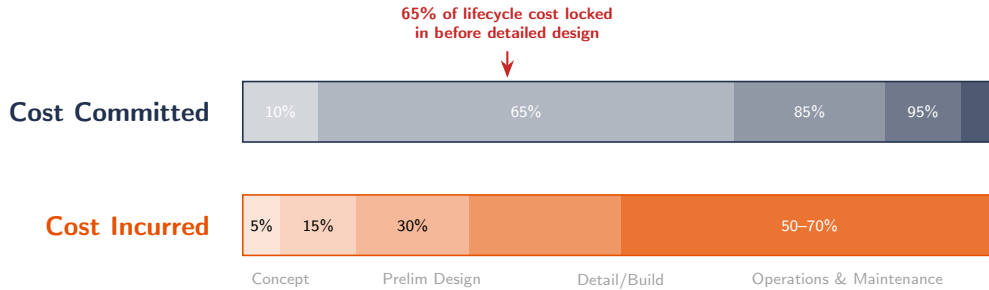


Figure 7.5: Lifecycle cost paradox: the majority of cost is *committed* during early design when only a small fraction has been *incurred*. By the time you are building, 85% of lifecycle cost is already determined by design decisions. This is why O&M thinking must start at PDR, not at FDR.

7.2 Design for Serviceability

Four principles: **Access** (can the technician reach the component? Use access panels, hinged covers, quick-release fasteners), **Modularity** (can failed modules be swapped? Connectorized, independently testable), **Diagnostics** (can the fault be identified? Status LEDs, test points, error codes, BIST), **Standardization** (can common tools and parts be used? Standard fasteners, multi-source components).

7.2.1 Reliability Metrics

MTBF (Mean Time Between Failures) and MTTR (Mean Time To Repair) combine to give Availability:

$$A = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}.$$

Improving MTTR through better serviceability design is often cheaper than improving MTBF through better components. FMEA identifies failure modes and prioritizes design improvements.

7.2.2 The Bathtub Curve

Component failure rate follows three phases: (1) **Infant mortality** (decreasing rate): manufacturing defects fail early. Mitigate with burn-in testing. (2) **Useful life** (constant rate): random failures at steady rate; the MTBF region. (3) **Wear-out** (increasing rate): fatigue, corrosion, seal degradation. Mitigate with preventive replacement.

For student projects, most failures occur in Phase 1 (build quality). Burn in your system for 48+ hours before final demo.

7.2.3 FMECA: Failure Modes, Effects, and Criticality Analysis

FMECA extends FMEA by adding **Criticality Analysis**. For MEGN 301, use simplified FMEA with RPN (Risk Priority Number = Severity × Occurrence × Detection, each 1–10). Focus design effort on top-5 RPN items.

7.2.4 Writing Maintainability Requirements

Examples of maintainability requirements:

- “The system shall allow sensor replacement without disassembling the enclosure.”
- “The system shall provide visual indication (LED) of power, communication, and fault status.”
- “The system shall allow firmware update via USB without opening the enclosure.”
- “All field-replaceable modules shall use tool-free connectors (no soldering).”

GS GreenShield: O&M Requirement Trace

Req ID	O&M Requirement	Parent Need
OM-001	Sensor replacement w/o enclosure disassembly	Operator cannot solder
OM-002	Status LEDs for power, comms, fault	Remote monitoring need
OM-003	Battery replacement <5 min	Off-grid solar concept

Industry SE Parallels

- **INCOSE SEH 4th ed.:** §4.12–4.15 (Transition, Operation, Maintenance, Disposal).
- **NASA SP-2016-6105 Rev 2:** Ch. 6.3 Reliability, Maintainability, and Availability.
- **MIL-STD-1629A:** FMECA procedures.

7.3 Artifact Handoff: What Chapter 7 Produces

Artifact	Description	Used In
O&M Manual	Quick-start, operating, maintenance, troubleshooting guides	FDR
FMEA/FMECA	Failure mode analysis with RPN ranking	Design iteration
Spare Parts List	Wear items with vendors and intervals	Handoff package

7.4 Maintenance Planning

Four strategies: **Reactive** (run to failure; cheapest upfront, most expensive long-term), **Preventive** (scheduled replacement at fixed intervals), **Predictive** (condition-based monitoring), **Reliability-Centered** (FMEA-driven strategy per component). For student projects: define a preventive schedule and design for easy field replacement.

Build a maintenance schedule: identify wear items, define replacement intervals, create a spare parts list with vendors and lead times, and establish a calibration schedule.

7.5 Documentation, Training, and End-of-Life

Four documentation types: Quick-Start Guide (1 page, 10-minute setup), Operating Manual (complete reference by task), Maintenance Manual (step-by-step procedures with photos), Troubleshooting Guide (symptom-based flowcharts).

Training plan: identify audiences (operators, maintainers, trainers), methods (hands-on, written, video), and competency assessment. Knowledge transfer: document design rationale, known limitations, and workarounds.

7.5.1 Design for Disassembly (DfD)

Sustainability begins with designing systems that can be taken apart. Write this as a formal requirement:

DfD-REQ-001: The system shall be fully disassembled into its base material categories (metal, polymer, electronics, hazardous) using only standard hand tools (Phillips screwdriver, hex key set, pliers) in ≤ 5 minutes by a single technician.

This is a **testable requirement**; verify it by demonstration (TAID method “D”) at FDR. Time the disassembly; if it exceeds 5 minutes, the design fails the requirement. Practical rules: use fasteners instead of adhesives, avoid mixed-material bonding (epoxy on sensors is fast but makes recycling impossible), label material type on each component or enclosure panel, and design snap-fits to be releasable (not permanently locked).

End-of-life: design for disassembly (fasteners not adhesives), single-material design for recyclability, hazardous material handling (batteries, solvents), data sanitization.

7.6 Operational Concepts

Every engineered system has an operational context: who uses it, where, how often, for how long, and under what conditions. The operational concept document (ConOps) describes this context and drives design decisions for reliability, maintainability, and user interface.

Duty cycle: how much of the time is the system operating? A continuously running process monitor (100% duty cycle) has very different reliability requirements than a laboratory instrument used 2 hours per week (1.2% duty cycle). High duty cycles demand higher-reliability components and shorter maintenance intervals.

Operating environment: indoor laboratory (controlled temperature, clean, stable power) vs. outdoor field installation (temperature extremes, moisture, vibration, unreliable power) vs. vehicle-mounted (high vibration, temperature cycling, EMI from engine). The environment determines the required IP rating, vibration qualification, temperature range, and EMC compliance.

Operator skill level: a research instrument operated by a PhD engineer can tolerate a complex interface and manual calibration procedures. A production tool operated by a technician must be intuitive with minimal training. A consumer product must be operable by anyone with no training. Design the interface for the least skilled expected operator.

7.7 Reliability Engineering Basics

Mean Time Between Failures (MTBF): the average time a system operates before a failure occurs. For electronic systems, MTBF is predicted from component failure rates using MIL-HDBK-217 or the FIDES methodology. The system MTBF is approximately the reciprocal of the sum of all component failure rates: $MTBF_{\text{system}} \approx 1 / \sum \lambda_i$.

Reliability: the probability that the system operates without failure for a specified time. For exponentially distributed failures: $R(t) = e^{-t/MTBF}$. For a system with $MTBF = 10,000$ hours, the probability of surviving 1000 hours without failure is $R = e^{-1000/10000} = 0.905$ (90.5%).

Availability: the fraction of time the system is operational: $A = MTBF / (MTBF + MTTR)$ where MTTR is the Mean Time To Repair. A system with $MTBF = 10,000$ hours and $MTTR = 2$ hours has availability = 99.98%.

7.8 Maintainability Design Principles

Accessibility: components that require periodic maintenance (filters, batteries, calibration adjustments) must be accessible without disassembling the entire system. Place maintenance items on the exterior or behind easily removed panels.

Standardized fasteners: use the minimum number of fastener types. Ideally, the entire system can be serviced with one screwdriver size. Avoid exotic fastener types (Torx, tri-wing, pentalobe) unless security is a requirement.

Modular replacement: design subsystems as Line Replaceable Units (LRUs) that can be swapped in the field without specialized tools or calibration. A failed sensor module should be replaceable by unplugging a connector, removing two screws, and plugging in a new module. Total repair time target: < 15 minutes for any single-point failure.

Diagnostic capability: include built-in self-test (BIST) features that identify the failed subsystem. An LED that blinks a fault code, a serial output that reports sensor status, or a display that shows system health enables rapid diagnosis. Without diagnostics, troubleshooting is trial-and-error, which is slow and risks introducing new problems.

Spare parts strategy: identify the components most likely to fail (wear parts, connectors, fuses) and stock spares. For MEGN 301 projects, include a “recommended spares” list in the maintenance manual.

7.9 User Documentation

A delivered system requires three documents:

User manual: how to operate the system. Step-by-step instructions for startup, normal operation, shutdown, and common adjustments. Written for the end user (non-engineer). Include photographs, not just text.

Maintenance manual: how to maintain and repair the system. Scheduled maintenance procedures (cleaning, calibration, consumable replacement). Troubleshooting guide (symptom → probable cause → corrective action). Exploded assembly diagrams showing part numbers and locations.

Technical reference: for the engineer who will eventually modify or upgrade the system. Complete schematics, firmware source code, mechanical drawings, BOM, calibration data, and design rationale. This is essentially the FDR package, archived for future reference.

7.10 Preventive Maintenance Scheduling

Based on component life data and operating conditions, create a maintenance schedule:

- **Daily:** visual inspection, check indicator lights, verify normal operation.
- **Weekly:** clean sensors and optical surfaces, check fluid levels, verify calibration against reference.
- **Monthly:** inspect wiring and connectors for damage, exercise manual overrides, backup data and firmware.
- **Annually:** replace batteries, recalibrate all sensors against traceable standards, inspect mechanical wear items (bearings, seals, belts), update firmware to latest version.

7.11 Failure Analysis and Root Cause Investigation

When a system fails in operation, a structured failure analysis prevents recurrence:

Preserve the evidence: do not immediately “fix” the failure. Photograph the system in its failed state. Record sensor logs, error messages, and environmental conditions at the time of failure. If a component physically failed (burned resistor, cracked solder joint, stripped gear), save the failed part for examination.

Fault tree analysis: work backward from the failure event to identify all possible causes. Draw a tree diagram with the top event (system failure) and branch into causes and sub-causes. For “motor stopped running”: branches include power supply failure (fuse blown, wire disconnected), motor failure (winding open, bearing seized), driver failure (MOSFET destroyed, control signal absent), and software failure (watchdog reset, state machine stuck).

5 Whys: for each branch of the fault tree, ask “why?” repeatedly until the root cause is identified. “Motor stopped” → why? “Bearing seized” → why? “Lubrication depleted” → why? “No lubrication schedule in maintenance plan” → root cause: inadequate maintenance documentation. The corrective action addresses the root cause (add lubrication to the maintenance schedule), not just the symptom (replace the bearing).

7.12 Warranty and Service Planning

For products intended for deployment beyond the classroom:

Warranty period: how long does the manufacturer (your team) guarantee the product will work? Industry standard for electronic products: 1–3 years. During the warranty period, the manufacturer repairs or replaces the product at no cost.

Service model: who performs maintenance and repair? Three models:

- (1) *Return to manufacturer:* the user sends the product back for repair. Longest downtime but ensures repairs are done correctly. Appropriate for complex, high-value products.
- (2) *Field service:* a trained technician visits the user’s site. Fastest repair but most expensive per incident. Appropriate for critical equipment that cannot be offline.
- (3) *User-serviceable:* the user performs routine maintenance and simple repairs using the maintenance manual. Lowest cost but requires adequate documentation and training. The GreenShield model: design for user serviceability with a clear maintenance manual.

7.13 Technical Data Package

At the end of the project, the team delivers a complete Technical Data Package (TDP) that enables anyone to reproduce, maintain, modify, and eventually retire the system:

- System Design Requirements Document (SDRD), final revision
- Mechanical drawings (CAD files, 2D drawings, assembly instructions)
- Electrical schematics and PCB layout files
- Bill of Materials with supplier links and current pricing
- Firmware source code with build instructions and version history
- Calibration procedure and data
- Test plan, test procedures, and test results (VCRM)
- User manual
- Maintenance manual with spare parts list
- Disassembly and disposal procedure (Chapter 9)

- Design rationale document (key decisions with justification)

The TDP is the team's professional legacy. A complete TDP demonstrates engineering maturity; an incomplete one demonstrates technical debt.

7.14 Obsolescence Management

Electronic components have shorter market lifetimes than the products they go into. An IC available today may be discontinued in 3–5 years. Strategies:

Design with widely used, multi-sourced components: the ESP32 has millions of units in production; it will be available for years. An obscure sensor from a single manufacturer may be discontinued without notice.

Document alternatives: for each critical component, identify at least one alternative that could be substituted with minor redesign. Record these in the BOM notes.

Lifetime buy: if a critical component is announced for end-of-life (EOL), purchase enough stock for the expected remaining production run. This is a common industrial practice but rarely relevant for MEGN 301 projects.

Modular design: isolating the component-specific interface in a replaceable module means that when a component becomes obsolete, only the module needs redesign, not the entire system.

7.15 Practice Questions

1. A system has $MTBF = 2000$ hrs and $MTTR = 24$ hrs. Calculate availability. Propose two design changes to improve availability, one targeting MTBF and one targeting MTTR.
2. Write a quick-start guide (1 page, numbered steps) for a greenhouse temperature monitoring system.
3. Your project sponsor asks why you budgeted \$200 for spare parts on a \$2000 project. Justify the expenditure.
4. Identify all wear items in an electromechanical system with a DC motor, belt drive, pressure sensor, and lithium battery. Define replacement intervals for each.
5. Sketch the bathtub curve and label all three regions. Your system fails twice in the first week then runs flawlessly for 6 months. What region were the initial failures in, and how should you have prevented them?
6. Write a simplified FMEA (Severity/Occurrence/Detection, 1–10 each) for a greenhouse frost protection system with at least 5 failure modes. Calculate RPN for each.
7. Your sponsor asks why you need an O&M manual for a one-semester student project. Argue why it is still necessary.

7.15.1 Answers

1. $Availability = MTBF / (MTBF + MTTR) = 2000 / (2000 + 24) = 98.8\%$. Improving MTBF: use industrial-grade components with higher rated life; add redundancy (dual sensors). Improving MTTR: add diagnostic LEDs and test points for faster fault isolation; use modular, connectorized design so a failed module can be swapped in <5 minutes instead of 24 hours.

2. Quick-start guide: (1) Unbox system. (2) Mount sensor probe inside greenhouse at plant height using provided bracket. (3) Connect sensor cable to controller box (keyed connector). (4) Plug wall adapter into 120 VAC outlet. (5) Verify green POWER LED is on. (6) Set temperature threshold using the dial on the front panel (default: 3 °C). (7) Verify amber MONITORING LED blinks once per second. (8) Test: hold ice cube near sensor. Within 30 s, red HEATER LED should illuminate. Setup complete.
3. The \$200 spare parts budget is insurance against project failure. Motors wear out, sensors break, PCBs develop faults during testing, and connectors get damaged during repeated assembly/disassembly. Without spares, a single component failure can halt the project for 1–3 weeks (ordering + shipping). At $\$200/\$2000 = 10\%$ of project cost, this is standard engineering practice. Industry typically budgets 5–15% of project cost for spares and consumables.
4. Wear items: DC motor brushes (replace at 2000–5000 hrs), belt (inspect at 500 hrs, replace at 1000 hrs or on visible cracking), pressure sensor diaphragm (inspect annually, replace at first sign of drift), lithium battery (capacity degrades; replace when capacity <80% of rated, typically 300–500 charge cycles). Additionally: all O-rings and seals in fluid connections (annual replacement), and any connectors subject to repeated mating cycles.
5. Bathtub curve: decreasing (infant mortality), constant (useful life), increasing (wear-out). Two failures in week one = infant mortality phase. Prevention: (a) burn-in test for 48+ hours before deployment, (b) inspect all solder joints and connections during assembly, (c) verify all power rails under load before system-level testing. The flawless 6-month period = useful life phase with constant (low) failure rate.
6. FMEA example (abbreviated): Sensor reads high (S=8, O=3, D=4, RPN=96), Relay stuck closed (S=9, O=2, D=6, RPN=108), MCU brown-out (S=7, O=4, D=5, RPN=140), Cellular no signal (S=5, O=3, D=3, RPN=45), Power supply fails open (S=8, O=2, D=7, RPN=112). Action priority: MCU brown-out first (decoupling + supervisor), then power supply (add status LED + fuse), then relay (add thermal fuse), then sensor (add second sensor for comparison).
7. The O&M manual is necessary because: (a) it is a primary FDR deliverable, the project is not complete without it; (b) the instructor and future teams need to operate, maintain, and build upon your work; (c) writing the manual forces you to think about failure modes, maintenance procedures, and spare parts, improving the design; (d) in industry, a product without O&M documentation is undeliverable.

Chapter 8

Post-Build Testing, Verification & Validation

When to Read This Chapter

Sprint 5: Read the full chapter when executing system-level V&V. Focus on the test execution workflow, non-conformance reports, and root cause analysis. **Sprint 6:** Use the external testing sections to prepare your handoff package. **Sprint 7:** Return to the documentation sections when compiling FDR evidence. **Related:** test planning in Chapter 5 (p. 46); debugging methodology in Chapter 20 (p. 210).

8.1 Why Testing Is a Separate Engineering Activity

Building the system is not the end of the engineering process; it is the beginning of a critical phase that determines whether the system actually works. Testing reveals the gap between what you designed and what you built. Verification asks: “did we build the product right?” (does it meet its specifications). Validation asks: “did we build the right product?” (does it satisfy the user’s actual needs). Both must be answered affirmatively before the system is ready for deployment.

In MEGN 301, this chapter addresses what happens after your GreenShield prototype is assembled and powered on for the first time. The activities described here occur between subsystem integration (Chapter 5) and the Final Design Review (FDR), and they produce the evidence that the FDR review board needs to assess your project’s success.

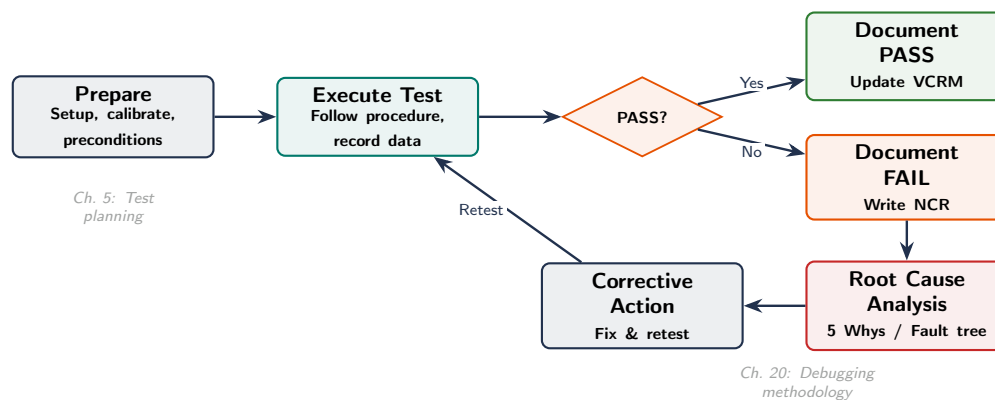


Figure 8.1: Post-build test execution workflow. Each requirement follows this cycle: prepare the test setup, execute the procedure, evaluate the result, and either document a PASS (updating the VCRM) or a FAIL (triggering root cause analysis and corrective action before retesting). Non-conformance reports (NCRs) document every failure and its resolution.

8.2 Test Planning and Documentation

8.2.1 The Test Plan

A test plan is a document (see the execution workflow in Figure 8.1) that specifies what will be tested, how, by whom, with what equipment, and what constitutes pass/fail. It should be written during the design phase (Chapter 4) and refined during integration. Key elements:

Test ID: unique identifier (e.g., TVR-007 for Test/Verification/Requirement 007).

Requirement traced: the specific “shall” statement being verified (from the SDRD).

Test method: one of four categories from systems engineering: Test (T), Analysis (A), Inspection (I), or Demonstration (D).

Test procedure: step-by-step instructions that another engineer could follow to reproduce the test. Include setup, equipment list, input conditions, measurement points, and acceptance criteria.

Pass/fail criteria: quantitative thresholds derived directly from the requirement. “The motor shall produce at least 0.5 N·m of torque at 500 RPM” becomes “PASS if measured torque ≥ 0.5 N·m at 500 ± 10 RPM; FAIL otherwise.”

Test data record: a template for recording the raw data, environmental conditions, equipment serial numbers, and tester’s name. This is the evidence file for the FDR.

8.2.2 Verification Cross-Reference Matrix (VCRM)

The VCRM is a table that maps every requirement in the SDRD to its corresponding test. It answers: “for each requirement, how do we know it is met?” A gap in the VCRM (a requirement with no corresponding test) means that requirement is unverified; it could be met or not, and nobody knows. An incomplete VCRM is one of the most common deficiencies found at FDR.

GS GreenShield Template: VCRM Format

Req ID	Requirement	Method	Test ID	Status
SYS-001	Shall operate on 12 V DC	T	TVR-001	PASS
SYS-002	Shall weigh <2 kg	I	TVR-002	PASS
SYS-003	Shall maintain 50 ± 1 °C	T	TVR-003	FAIL (see NCR-01)
Any FAIL result must be documented in a Non-Conformance Report (NCR) with root cause analysis and corrective action plan.				

8.3 Functional Testing

Functional testing verifies that each function specified in the requirements is operational. It is the most basic level of testing: does the system do what it is supposed to do?

Power-on self-test (POST): verify the system initializes correctly. Check: power LED illuminates, MCU boots (serial output or indicator), all sensors report plausible values (not NaN, not railed), all actuators respond to a brief test command (motor spins for 0.5 s, LED blinks, valve clicks). The POST should be the first thing you run after integration, and it should be automated in firmware.

Input-output verification: for each sensor, apply a known input and verify the output is within the calibrated range. For each actuator, command a known output and verify the physical

response. This confirms that the wiring, signal conditioning, firmware, and calibration are all correct.

Nominal operation test: run the system through its normal operating sequence (the “happy path”). For a temperature controller: set the target, wait for the system to reach setpoint, verify stability, change the setpoint, verify the transition. Document the results with time-stamped data and photographs.

8.4 Performance Testing

Performance testing measures how well the system performs its functions. It goes beyond “does it work?” to “how well does it work?”

Accuracy testing: compare the system’s output to a calibrated reference across the operating range. For a temperature controller: measure the actual temperature with a calibrated reference thermometer at 5–10 setpoints spanning the range. Plot the error (system reading minus reference) vs. setpoint. The maximum error must be within the accuracy specification.

Repeatability testing: repeat the same measurement or operation 10–20 times under identical conditions. Compute the standard deviation. The repeatability must be within the specification. If the system meets the accuracy spec on average but has high scatter (poor repeatability), it is unreliable even if individual tests happen to pass.

Response time testing: measure the time from a step input (setpoint change, disturbance) to the output reaching the specified percentage of its final value (typically 90% or 95%). Compare to the response time specification. For control systems: measure rise time, settling time, and overshoot from the step response data.

Boundary and stress testing: operate the system at the extremes of its specified range. Test at the maximum and minimum input, the maximum and minimum power supply voltage ($\pm 10\%$ typical), the maximum and minimum ambient temperature, and the maximum load. Verify that the system still meets all specifications at these extremes. Systems that pass nominal testing but fail at boundaries have insufficient design margin.

8.5 Environmental and Robustness Testing

Vibration and shock: for portable or vehicle-mounted systems, shake or drop the system and verify continued operation. Even a brief vibration test (hand-shake the enclosure while monitoring sensor outputs) reveals loose connections, poorly secured components, and resonant structural elements.

Thermal cycling: if the system will operate in varying temperatures, cycle it between the temperature extremes 5–10 times and verify operation at each extreme. Thermal cycling stresses solder joints, connector contacts, and adhesive bonds.

EMI susceptibility: operate the system near a running motor, a switching power supply, or a cell phone transmitting. Monitor sensor outputs for noise, glitches, or erroneous readings. If the system is sensitive to EMI, improve shielding and grounding before deployment.

Water and dust exposure: if the system will operate outdoors or in industrial environments, test resistance to splash (IP rating). Even a brief spray test reveals unsealed enclosure joints, exposed connectors, and inadequate conformal coating.

8.6 Validation: Does It Solve the Problem?

Verification confirms the system meets its specifications. Validation confirms the specifications themselves are correct: that meeting them actually solves the user's problem. Validation is harder because it requires engagement with real users in realistic conditions.

User acceptance testing (UAT): give the system to a representative user (not a team member) and observe them operating it. Note: where they hesitate, what they do wrong, what they find confusing, and whether the outcome meets their expectations. The user's experience often reveals requirements that were missed or misinterpreted.

Field trial: operate the system in its intended environment for an extended period (days to weeks). Monitor for degradation, unexpected failure modes, environmental effects, and user feedback. The field trial often reveals issues that laboratory testing cannot: intermittent failures, long-term drift, user workflow conflicts, and environmental exposures not anticipated in the test plan.

Stakeholder review: present the test results to all stakeholders (user, sponsor, advisor, safety officer) and obtain explicit acceptance. The FDR serves this function in MEGN 301.

8.7 Non-Conformance and Corrective Action

When a test fails, the response follows a structured process:

(1) **Document the failure:** what was tested, what was expected, what was observed, and what the deviation is. This is the Non-Conformance Report (NCR).

(2) **Root cause analysis:** why did it fail? Use the "5 Whys" technique: ask "why" iteratively until the root cause (not just the symptom) is identified. "The motor stalled" → why? "Insufficient torque" → why? "Current limited by the MOSFET driver" → why? "Gate voltage only 3.3 V, MOSFET needs 5 V for full enhancement" → root cause: wrong MOSFET selected for 3.3 V logic.

(3) **Corrective action:** fix the root cause, not just the symptom. Replace the MOSFET with a logic-level type, not just increase the motor voltage.

(4) **Retest:** after the corrective action, retest the specific requirement that failed. Also retest any related requirements that could have been affected by the change (regression testing).

(5) **Close the NCR:** document the corrective action taken, the retest results, and the final disposition (PASS after rework, or Accept-as-is with justification, or Reject/Redesign).

8.8 Test Report and FDR Evidence Package

The test report is the primary evidence document for the FDR. It should include:

- Test plan (what was supposed to be tested)
- VCRM with status of each requirement (PASS/FAIL/UNTESTED)
- Raw test data with timestamps, equipment used, and environmental conditions
- Photographs and videos of key tests
- Analysis of any failed tests (NCRs with root cause and corrective action)
- Summary statistics (number of requirements, number verified, number of failures, number of open NCRs)
- Conclusion: overall system readiness assessment

Test What You Ship, Ship What You Test

The system presented at FDR must be the same system that was tested. Any modification made after testing (even a “minor” firmware update or a “quick” wiring fix) invalidates the test results for any requirement that could be affected by the change. If you make changes after testing, retest the affected requirements and update the test report.

8.9 Test Equipment for MEGN 301

The following equipment is available in the MEGN 301 lab and should be used for post-build testing:

Digital multimeter (DMM): measures DC/AC voltage, current, and resistance. Use for: verifying power supply voltages, checking continuity of wiring, measuring motor current draw, and validating sensor output ranges. Accuracy: $\pm 0.5\%$ for DC voltage, adequate for most checks.

Oscilloscope: visualizes time-varying signals. Use for: verifying PWM frequency and duty cycle, checking for noise on sensor signals, measuring signal rise times, debugging communication buses (UART, I2C, SPI), and observing motor drive waveforms. The oscilloscope reveals problems invisible to the DMM: ringing, noise spikes, ground bounce, and timing errors.

Function generator: produces known waveforms (sine, square, triangle) for input to the system. Use for: testing filter frequency response, verifying amplifier gain and bandwidth, and simulating sensor signals during software development (before the real sensor is available).

Power supply (bench): provides adjustable, current-limited DC voltage. Use for: powering the system during testing (before batteries are connected), varying the supply voltage to test robustness ($\pm 10\%$ supply variation), and current-limiting to prevent damage during initial power-on.

Reference instruments: calibrated thermometer, precision weights, calibrated pressure source, or other reference standards appropriate to the measurand. Use for: accuracy verification tests.

8.10 Automated Testing

For systems with many requirements or repetitive test procedures, automate the testing through firmware:

Built-in self-test (BIST): firmware routine that exercises each subsystem and reports pass/fail. Run BIST after every firmware update and before every demonstration. The BIST results are logged with timestamp and can be included in the FDR evidence package.

Automated data collection: for accuracy and repeatability tests, program the MCU to automatically cycle through setpoints, record measurements, and compute statistics. This eliminates operator variability and produces consistent, timestamped data.

Regression testing: after any design change (firmware update, component substitution, wiring modification), re-run the full automated test suite to verify that the change did not break previously passing requirements. This is called regression testing and is standard practice in software engineering.

8.11 Statistical Analysis of Test Results

Test results must be analyzed statistically to determine whether requirements are met:

Accuracy: compute the mean error (bias) and the standard deviation (scatter) across the calibration range. The total accuracy (95% confidence) is: $|\text{bias}| + 2s$. This must be within the accuracy requirement.

Repeatability: compute the standard deviation of N repeated measurements at the same condition. The repeatability (95%) is $2s$. Compare to the repeatability requirement.

Response time: from N step-response tests, compute the mean and standard deviation of the settling time. Report as mean $\pm 2s$. The upper bound (mean $+2s$) must be within the response time requirement to ensure 95% of responses meet the spec.

Pass/fail decision: a requirement is “PASS” only if the measured performance meets the specification with the stated confidence level. If the measured value is right at the specification limit (e.g., accuracy = 1.0°C against a $\leq 1.0^\circ\text{C}$ requirement), consider whether measurement uncertainty could place the true performance above the limit. A result that passes only because of favorable measurement noise is not truly passing.

8.12 Common Testing Mistakes

(1) **Testing in ideal conditions only:** the lab is 22°C , the power supply is clean, there is no vibration. The real environment may be different. Test at the specified environmental extremes.

(2) **Confirming instead of testing:** the tester expects a pass and interprets ambiguous results as passing. Use blind testing (the tester does not know which condition is being measured) or automated testing to eliminate bias.

(3) **Insufficient sample size:** one successful test does not prove the requirement is met. Perform ≥ 10 repetitions for statistical validity.

(4) **Not documenting failures:** a test that fails and is “fixed” without documentation is a lost learning opportunity and an audit risk. Every failure must be documented in an NCR.

(5) **Testing the demo, not the requirement:** the team demonstrates that the system “works” but does not verify the specific quantitative requirements. “It controls temperature” is not evidence that “it controls temperature to $\pm 1^\circ\text{C}$.”

8.13 Practice Questions

1. Your system has 25 requirements in the SDRD. You have completed testing on 20 of them (18 PASS, 2 FAIL). What is your verification coverage? What should you present at FDR regarding the 5 untested requirements?
2. A test reveals that your temperature controller maintains $50 \pm 2.5^\circ\text{C}$ against a requirement of $50 \pm 1^\circ\text{C}$. Write an NCR: document the non-conformance, propose a root cause, and suggest a corrective action.
3. Explain the difference between verification and validation using the GreenShield project as an example. Give one specific test for each.
4. Your system passes all laboratory tests but fails during the first outdoor field trial due to direct sunlight heating the enclosure above the operating temperature limit. What type of testing should have been included in the test plan to catch this?
5. Design a power-on self-test (POST) sequence for a system with: one temperature sensor, one motor, one pump, one LCD display, and an ESP32 microcontroller. List each check and the expected result.

8.13.1 Answers

1. Verification coverage = $20/25 = 80\%$. At FDR: present the 18 PASS results, the 2 FAIL results with NCRs and corrective action status, and a plan for completing the 5 untested requirements (schedule, resources needed). The FDR board may conditionally accept the design with a requirement to complete the remaining tests by a specified date.
2. NCR: Requirement SYS-TEMP-001 specifies $50 \pm 1^\circ\text{C}$. Measured: $50 \pm 2.5^\circ\text{C}$ over 30 minutes. Deviation: $1.5\times$ the allowable tolerance. Root cause: PID integral gain too low, allowing slow oscillation. Corrective action: increase K_i from 0.3 to 0.8 and add derivative action ($K_d = 2$). Retest after tuning.
3. Verification: “The system shall maintain temperature within $\pm 1^\circ\text{C}$ of setpoint.” Test: measure the temperature with a calibrated reference at the setpoint for 30 minutes. This verifies the specification is met. Validation: “The system shall keep the user’s beverage at a comfortable drinking temperature.” Test: give the system to 5 users with their beverages. Observe whether they find the temperature satisfactory. This validates that the specification itself is correct (maybe $\pm 1^\circ\text{C}$ at 50°C is not what users actually want; they might prefer 60°C).
4. Environmental testing: specifically, solar radiation (thermal loading) testing. The test plan should have included a test with a heat lamp simulating solar heating to characterize the enclosure temperature rise under direct sunlight. Additionally, the requirements should have specified the environmental conditions (max ambient + solar load), and the thermal design should have included sun shielding or ventilation.
5. POST sequence: (1) LCD: display “POST...” (visual confirmation display works). (2) Temperature sensor: read value; PASS if $10\text{--}40^\circ\text{C}$ (plausible room temperature); FAIL if <0 or >50 or NaN. (3) Motor: energize for 0.5 s at 20% PWM; PASS if current draw detected ($>50\text{ mA}$); FAIL if zero current (open circuit) or $>2\text{ A}$ (short). (4) Pump: energize for 0.5 s; PASS if current draw in expected range. (5) ESP32 Wi-Fi: scan for networks; PASS if ≥ 1 network found. (6) Display results: “POST PASS” (all green) or list failures in red.

Chapter 9

End of Life & System Retirement

When to Read This Chapter

Sprint 3: Skim the Design for Recycling and Design for Disassembly sections; let them influence material and fastener choices at CDR. **Sprint 7:** Read the full chapter when writing your FDR disassembly procedure and disposal plan. **Related:** FMEA from Chapter 3 (p. 20); O&M from Chapter 7 (p. 66); FDR deliverables in Chapter 23 (p. 230).

9.1 Why End of Life Matters

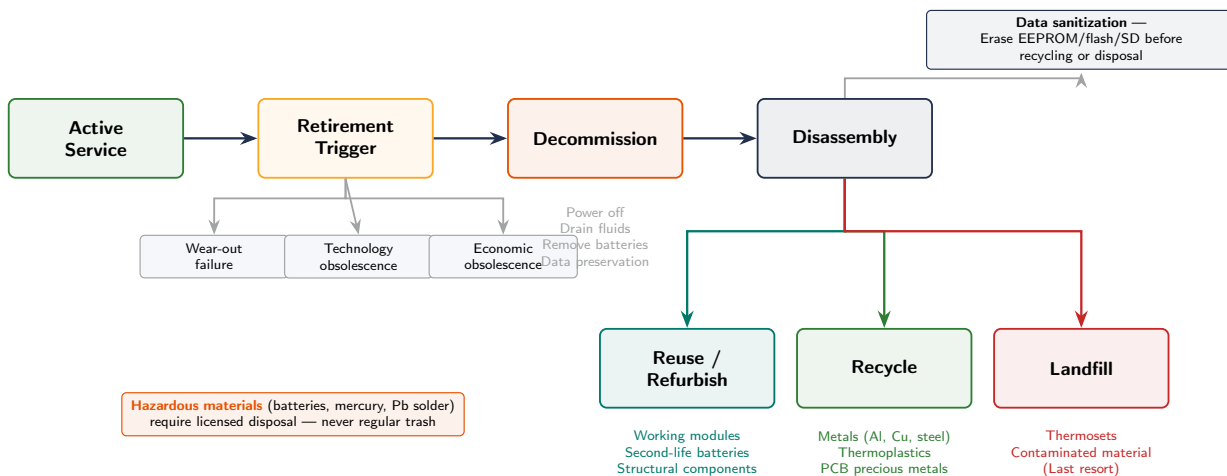


Figure 9.1: End-of-life pathways: from active service through retirement, disposal, and material recovery.

Every engineered system eventually reaches the end of its useful life (Figure 9.1). The decisions made during the design phase determine whether end-of-life (EOL) is a manageable, planned event or an expensive, hazardous crisis. Responsible engineering requires considering the complete life cycle: design, manufacture, use, maintenance, and retirement. This chapter addresses the final phase, which is the most commonly neglected in student projects and early-career engineering.

The European Union's Waste Electrical and Electronic Equipment (WEEE) Directive, the Restriction of Hazardous Substances (RoHS) Directive, and similar regulations worldwide make EOL planning a legal requirement for commercial products. Even for MEGN 301 prototypes, understanding EOL principles demonstrates professional maturity and prepares you for industry practice.

9.2 Failure Modes That Trigger Retirement

Systems are retired when they can no longer perform their intended function at acceptable cost or risk:

Wear-out failure: components degrade with use. Bearings develop play, seals harden and leak, brushes wear to nubs, solder joints fatigue from thermal cycling, and batteries lose capacity. Wear-out is predictable and can be mitigated by maintenance and part replacement, but eventually the cumulative degradation makes continued operation uneconomical.

Technological obsolescence: the system still functions but is outperformed by newer technology at lower cost. A 10-year-old data acquisition system with 12-bit resolution and 100 kS/s is functionally obsolete when current devices offer 24-bit and 1 MS/s for the same price. Obsolescence is driven by Moore's Law for electronics and by continuous improvement in manufacturing processes.

Component obsolescence: a critical component (IC, sensor, connector) is discontinued by the manufacturer and no equivalent replacement is available. This is increasingly common as component life cycles shorten. Designing with widely available, multi-sourced components extends the system's serviceable life.

Regulatory obsolescence: new safety, environmental, or performance regulations make the system non-compliant. A product containing lead-based solder became non-compliant in the EU after RoHS. A vehicle emissions system may become obsolete when emission standards tighten.

Economic obsolescence: maintenance costs exceed the cost of replacement. When annual repair costs approach 50% of replacement cost, retirement and replacement is typically more economical.

9.3 Design for End of Life (DfEOL)

DfEOL decisions made during the design phase dramatically affect EOL cost and environmental impact.

9.3.1 Design for Recycling (DfR)

Beyond disassembly, DfR ensures the separated materials can actually be recycled:

Avoid material contamination: a steel screw in an aluminum part contaminates the aluminum melt. Use same-material fasteners where possible, or use fasteners that are easily removable.

Avoid composite materials: fiber-reinforced plastics, metal-plastic laminates, and overmolded assemblies cannot be separated into their constituent materials and are generally land-filled.

Avoid hazardous materials: lead (solder), cadmium (some plating), mercury (certain switches and sensors), hexavalent chromium (some corrosion treatments), and brominated flame retardants (some PCB materials) are regulated by RoHS and WEEE and complicate recycling. Use lead-free solder (SAC305), RoHS-compliant components, and halogen-free PCB substrates.

9.3.2 Design for Remanufacturing

Remanufacturing returns a used product to like-new condition by replacing worn components while retaining the structural elements. It requires: standardized interfaces (so replacement parts

from the next generation fit), accessible wear parts (bearings, seals, brushes located for easy replacement), and robust structural elements (designed for multiple service lives).

Remanufacturing reduces environmental impact by 80–95% compared to manufacturing from raw materials, because the energy-intensive structural components (castings, machined housings, stamped frames) are reused.

9.4 Battery End of Life

Lithium batteries (LiPo, Li-ion, LFP) require special EOL handling because they contain flammable electrolytes, toxic electrode materials, and store significant electrical energy even when “dead” (a “depleted” LiPo still holds ~ 3.0 V).

Collection and transportation: discharged batteries are classified as hazardous materials for shipping. They must be individually packaged in non-conductive material, with terminals taped to prevent short circuits. Damaged or swollen batteries require fireproof containers for transport.

Recycling process: batteries are mechanically shredded in an inert atmosphere, then hydrometallurgically processed to recover lithium, cobalt, nickel, and copper. Recovery rates: cobalt > 95%, nickel > 90%, lithium $\sim 50\%$ (improving with new processes). The recovered metals re-enter the battery supply chain.

Second-life applications: EV batteries that no longer meet automotive performance requirements (below 80% capacity) often have sufficient capacity for stationary energy storage. Repurposing for grid storage extends the useful life by 5–10 years before final recycling.

Never Dispose of Lithium Batteries in Regular Trash

Lithium batteries in landfills are punctured by compactors, causing thermal runaway (fire and toxic gas release). They are the leading cause of recycling facility fires. All lithium batteries must be returned to designated collection points (Best Buy, Home Depot, university recycling centers) or shipped to licensed battery recyclers (e.g., Call2Recycle).

9.5 Electronics End of Life

Printed circuit boards contain: copper (traces and vias), tin and lead or tin-silver-copper (solder), gold (connector fingers and wire bond pads), fiberglass (substrate), epoxy (FR-4 resin), and various metals in components (iron, nickel, tantalum, rare earths). PCBs are classified as e-waste and must be processed by licensed e-waste recyclers.

The recycling process: (1) mechanical shredding, (2) magnetic separation (ferrous metals), (3) eddy-current separation (non-ferrous metals), (4) smelting to recover copper and precious metals, (5) controlled incineration of organic fraction (fiberglass, epoxy) with emissions treatment. Recovery rates for precious metals exceed 98%; base metals 90–95%.

Data security: if the system contains non-volatile memory (EEPROM, flash, SD card) with sensitive data (user information, calibration secrets, firmware IP), the memory must be securely erased or physically destroyed before recycling.

9.6 Mechanical Component End of Life

Metals: aluminum, steel, copper, and brass are infinitely recyclable without degradation. Aluminum recycling uses only 5% of the energy required for primary production. Ensure parts are free of non-metallic contaminants (adhesives, plastics, coatings) before recycling, or accept the lower-value “mixed metal” stream.

Plastics: thermoplastics (ABS, PET, PP, HDPE) can be melted and reprocessed, but each recycling cycle degrades the polymer (reduced strength, discoloration). Thermosets (epoxy, phenolic) and elastomers (silicone, neoprene) cannot be remelted and are typically landfilled or incinerated for energy recovery. 3D-printed PLA is industrially compostable but does not degrade in home compost or landfill conditions.

Fasteners: mixed fasteners (steel screws in aluminum parts) contaminate both material streams. During disassembly, sort fasteners by material. Stainless steel and zinc-plated steel are both recyclable but must be separated.

9.7 System Retirement Procedure

A structured retirement procedure ensures safety, data preservation, and responsible disposal:

1. **Decommission:** disconnect from power, drain fluids, discharge capacitors, remove batteries. Lock out/tag out per safety procedures.
2. **Data preservation:** download all logged data, calibration records, and maintenance history. Archive for the required retention period (7 years for quality records per ISO 9001).
3. **Data destruction:** securely erase or destroy non-volatile memory containing sensitive information.
4. **Hazardous material removal:** remove batteries, fluids (oil, coolant), mercury-containing components, and any materials requiring special handling. Package for hazardous waste disposal.
5. **Disassembly:** separate into material-homogeneous modules per the DfD procedure.
6. **Material disposition:** route each material stream to the appropriate recycler or disposal pathway. Document the disposition of each module.
7. **Record keeping:** file the retirement record documenting the date, responsible person, material dispositions, and any hazardous waste manifests.

9.8 Life Cycle Cost and Total Cost of Ownership

The total cost of ownership (TCO) includes all costs from acquisition through retirement:

$$\text{TCO} = C_{\text{acquisition}} + \sum_{t=1}^L \frac{C_{\text{operating}}(t) + C_{\text{maintenance}}(t)}{(1+r)^t} + \frac{C_{\text{disposal}}}{(1+r)^L} \quad (9.1)$$

where L is the service life (years), r is the discount rate, and the summation accounts for the time value of money. For many systems, the acquisition cost is only 20–30% of TCO; operating and maintenance costs dominate. Designing for low maintenance cost (reliable components, easy access to wear parts, self-diagnostic firmware) reduces TCO more than reducing the purchase price.

9.9 Life Cycle Assessment: A Simplified Approach for MEGN 301

A full Life Cycle Assessment (LCA) per ISO 14040 is a months-long effort. For MEGN 301, a simplified assessment captures the key insights:

Step 1: Material inventory. Weigh each component or estimate from CAD models. Categorize by material: aluminum, steel, ABS, PLA, copper (PCB + wire), lithium (battery), silicone, glass (LCD), etc.

Step 2: Embodied energy. Multiply each material mass by its embodied energy factor (MJ/kg): aluminum 170, steel 25, ABS 95, PLA 55, copper 60, lithium battery 400, glass 15, silicone 110. Sum to get total embodied energy.

Step 3: Use-phase energy. Power consumption $\times$ operating hours $\times$ expected lifetime. A 10 W device operating 8 hours/day for 5 years: $10 \times 8 \times 365 \times 5 = 146,000 \text{ Wh} = 526 \text{ MJ}$.

Step 4: Compare. If the total embodied energy (Step 2) is 50 MJ and the use-phase energy (Step 3) is 526 MJ, then 91% of the life-cycle energy is in the use phase. Design effort should focus on reducing power consumption, not reducing material weight. Conversely, if the product is passive (no power consumption), 100% of the impact is in materials, and material selection and recyclability dominate.

Step 5: End-of-life credit. Recycling recovers some embodied energy. Aluminum recycling saves 95% of primary production energy. Steel: 75%. Plastics: 50–80% (depending on contamination). Subtract the recycling credit from the total to get the net life-cycle energy.

9.10 Circular Economy Principles

The traditional linear economy follows “take-make-dispose”: extract raw materials, manufacture products, use them, and discard them. The **circular economy** (Figure 9.2) aims to keep materials in use indefinitely through:

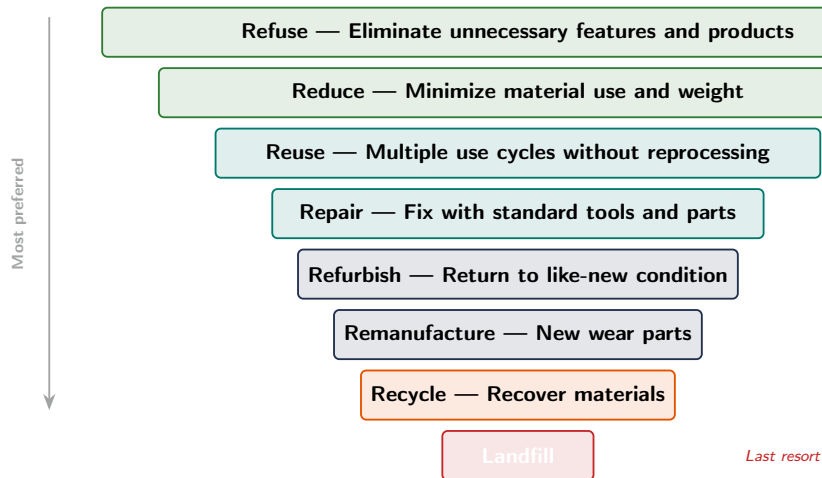


Figure 9.2: Circular economy hierarchy (most to least preferred). Design decisions should target the highest feasible tier. Recycling is better than landfill but is the lowest form of material recovery. Design for Repair and Design for Disassembly (Chapter 7, p. 66) enable the middle tiers.

Refuse: eliminate unnecessary products and features. Does the product really need a full-color LCD, or would a simple LED indicator suffice?

Reduce: minimize material use. Topology optimization, thin-wall design, and lightweighting reduce the mass (and embodied energy) of each product.

Reuse: design for multiple use cycles without reprocessing. Refillable containers, rechargeable batteries, and modular upgradeable electronics extend the useful life.

Repair: design for easy repair with standard tools and available spare parts. A product that lasts 10 years with one repair is more sustainable than one that lasts 3 years and is discarded.

Refurbish: return used products to like-new condition by replacing worn components and updating firmware/software.

Remanufacture: disassemble, clean, inspect, and reassemble with new wear parts. The structural components serve multiple product generations.

Recycle: recover materials for use in new products. This is the last resort in the circular hierarchy, not the first.

9.11 Regulatory Landscape

WEEE Directive (EU): producers are financially responsible for collecting and recycling end-of-life electrical and electronic equipment. Products must be marked with the crossed-out wheeler bin symbol. Consumers can return e-waste to retailers free of charge.

RoHS Directive (EU): restricts the use of lead, mercury, cadmium, hexavalent chromium, PBB, and PBDE in electrical equipment. Maximum concentrations: 0.1% by weight (0.01% for cadmium). Applies to products sold in the EU regardless of where they are manufactured.

REACH Regulation (EU): requires registration of chemical substances used in products. Substances of Very High Concern (SVHCs) must be disclosed to customers and may be restricted or banned.

California Proposition 65: requires warnings for products containing chemicals known to cause cancer or reproductive harm. Broadly applicable because products sold online may reach California consumers.

EPA regulations (US): governs disposal of hazardous waste including batteries, mercury-containing devices, and PCBs (polychlorinated biphenyls, found in old transformer oil and some capacitors). Improper disposal carries civil and criminal penalties.

9.12 Product Takeback and Extended Producer Responsibility

Under Extended Producer Responsibility (EPR) laws, the manufacturer bears the cost of collecting and recycling their products at end of life. This creates a direct financial incentive to design for recyclability: a product that costs \$5 to recycle is more profitable than one that costs \$50.

Design strategies to reduce takeback cost: (1) minimize material variety (fewer recycling streams), (2) eliminate hazardous materials (avoids special handling), (3) design for automated disassembly (reduces labor cost), (4) mark all materials clearly (enables automated sorting), and (5) make the product compact when disassembled (reduces shipping cost for returns).

9.13 Practice Questions

1. Your GreenShield prototype contains: an aluminum enclosure, an ABS 3D-printed housing, a PCB with ESP32, a 2000mAh LiPo battery, a DC motor, silicone tubing, and stainless steel fasteners. Write a disassembly procedure and identify the disposal pathway for each component.

2. Calculate the TCO for a system with acquisition cost \$500, annual operating cost \$50, annual maintenance cost \$100, disposal cost \$75, service life 5 years, and discount rate 5%. Compare to just the acquisition cost.
3. A competitor's product uses adhesive to bond the battery to the frame, making battery replacement impossible. Your product uses a screwed battery compartment. Compare the EOL implications. Which design is more sustainable, and why?
4. Research the RoHS Directive. List the six originally restricted substances and their maximum concentration limits. Which of these might be present in a typical MEGN 301 prototype?
5. Your team's motor has a rated life of 5000 hours. The system operates 8 hours/day, 250 days/year. When should the motor be replaced? What design feature would make this replacement easier?

9.13.1 Answers

1. Disassembly: (1) Remove 4 screws from enclosure lid (Phillips #2). (2) Disconnect battery JST connector; remove battery from Velcro mount. → Battery recycler (Call2Recycle). (3) Unscrew PCB (4× M3). Disconnect motor and tubing connectors. → PCB to e-waste recycler. (4) Remove motor (2× M3). → Motor to e-waste (contains copper and magnets). (5) Pull silicone tubing from barb fittings. → Landfill (silicone not recyclable). (6) Separate ABS housing (no fasteners to aluminum). → ABS: plastics recycler (#7). (7) Aluminum enclosure + SS screws. → Separate screws (steel) from enclosure (aluminum). Both to metal recycler. Time: ~4 minutes.
2. $TCO = 500 + \sum_{t=1}^5 (50 + 100) / (1.05)^t + 75 / (1.05)^5 = 500 + 150(4.329) + 75(0.784) = 500 + 649 + 59 = \$1,208$. The acquisition cost (\$500) is only 41% of TCO. Operating and maintenance (\$649) dominate.
3. Adhesive-bonded battery: battery cannot be replaced when degraded (entire product discarded, ~2 years). Battery cannot be separated for recycling (contaminated with adhesive). Product life limited by battery life. Screwed compartment: battery replaced in 2 minutes, extending product life by 2–3 battery generations (~6–8 years). Battery cleanly removable for recycling. The screwed design is more sustainable by a factor of 3–4× in product lifetime and eliminates the adhesive contamination problem.
4. RoHS restricts: lead (<0.1%), mercury (<0.1%), cadmium (<0.01%), hexavalent chromium (<0.1%), PBB (<0.1%), PBDE (<0.1%). In a typical MEGN 301 prototype: lead may be present in hand-soldered joints if leaded solder (Sn63/Pb37) is used instead of lead-free. Hexavalent chromium may be present in some zinc-plated fasteners. Solution: use SAC305 lead-free solder and verify fastener plating is RoHS-compliant.
5. Service life: $5000 / (8 \times 250) = 2.5$ years. Plan motor replacement at 2 years (before failure, with safety margin). Design for easy replacement: mount the motor with a quick-release bracket (two screws), use a detachable electrical connector (not hardwired), and include the motor part number in the maintenance manual.

Part II

Technical Reference

Chapter 10

Electrical & Fluid Connectors

When to Read This Chapter

Sprint 3 (CDR): Read when selecting connectors for your BOM. Focus on the connector family quick-reference table and the selection framework. **Sprint 4–5:** Return to the failure modes section (fretting corrosion, contact resistance) when debugging intermittent connections. **Fluid projects:** Read the NPT vs. BSP section before ordering any fittings.

When to Consult This Chapter

Sprint 3 (CDR): When specifying connectors in your BOM and wiring diagrams. Use the quick-reference table at the end for common families. **Sprint 4–5:** When debugging intermittent connections during integration.

10.1 Electrical Connectors

10.1.1 Connector Anatomy

Every electrical connector consists of three fundamental elements:

1. **Contacts.** The conductive pins (male) and sockets (female) that carry electrical current. Typical base metals are phosphor bronze and beryllium copper (BeCu), plated with either gold (best corrosion resistance, lowest contact resistance) or tin (lower cost, adequate for benign environments).
2. **Housing.** The insulating body that mechanically positions and electrically isolates the contacts. Common housing materials include:
 - Nylon (PA66); general purpose, good toughness, low cost.
 - PBT (polybutylene terephthalate); higher stiffness, better chemical resistance.
 - LCP (liquid crystal polymer); highest temperature rating, used in reflow-soldered connectors.
3. **Termination.** The mechanical and electrical joint between the wire conductor and the contact. The five major termination methods are detailed in Section 10.1.6.

10.1.2 Gender Convention

- **Male** (plug/header): pins protrude from the connector face.

- **Female** (receptacle/socket): pins are received into sockets.
- In wire-to-wire assemblies, one cable carries the plug, the other the receptacle.
- In wire-to-board applications, the board-mounted header is typically male.

10.1.3 Keying and Polarization

Connectors use physical features to ensure correct mating orientation and prevent damage from incorrect insertion. Common keying methods include key slots, asymmetric housing profiles, color coding, and differentiated contact arrangements.

Design Rule

Always specify keyed connectors when your design has multiple similar-looking connections. Incorrect mating can destroy components, and in student projects this is one of the most common causes of electrical failure on first power-up.

10.1.4 Connection Types

Electrical connectors are classified by *what* they connect:

Type	Description	Typical Examples
Wire-to-Wire	Joins two discrete wires or cable harnesses. Both sides are free-hanging connectors.	Molex Mini-Fit Jr., JST SM, Deutsch DT, automotive weatherpacks
Wire-to-Board	Connects a wire harness to a printed circuit board (PCB). One side is a cable connector; the other is soldered to the board.	JST XH/PH, Molex KK 254, pin headers with crimp housings
Board-to-Board	Connects two PCBs directly, often in stacked or mezzanine configurations. No wires involved.	Board-to-board headers, mezzanine connectors, card-edge connectors

Table 10.1: Electrical connector classification by connection type.

10.1.5 Harsh-Environment Connectors

Students designing vehicle harnesses, outdoor systems, or launch hardware will encounter connectors engineered for abuse. Understanding the physical components of these systems is essential for specifying them correctly.

Deutsch DT / DTM Series (automotive/off-highway): These use a wedgelock that mechanically retains all contacts simultaneously, independent of the wire seal. Each cavity receives a silicone wire seal that grips the insulation OD, providing IP67/IP69K protection. Contacts are stamped-and-formed with a barrel crimp; the crimp must capture both the conductor and the insulation for strain relief. Deutsch connectors are the de facto standard for SAE Baja, Formula SAE, and agricultural equipment harnesses.

MIL-DTL-38999 Series III (aerospace/defense): A bayonet-coupled circular connector with a self-locking coupling ring that resists vibration. Uses machined contacts (not stamped) with

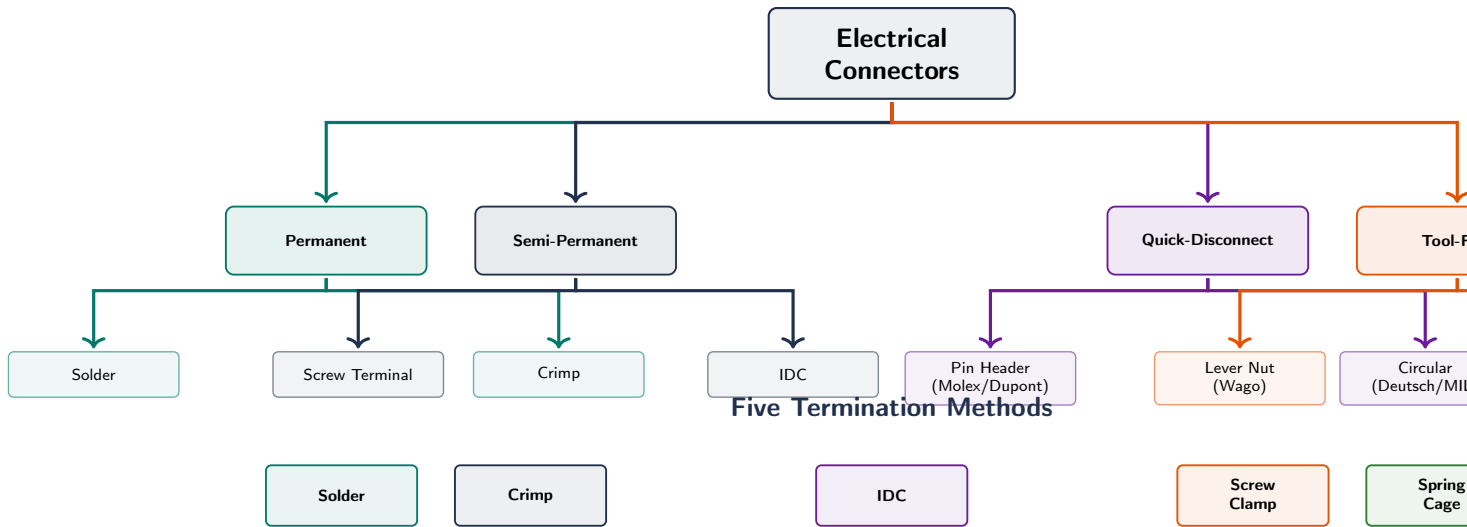


Figure 10.1: Electrical connector taxonomy by connection type with five termination methods.

interference-fit rear grommets for environmental sealing. Rated for extreme vibration (per MIL-STD-810), wide temperature ranges (-65 to $+200^{\circ}\text{C}$), and EMI shielding via conductive shell-to-shell contact. If your design involves launch hardware, flight systems, or classified defense work, 38999 is the starting point.

Common to both: Proper barrel crimps are critical; the crimp must form a gas-tight cold weld in the conductor barrel *and* a secure grip in the insulation barrel. Using generic pliers instead of a calibrated ratchet tool produces crimps that look acceptable but fail under vibration or thermal cycling.

10.1.6 Termination Methods

The termination method is how the wire conductor is mechanically and electrically bonded to the connector contact. Each method involves different tools, skill levels, and performance characteristics.

Crimp

A precision die compresses a metal barrel around a stripped wire conductor, creating a gas-tight joint through cold-welding at the metal interface.

- **Strengths:** Fast, repeatable, no heat required, excellent vibration resistance, gas-tight joint resists corrosion.
- **Limitations:** Requires calibrated crimp tools (hand ratchet or pneumatic); die must match the contact exactly. Poor crimps are invisible failures.
- **Best for:** Production wiring, automotive harnesses, harsh-environment connectors.
- **Quality verification:** Pull-test a sample of crimps per IPC/WHMA-A-620. Cross-section crimps periodically to verify cold-weld zone.

Pin Extraction: Use the Right Tool

In prototype environments, students inevitably insert a contact into the wrong cavity. **Never** yank the wire or use a paperclip to extract a crimped pin; this damages the housing retention lance and the contact, creating an unreliable connection that will fail in service. Every connector family has a specific extraction tool (typically a thin-walled sleeve that depresses the retention lance). Keep the correct extraction tools on hand alongside your crimp tools; they cost a few dollars and save hours of rework and scrapped housings.

Solder

Molten solder alloy (typically SAC305 lead-free or Sn63/Pb37) forms a metallurgical bond between wire and contact.

- **Strengths:** Low contact resistance, well-understood process, easy visual inspection.
- **Limitations:** Requires heat (risk of insulation damage), not reworkable without desoldering tools, solder joints are brittle under vibration and thermal cycling.
- **Best for:** Board-level connections (through-hole and SMT), prototype wiring, low-vibration environments.

Insulation Displacement Contact (IDC)

A sharp, slotted contact blade pierces through wire insulation and makes contact with the conductor, all in a single operation.

- **Strengths:** Extremely fast, no stripping required, good for mass-termination of ribbon cables.
- **Limitations:** Limited to small wire gauges (typically 22–28 AWG), not suitable for high-vibration environments, contact can relax over time.
- **Best for:** Ribbon cable connectors, low-current signal wiring, internal PCB connections.

Screw Terminal

A screw clamp compresses the wire conductor against a metal plate or barrel inside the terminal block.

- **Strengths:** Tool-free or simple screwdriver, field-serviceable, wide wire gauge range, easy to inspect.
- **Limitations:** Bulky, screws can loosen under vibration, relatively high contact resistance compared to crimp.
- **Best for:** Panel-mount terminal blocks, DIN-rail wiring, power distribution, industrial controls.

Press-Fit (Compliant Pin)

A specially shaped contact pin (often an “eye-of-needle” or “action pin” geometry) is pressed into a plated through-hole on a PCB, creating a gas-tight interference fit.

- **Strengths:** No solder required, excellent reliability, eliminates solder joint fatigue, good for backplane connectors.
- **Limitations:** Requires controlled insertion force and tooling, board must have proper hole tolerances, not field-reworkable.
- **Best for:** Backplane connectors, automotive ECU headers, high-reliability board-level connections.

Termination Method Selection Summary

Method	Mechanism	Strengths	Limitations	Best For
Solder	Molten alloy bonds wire to pad/pin	Lowest resistance Permanent Compact	Requires skill Not field-serviceable Heat-sensitive parts	PCB assembly Permanent wiring High-reliability
Crimp	Cold weld via controlled deformation	Gas-tight joint Repeatable Fast	Requires tooling Wire-size specific One-time use	Harnesses Automotive Aerospace
IDC	Blade pierces insulation to contact wire	No stripping Mass termination Low cost	Limited current Vibration-sensitive Small wire only	Ribbon cables Internal PCB connections
Screw Clamp	Screw presses wire against plate	Re-usable Field-serviceable No special tools	Loosens with vibration Bulky	Panel wiring Industrial Prototyping
Spring Cage	Spring clamp grips wire automatically	Vibration-proof Tool-free Fast	Higher cost Limited wire range	Industrial automation Wago 221 series

Figure 10.2: Termination tradeoff matrix; mechanism, strengths, limitations, and best applications.

10.1.7 Key Electrical Specifications

When specifying an electrical connector in your design documentation, you must define each of the following parameters.

Current Rating

The maximum continuous current per contact, in amperes. This depends on contact size, material, plating type, and number of loaded contacts.

Thermal Derating

Always derate for multi-pin loaded connectors. When adjacent contacts all carry current, thermal stacking raises the temperature inside the housing, reducing each contact's effective capacity. A connector rated at 10 A per contact at 25 °C may only safely handle 7 A per contact at 85 °C ambient. Manufacturers provide derating curves; never ignore them.

Bridging to Wire Selection (AWG)

The connector's current rating sets the *minimum* wire gauge you can use. Standard practice: select wire whose ampacity (per NEC Table 310.16 or MIL-W-22759 for aerospace) meets or exceeds the derated connector contact rating. Common reference points for chassis wiring in free air at 30 °C: 22 AWG $\approx$ 7 A, 18 AWG $\approx$ 10 A, 16 AWG $\approx$ 13 A, 14 AWG $\approx$ 17 A, 12 AWG

≈ 23 A, 10 AWG ≈ 33 A. Always verify that the connector contact accepts your chosen wire gauge; crimp contacts are rated for a specific AWG range (e.g., 18–22 AWG), and using the wrong gauge produces unreliable crimps.

Voltage Rating

The maximum working voltage (VAC or VDC). This is determined by creepage (surface distance) and clearance (air gap) distances between adjacent contacts, as specified by IEC 61984.

Important: Always specify the *rated working voltage*, not the dielectric withstand (breakdown) voltage. Working voltage includes safety margin; breakdown voltage does not.

Contact Resistance

The electrical resistance measured across a mated contact pair, typically in milliohms ($m\Omega$). Gold-plated contacts achieve $<20 m\Omega$; tin-plated contacts may range from 20–50 $m\Omega$ when new.

Why it matters: At 10 A, even 50 $m\Omega$ of contact resistance produces:

$$P = I^2 R = (10)^2 \times 0.050 = 0.5 \text{ W of heat per contact} \quad (10.1)$$

Contact resistance increases with mating cycles and in corrosive environments. Always verify with a milliohm meter during commissioning.

IP Rating (Ingress Protection)

Defined by IEC 60529, the IP code uses two digits:

- **First digit (0–6):** Protection against solid objects, from “no protection” (0) to “dust-tight” (6).
- **Second digit (0–9K):** Protection against liquids, from “no protection” (0) to “high-pressure, high-temperature washdown” (9K).

IP Rating	Meaning	Typical Application
IP20	Finger-safe; no liquid protection	Indoor panel wiring
IP54	Dust-protected; splash-proof	General industrial
IP67	Dust-tight; 1 m immersion for 30 min	Outdoor, vehicle, marine
IP68	Dust-tight; continuous submersion	Submersible equipment
IP69K	Dust-tight; high-pressure washdown	Food/pharma equipment

Table 10.2: Common IP ratings and their applications.

Mating Cycles

The rated number of connect/disconnect operations before mechanical or electrical degradation. Ranges vary enormously:

- 25–50 cycles: permanent/semi-permanent installations.
- 500–1,000 cycles: periodic maintenance access.
- 5,000–10,000+ cycles: test fixtures, instrumentation.

Gold plating significantly outlasts tin plating. Contact wear directly increases resistance over mating life.

10.1.8 Electrical Connector Failure Modes

Understanding failure modes is essential for designing robust connections. These are listed roughly in order of how commonly they cause problems in practice.

Fretting Corrosion

Mechanism: Micro-motion from vibration (amplitudes as small as $10\mu\text{m}$) wears through the contact plating, exposing base metal. Oxides form on the exposed metal, creating high-resistance debris particles that accumulate at the contact interface.

Why it is dangerous: Fretting corrosion is the most insidious failure mode because it develops slowly, manifests as intermittent contact, and is difficult to diagnose. Resistance may fluctuate between normal and open-circuit depending on vibration state.

Mitigations:

- Gold plating (resists oxide formation).
- Higher contact normal force (resists micro-motion).
- Strain-relief on cables (reduces transmitted vibration).
- Secondary locking features on connectors.

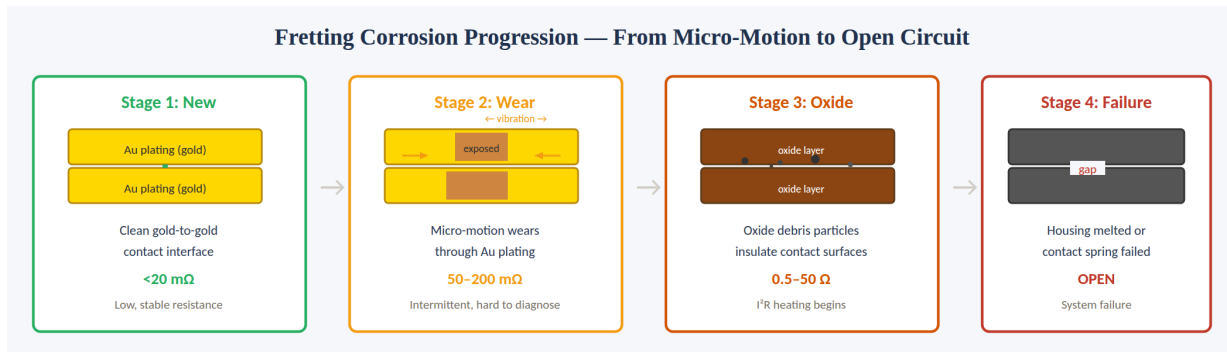


Figure 10.3: Fretting corrosion progression, from clean gold-plated contacts through oxide accumulation to open-circuit failure. Note resistance escalation across stages.

Figure 10.3 illustrates the four-stage progression of fretting corrosion. At Stage 1, clean gold-plated contacts exhibit stable resistance below $20\text{ m}\Omega$. Vibration-induced micro-motion gradually wears through the plating at Stage 2, intermittently exposing base metal and raising resistance to $50\text{--}200\text{ m}\Omega$, often misdiagnosed as a “loose connection.” By Stage 3, accumulated oxide debris insulates the contact surfaces, resistance climbs to the ohm range, and I^2R heating accelerates further degradation. Stage 4 is outright failure: melted housing, collapsed contact springs, or open circuit. Because Stage 2 symptoms are intermittent and position-dependent, fretting often passes bench-top continuity checks only to fail under dynamic load, a pattern students must learn to recognize.

Overheating (I^2R Heating)

Mechanism: Exceeding the current rating causes resistive heating at the contact interface. Heat accumulates in the housing, softening thermoplastic materials and potentially igniting adjacent insulation or components.

Mitigations:

- Never exceed rated current; derate for ambient temperature.
- Select contacts with lower resistance (gold plating, larger cross-section).
- Distribute high currents across multiple contacts.

Creep and Stress Relaxation

Mechanism: Contact springs gradually lose their clamping force at elevated temperatures. The contact normal force decreases over time, increasing resistance and accelerating other failure modes.

Mitigations:

- Specify beryllium copper (BeCu) contacts for any application above 60 °C continuous. BeCu resists creep far better than brass or phosphor bronze.
- Overspecify contact normal force for high-temperature applications.

Moisture Ingress

Mechanism: Water bridges between contacts, causing electrochemical corrosion, increased resistance, or direct short circuits.

Mitigations:

- Specify appropriate IP rating per IEC 60529.
- Use sealed connectors with O-ring face seals and wire seals for any outdoor, vehicle, or wash-down application.
- Apply conformal coating on exposed PCB connections.

10.2 Fluid System Connectors

10.2.1 Threaded Pipe Fitting Standards

Two incompatible tapered thread standards dominate global pipe fitting. Geography determines which you encounter.

NPT: National Pipe Taper (ASME B1.20.1)

- **Thread angle:** 60° with pointed crests and roots.
- **Taper:** 1:16 ($\frac{1}{4}$ inch per inch of thread length, equivalent to $\frac{3}{4}$ inch per foot).
- **Sealing mechanism:** Thread interference creates a metal-to-metal wedge seal. Because the thread form has clearance at crests and roots, NPT joints *always* require supplemental sealing; either PTFE (Teflon) tape or anaerobic pipe dope.
- **NPTF variant:** “Dryseal” NPT uses a modified thread form with tighter root/crest tolerances that crushes metal at the interface, theoretically sealing without sealant. In practice, sealant is still recommended.
- **Size range:** 1/8 in. to 24 in. nominal pipe size (NPS).
- **Pressure rating:** Up to 15000 psi depending on material and size.
- **Geography:** Dominant in the United States and Canada.

BSP: British Standard Pipe (ISO 228 / ISO 7-1)

- **Thread angle:** 55° Whitworth form with rounded crests and roots.
- **Two sub-types:**
 - **BSPT** (designation “R”): Tapered threads, seals like NPT via thread interference plus sealant.
 - **BSPP** (designation “G”): Parallel (straight) threads that *do not* seal on the threads. Sealing is achieved with a bonded washer or O-ring compressed against a flat face.
- **Size range:** 1/8 in. to 6 in. (inch-denominated but used globally outside North America).
- **Pressure rating:** Up to 6000 psi typical.
- **Geography:** Dominant in Europe, Asia, and most of the world outside North America.

The Point of No Return: Thread Galling

NPT threads are tapered at 60°. BSP threads are tapered at 55°. A 55° BSP fitting *will* start threading into a 60° NPT port, and then seize. In stainless steel, this produces **galling**: a cold-weld that permanently fuses the two components. Both parts become scrap metal. There is no recovery.

The Finger Test: before applying any wrench, thread the fitting by hand. If you cannot achieve at least **three full turns by finger pressure alone**, STOP. The threads are incompatible, cross-threaded, or contaminated. Any attempt to force the connection with a wrench will result in irreversible damage. Verify the thread standard (NPT vs. BSP), clean both threads, apply appropriate sealant (Teflon tape for NPT, bonded washer for BSP parallel), and try again.

Thread Galling: The Point of No Return

In stainless steel or aluminum, forcing a mismatched or cross-threaded fitting with a wrench causes **galling**; a cold-weld that permanently destroys both components. Once galling begins, reaching for a bigger wrench turns a \$50 component into a \$100 mistake (two destroyed parts plus the replacement).

The Finger Test: if you cannot thread a fitting at least three full turns by hand, **STOP**. You are either cross-threaded or mixing thread standards. No amount of force or sealant can fix galled threads.

Prevention for stainless-on-stainless connections: apply nickel-based anti-seize compound (e.g., Permatex Nickel Anti-Seize or Loctite LB 8023) to the male threads before assembly. Anti-seize reduces friction and prevents the metal-to-metal adhesion that initiates galling. This is mandatory for any stainless steel fluid fitting in your design.

Critical Incompatibility

NPT (60° pointed threads) and BSP (55° rounded threads) are **NOT** interchangeable. They may appear to engage for a few turns but will not seal properly. Cross-threading causes galling, thread damage, and leaks that may develop catastrophically under pressure. **Always** verify thread standard with a thread pitch gauge before assembly.

Feature	NPT (ASME B1.20.1)	BSP (ISO 228 / ISO 7-1)
Thread angle	60°, pointed crests/roots	55° Whitworth, rounded crests/roots
Taper	1:16 (all NPT tapered)	BSPT tapered; BSPP parallel
Sealing	Thread interference + sealant	BSPT: thread + sealant; BSPP: washer/O-ring
Geography	U.S., Canada	Europe, Asia, rest of world
Max pressure	15000 psi	6000 psi typical

Table 10.3: NPT vs. BSP thread comparison.

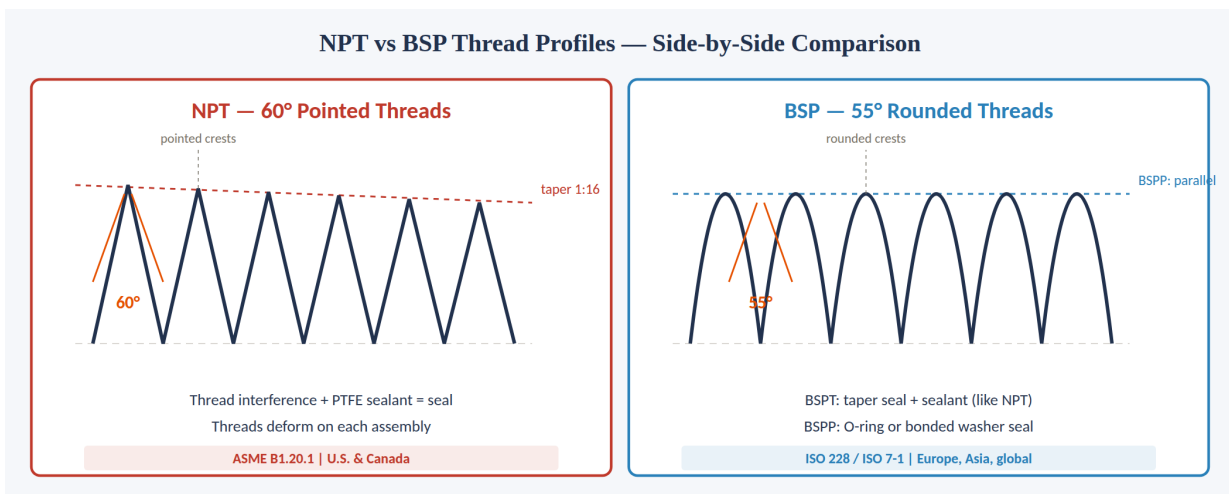


Figure 10.4: NPT vs. BSP thread profiles; note the pointed 60° crests on NPT versus the rounded 55° Whitworth crests on BSP. These subtle geometric differences make the two standards completely incompatible.

Figure 10.4 places the two thread forms side by side so the geometric distinction is immediately visible. NPT's 60° pointed crests create a wedging action as the taper engages; metal deforms at each assembly, which is why PTFE sealant is required and why each reassembly cycle slightly degrades the thread. BSP's 55° Whitworth form uses rounded crests that are less prone to galling but follow an entirely different pitch schedule. The BSPP (parallel) variant does not seal on the thread at all; the thread merely provides clamping force while an O-ring or bonded washer creates the pressure boundary against a flat face. This is a critical distinction: applying PTFE tape to a BSPP fitting is a common student mistake that serves no purpose and can actually prevent proper washer compression.

10.2.2 Tube Fittings and Quick-Connect Couplings

Unlike threaded pipe fittings (which connect pipes), tube fittings connect *tubing*; a product specified by outside diameter (OD) and wall thickness, with tighter dimensional tolerances than pipe.

JIC 37° Flare (SAE J514)

The tube end is flared to a 37° cone that mates with a matching seat in the fitting body. The joint seals metal-to-metal; no sealant, tape, or elastomeric seal is required. The fitting uses a straight (non-tapered) thread; the thread provides clamping force only and does not contribute to sealing.

- Reassemblable without degradation (unlike taper threads which deform with each assembly).
- The hydraulic industry standard worldwide.
- Pressure ratings to 10000 psi.
- AN (Army–Navy) fittings are the military equivalent of JIC, with identical 37° geometry.

Compression Fittings (Swagelok-type)

One or two ferrules (metal rings) are swaged (compressed) onto the tube OD when the nut is tightened. The ferrule bites into the tube, creating both a grip and a seal.

- No welding, brazing, flaring, or special tools required.
- Tube OD is the sizing basis (1/4 in., 3/8 in., 1/2 in., etc.).
- **Gaugeable:** Many designs include a visual indicator (gap gauge) so you can verify proper assembly by inspection.
- Pressure ratings up to 20000 psi for medium-pressure configurations.
- Brands: Swagelok, Parker CPI/A-LOK, Ham-Let.

Push-to-Connect

The tube pushes into the fitting body. An internal collet (toothed ring) grips the tube OD, and an O-ring provides the seal. Assembly is one-handed and tool-free.

- Common in pneumatics (air systems), typically up to ~250 psi.
- Common brands: SMC, Festo, John Guest.
- Fast assembly but limited to lower pressures and non-critical sealing applications.

Quick-Disconnect (QD) Couplings

Two halves, a plug and a socket, connect and disconnect by hand, often with a locking sleeve or collar. Self-sealing valves in one or both halves prevent fluid loss on disconnect.

- **Flat-face:** Minimal fluid spillage on disconnect, used in clean or environmentally sensitive applications.
- **Poppet-valve:** Spring-loaded valve seals on disconnect, less expensive but more spillage.
- Common applications: hydraulic tool circuits, coolant loops, test equipment.

Fitting Family	Pressure	Seal Type	Key Characteristic
JIC 37° Flare	10000 psi	Metal-to-metal	Reassemblable; hydraulic standard
Compression	20000 psi	Ferrule swage	No special tools; gaugeable
Push-to-Connect	250 psi	O-ring	Tool-free; pneumatics
Quick-Disconnect	Varies	Valve + O-ring	Hand-connect; self-sealing

Table 10.4: Tube fitting and coupling comparison.

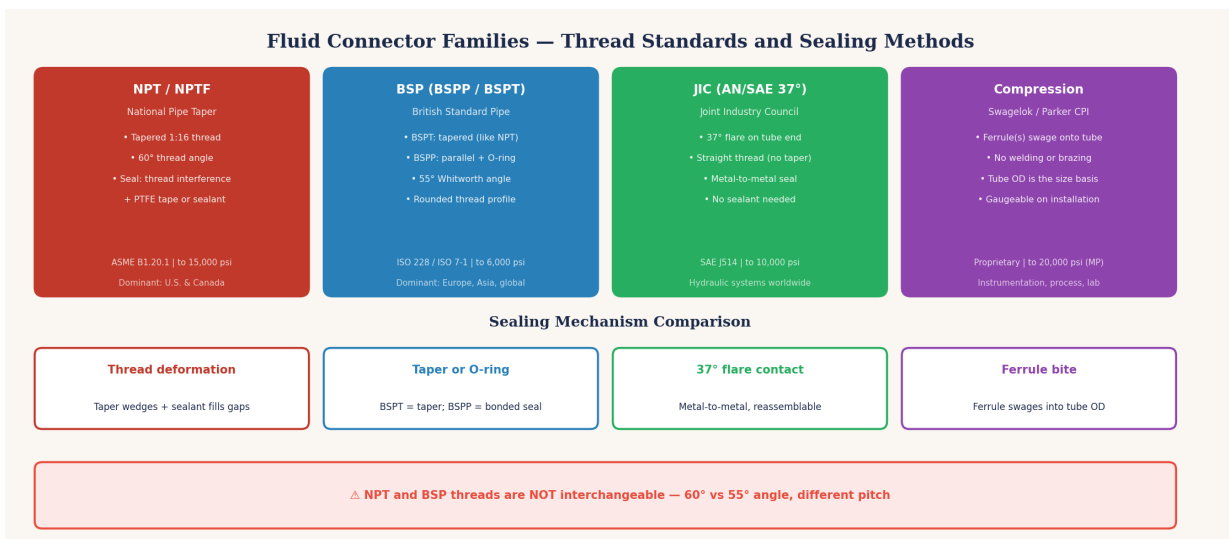


Figure 10.5: Four major fluid connector families with thread standards, pressure ratings, and sealing mechanisms.

Figure 10.6 shows the internal geometry of each sealing method so students can visualize *why* proper torque and assembly procedures matter. In the NPT taper seal, the thread flanks themselves deform to create a pressure boundary, which is why sealant is mandatory to fill the remaining micro-gaps and why repeated assembly progressively damages the seal surfaces. In the JIC 37° flare, the cone-shaped tube end presses against a matching seat in the fitting body; the straight thread provides clamping force only. This metal-to-metal geometry is fully reassemblable because the sealing surfaces do not deform. In compression fittings, the front ferrule bites radially into the tube OD while the back ferrule provides axial grip; the nut drives both ferrules during initial makeup. Once swaged, the ferrule is permanently deformed, which limits reassembly. In the BSPP face seal, the parallel thread clamps the fitting against the port face while an O-ring in a groove creates the pressure boundary, the thread contributes zero sealing.

10.2.3 Key Fluid Connector Specifications

When specifying a fluid connector in your design documentation, define each of the following.

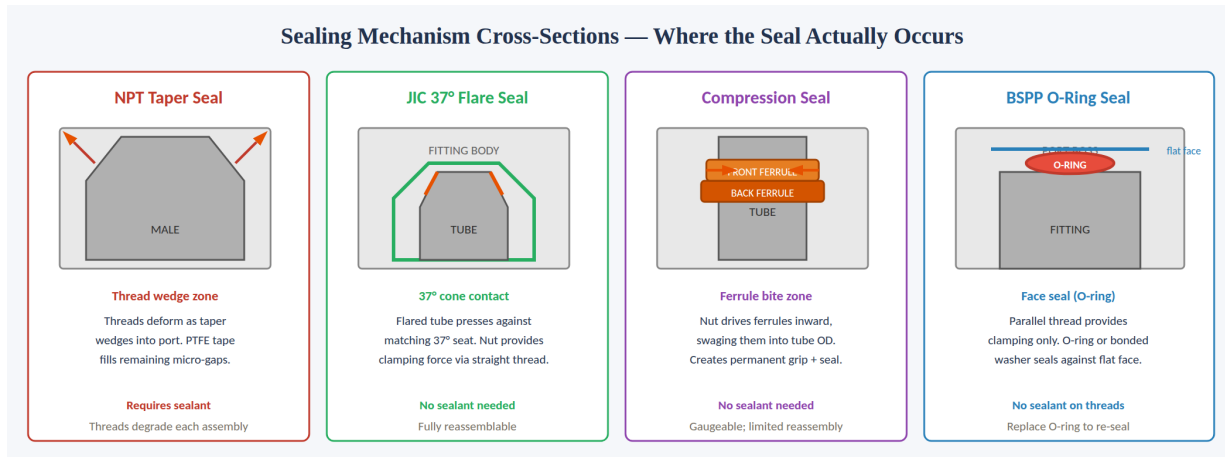


Figure 10.6: Sealing mechanism cross-sections showing exactly where each seal forms: NPT thread wedge, JIC 37° flare cone contact, compression ferrule bite zone, and BSPP face-seal O-ring.

Working Pressure

The maximum continuous operating pressure (psi or bar). Always specify *working* pressure, not burst pressure. Standard safety factors:

- Hydraulic: burst $\geq 4\times$ working.
- Pneumatic: burst $\geq 3\times$ working.
- **Design rule:** Never operate above 25% of burst pressure.

Temperature Range

Temperature determines seal material selection:

Seal Material	Range	Notes
Buna-N (Nitrile, NBR)	−40 to 250 °F	General purpose; petroleum-compatible
Viton (FKM)	−20 to 400 °F	Chemical resistance; fuel systems
PTFE (Teflon)	−100 to 500 °F	Widest range; chemically inert; no elasticity
EPDM	−65 to 300 °F	Steam/water service; not oil-compatible

Table 10.5: Seal material temperature ranges and applications.

Always verify seal compatibility with both the fluid *and* the operating temperature.

Fluid Compatibility

Verify fitting body material and seal material against the working fluid using manufacturer chemical compatibility charts:

- **Stainless steel:** Resists most chemicals; highest cost.
- **Brass:** Common for water, air, and general service; not for ammonia or acetylene.
- **Carbon steel:** Standard for hydraulic oil systems; requires corrosion protection.

Thread / Tube Size

- **Threaded fittings:** Specified by Nominal Pipe Size (NPS). Note that NPS does *not* equal the actual outside diameter. For example, 1/2 in. NPS pipe has an OD of 0.840 in.
- **Tube fittings:** Specified by tube outside diameter (OD). Common sizes: 1/4 in., 3/8 in., 1/2 in.
- **Always measure with calipers and thread gauges.** Never assume a size by visual inspection.

End Configuration

Select the geometry that connects the fitting to your system: straight, elbow (45° or 90°), tee, cross, union, reducer, or bulkhead.

Minimize Joints

Every fitting joint is a potential leak point. Design your routing to minimize the total number of fittings in a line. Avoid stacking street elbows in series; use a single 90° elbow or a pre-bent tube run instead.

10.2.4 Fluid Connector Failure Modes

Cross-Threading

Mechanism: Attempting to mate NPT into BSP (or vice versa), or even same-standard threads started at an angle, destroys thread profiles and causes immediate or delayed leaks.

Mitigations:

- Always verify thread standard with a thread pitch gauge before assembly.
- Start threads by hand for at least two full turns before applying a wrench.
- NPT: 60° pointed threads. BSP: 55° rounded threads. They *feel* similar but are not compatible.

Over-Tightening

Mechanism: Excessive torque cracks port bosses (especially aluminum and cast iron), deforms compression ferrules beyond recovery, and splits flare cones. Over-tightening destroys more fittings than under-tightening.

Mitigations:

- Always use manufacturer-specified torque values.
- Use a calibrated torque wrench, not a “feel” wrench.
- For Swagelok-type fittings, follow the turns-past-finger-tight specification exactly.

Seal Degradation

Mechanism: O-rings and gaskets deteriorate from chemical attack (incompatible fluid), UV exposure, thermal cycling, and simple age. Seals swell, harden, crack, or take a permanent compression set.

Mitigations:

- Establish replacement intervals based on seal material and service conditions.
- Keep spare seal kits on hand.
- Verify seal material compatibility with the fluid and temperature range before installation.

Vibration Fatigue

Mechanism: Cyclic stress at tube-to-fitting junctions nucleates fatigue cracks. Under pressure, these cracks propagate to failure; potentially a sudden, high-pressure fluid release.

Mitigations:

- Use proper tube support clamps spaced per manufacturer recommendations.
- Route tubing to avoid sharp bends and unsupported cantilever lengths.
- Include flexible hose segments at vibration sources (pumps, engines).

High-Pressure Fluid Injection Injury

A hydraulic line leak may be invisible; a pinhole stream at 2000 psi can penetrate skin and inject fluid deep into tissue, causing catastrophic damage that requires emergency surgery within hours. **Never** check for hydraulic leaks by running a bare hand along a pressurized line. Use a piece of cardboard or a thermal camera instead. This is one of the most serious and least-understood hazards in hydraulic system work, and students designing or testing any pressurized fluid system must be explicitly aware of it before commissioning.

10.3 Selection and Design Practice

10.3.1 Connector Selection Logic

Whether electrical or fluid, connector selection follows a universal logic sequence:

1. **Rating:** What current, voltage, pressure, or flow must it handle?
2. **Environment:** Temperature range, vibration, moisture/IP, chemical exposure?
3. **Serviceability:** How often must it be connected/disconnected? What tools are available?
4. **Cost:** What is the budget per connection, including tooling?
5. **Standards:** Which industry/regulatory standards apply (IEC, SAE, ASME, MIL)?
6. **Space:** What are the physical envelope constraints?

Spec the Connector First

The connector determines wire gauge, tube OD, termination tooling, and housing envelope. Select the connector family first (based on rating, environment, and serviceability), then design the wiring or tubing around it, not the other way around.

10.3.2 Design Rules of Thumb

1. **Derate everything.** Never operate at 100% of rated current, pressure, or temperature. A 25% safety margin is minimum engineering practice; 50% is professional practice for critical systems.
2. **Design for Assembly (DFA).** Provide wrench clearance around fittings. Group fluid connections for leak-test access. Use consistent connector families within a subsystem to reduce tool count and spare parts inventory.

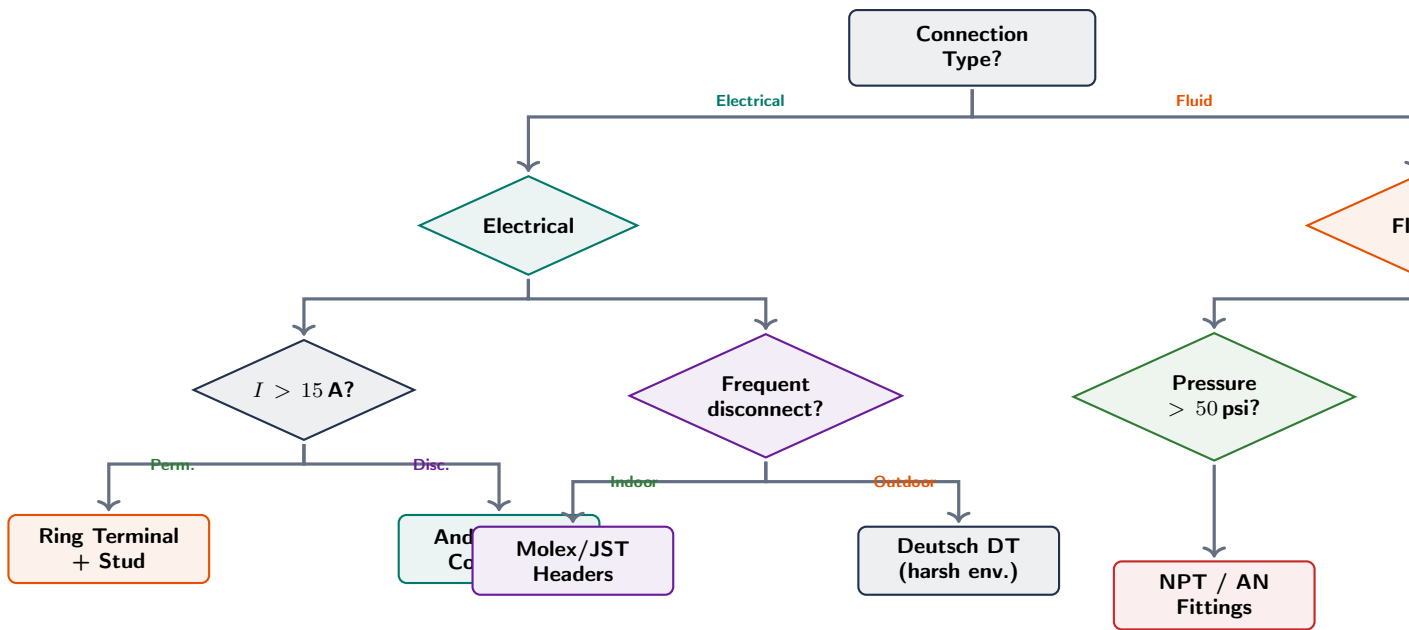


Figure 10.7: Decision tree for electrical and fluid connector selection.

3. **For fluid systems:** Verify thread standard with a gauge before assembly. Use manufacturer torque specifications. Pressure test every joint before commissioning; hydrostatic test at $1.5 \times$ working pressure per ASME B31.3.
4. **For electrical systems:** Pull-test a sample of crimps per IPC/WHMA-A-620. Verify contact resistance with a milliohm meter. Document every connector in your wire harness schedule.
5. **Document every choice.** Record part number, manufacturer, specifications, and selection rationale in your Bill of Materials (BOM). Your future self and your teammates will need this information for procurement, assembly, and troubleshooting.

10.3.3 Key Standards Reference

10.4 Applying Connectors in Your Design Projects

Incorrect connector selection is one of the most common failure modes in student design projects. The following workflow ensures systematic coverage.

1. **Create a connection schedule early.** List every electrical and fluid joint in your design. Classify each by type (wire-to-wire, wire-to-board, NPT, JIC, etc.). This becomes a living document through your design reviews.
2. **For every electrical connection, specify:** connector family, pin count, current rating (derated), voltage rating, IP rating, termination method, wire gauge, and mating cycles.
3. **For every fluid connection, specify:** fitting standard (NPT, BSP, JIC, or compression), nominal size, working pressure, seal material, fluid compatibility, and assembly torque.

Standard	Org	Scope
IEC 61984 [2]	IEC	Electrical connector safety requirements, testing, and performance criteria
IEC 60529 [3]	IEC	IP ingress protection rating system (IP67, IP68, IP69K)
MIL-DTL-38999	U.S. DoD	Military circular connectors for high-performance, harsh-environment applications
UL 486A-486B	UL	Safety certification for wire connectors: crimp, solder, and IDC terminations
SAE J514	SAE	JIC 37° flare hydraulic tube fittings; dimensions and performance requirements
ASME B1.20.1	ASME	NPT tapered pipe thread form, dimensions, and tolerances
ISO 228 / ISO 7-1	ISO	BSP parallel (BSPP) and tapered (BSPT) pipe thread standards
ASME B31.3	ASME	Process piping code; materials, fabrication, inspection, and testing

Table 10.6: Key connector and fluid-fitting standards.

4. **Build a complete BOM** with manufacturer part numbers and procurement sources. Recommended suppliers:
 - Electrical: Digi-Key, Mouser, Newark, Allied.
 - Fluid: Swagelok, Parker, McMaster-Carr, Grainger.
5. **Design review milestones:**
 - **PDR:** Present your connection schedule.
 - **CDR:** Present datasheets and torque specifications.
 - **FDR:** Demonstrate assembly procedures and test results.
6. **Test every connection before commissioning:** Pull-test electrical crimps, measure contact resistance with a milliohm meter, and pressure-test all fluid joints at $1.5\times$ working pressure.

10.5 Quick-Reference Summary

Five Key Takeaways

1. Electrical connectors are classified by *what they connect* (wire-to-wire, wire-to-board, board-to-board) and *how they terminate* (crimp, solder, IDC, screw, press-fit).
2. For every electrical connection, specify: current rating, voltage rating, IP rating, contact resistance, and mating cycles, and derate for temperature.
3. Fluid connectors require matching thread standards. $NPT \neq BSP$. JIC 37° flare and compression fittings avoid the pitfalls of tapered-thread sealing.
4. Failure modes differ by domain: electrical joints fail from fretting corrosion and I^2R over-heating; fluid joints fail from cross-threading and seal degradation.
5. Every connector specification follows the same logic: **rating** → **environment** → **serviceability** → **cost** → **standards** → **space**. Document everything in your BOM.

Artifact	Description	Forward Reference
Connector Specifications	Selected connectors with ratings in BOM	Ch. 5 (ICDs)
Wire/Tube Routing	Harness design and fluid line layout	Assembly drawings

10.6 Artifact Handoff: What This Chapter Supports

Requirements Mapping: From Connector Specs to Shall-Statements

Connector specifications generate system requirements:

- “Outdoor deployment in rain” → SYS-REQ: All external connectors shall be rated IP67 or higher.
- “Field-replaceable sensor” → SYS-REQ: Sensor connections shall use keyed, tool-free connectors rated for ≥ 100 mating cycles.
- “Motor draws 8 A continuous” → CONN-REQ: Motor power connectors shall be rated for ≥ 10 A continuous (20% margin) with 12 AWG wire termination.
- “Hydraulic test pressure 200 psi” → CONN-REQ: All fluid fittings shall be rated for ≥ 800 psi working pressure ($4\times$ safety factor per hydraulic standard).

Every connector in your BOM must trace to a requirement. If you cannot write a shall-statement for it, you have not justified the selection.

Quick Reference: Common Connector Families

Family	Current	Voltage	Pitch	Typical Use
JST-PH	2 A	100 V	2.0 mm	LiPo batteries, small sensors
JST-XH	3 A	250 V	2.5 mm	Battery packs, board-to-wire
Molex KK	4 A	250 V	2.54 mm	General board-to-wire, fans
Molex Micro-Fit	8.5 A	600 V	3.0 mm	Motor power, high-current
Dupont/0.1”	3 A	250 V	2.54 mm	Breadboard jumpers, headers
USB-C	3–5 A	5–20 V	—	Data + power, standard interface
Barrel jack	5 A	12–24 V	—	Wall adapters, bench supplies
XT60	60 A	500 V	—	High-current battery, motors
Anderson PP	15–45 A	600 V	—	EV battery, solar, high-power
Deutsch DT	13 A	500 V	—	Automotive, harsh environment

10.7 Practice Questions for Quiz Preparation

Use these questions to test your understanding. Answers follow on the next page.

10.7.1 Conceptual Questions

1. What are the three fundamental elements of every electrical connector?
2. A connector is rated at 15 A per contact at 25 °C. Your design has an ambient temperature of 75 °C and all 12 contacts are loaded. Why should you *not* assume 15 A per contact is safe?
3. Explain why fretting corrosion is considered the “most insidious” electrical connector failure mode.

4. You need to connect a hydraulic line rated at 5000 psi that will be disconnected for maintenance monthly. Which fitting family is most appropriate? Why?
5. A colleague suggests using PTFE tape on a BSPP fitting. Is this correct? Explain.
6. What is the difference between rated working voltage and dielectric withstand voltage? Which should appear in your design documentation?
7. Why does NPS (Nominal Pipe Size) create confusion when specifying fittings?
8. Describe two mitigations for vibration fatigue in fluid tubing systems.
9. You measure 50 m Ω contact resistance on a connector carrying 8 A. How much power is dissipated as heat at that contact? Is this a concern?
10. Compare the sealing mechanisms of NPT, JIC 37° flare, and compression fittings. Which requires sealant?

10.7.2 Application / Scenario Questions

11. Your design team is designing an outdoor weather station. You need a 6-pin connector for sensor signals (all under 500 mA) that will be installed once and never disconnected. What IP rating, plating, and mating-cycle specification would you select? Justify your choices.
12. A pneumatic system in your project uses 1/4 in. OD nylon tubing at 80 psi. Recommend a fitting type and explain why.
13. Your design uses both NPT and BSP components (sourced from U.S. and European manufacturers). What steps must you take to ensure safe assembly?
14. You are specifying connectors for a cooling loop carrying a 50/50 ethylene glycol–water mix at 200 °F and 50 psi. What seal material would you select? What body material?
15. At your CDR, a reviewer asks: “How did you verify your crimp terminations are reliable?” Describe the test procedure you would reference.
16. **Diagnostic scenario:** Your team’s prototype passes a low-voltage continuity check on the bench (all connections show $<1\ \Omega$), but the system experiences intermittent sensor dropouts and a motor controller that occasionally faults when the prototype is running on a vibration table or driving over rough terrain. Where do you start your investigation, and what failure modes should you suspect? Describe the diagnostic steps you would take, referencing at least three concepts from this module.

10.7.3 Answers

1. Contacts (conductive pins/sockets), housing (insulating body), and termination (wire-to-contact joint).
2. Two derating effects compound: (a) elevated ambient temperature reduces the thermal headroom of the housing, and (b) thermal stacking from all 12 contacts being loaded raises the internal housing temperature further. The manufacturer's derating curve must be consulted; actual safe current per contact could be as low as 10 A or less.
3. Fretting corrosion develops slowly from micro-vibration, manifests as intermittent (not permanent) high resistance, is invisible without disassembly, and is difficult to reproduce during troubleshooting. By the time symptoms are consistent, oxide debris has accumulated extensively.
4. JIC 37° flare (SAE J514). It handles 10000 psi (well above 5000 psi), provides a metal-to-metal seal that doesn't degrade with reassembly, and is designed for repeated disconnect/reconnect without special tools beyond standard wrenches. Compression would also work but ferrules deform on initial assembly, making reassembly less reliable.
5. Incorrect. BSPP (G-type) uses parallel threads that do *not* seal on the threads. Sealing is achieved with a bonded washer or O-ring against a flat face. PTFE tape is used on *tapered* threads (NPT, BSPT) where the thread interference provides the primary seal. Applying tape to BSPP serves no useful purpose and may prevent proper washer compression.
6. Working voltage is the maximum voltage at which the connector is rated for continuous, safe operation, including creepage and clearance margins per IEC 61984. Dielectric withstand (breakdown) voltage is the voltage at which insulation fails; it is a destructive test limit, not an operating specification. Always specify working voltage in design documentation.
7. NPS is a nominal designation, not a physical dimension. For example, 1/2 in. NPS has an OD of 0.840 in. Engineers unfamiliar with this convention may attempt to match fittings by measuring OD, leading to incorrect specifications. Always reference the NPS table and verify with thread gauges.
8. (a) Install tube support clamps at manufacturer-recommended intervals to eliminate unsupported spans. (b) Include flexible hose segments at vibration sources (pumps, engines) to decouple cyclic motion from rigid tubing.
9. $P = I^2 R = (8)^2 \times 0.050 = 3.2 \text{ W}$. Yes, this is a concern; 3.2 W of localized heat will raise the contact and housing temperature significantly. This level of contact resistance suggests degraded contacts that should be inspected and potentially replaced.
10. NPT: Thread interference creates a wedge seal, but clearance at crests/roots requires supplemental sealant (PTFE tape or pipe dope). JIC 37°: Metal-to-metal cone seal at the flared tube end; no sealant required. Compression: Ferrule swaged onto tube OD creates the seal mechanically; no sealant required. Only NPT (and BSPT) requires sealant.
11. IP67 (dust-tight, survives 1 m immersion for rain/flooding protection). Gold plating (resists corrosion in outdoor environment with humidity and temperature cycling). 25–50 mating cycles (permanent installation, only disconnected for rare maintenance). Lower mating cycle connectors are less expensive and can be more robust.

12. Push-to-connect fittings. At 80 psi with nylon tubing, this is well within the 250 psi range of push-to-connect. Benefits: tool-free assembly, compatible with nylon tubing, easy reconfiguration, and widely available in 1/4 in. OD size from SMC, Festo, or John Guest.
13. (a) Create a thread identification chart for the project listing which ports are NPT and which are BSP. (b) Verify every connection with a thread pitch gauge ($60^\circ = \text{NPT}$, $55^\circ = \text{BSP}$) before assembly. (c) If adaptation is needed, use NPT-to-BSP adapter fittings from a reputable manufacturer. Never force-thread. (d) Label all ports on your assembly drawings with thread standard.
14. Seal: Viton (FKM); compatible with ethylene glycol, rated to 400°F (above the 200°F operating temperature). Body: stainless steel or brass (both compatible with glycol–water). Avoid carbon steel unless corrosion-protected, as glycol solutions can be mildly corrosive to unprotected ferrous metals.
15. Reference IPC/WHMA-A-620 standard. Procedure: (a) Pull-test a statistical sample of crimps at the specified force for the wire gauge and contact size. (b) Cross-section a sample crimp to verify the cold-weld zone under magnification. (c) Measure contact resistance across each mated pair with a milliohm meter; verify $<20\text{ m}\Omega$ for gold-plated contacts. (d) Document results with crimp force values, pass/fail criteria, and inspector sign-off.
16. This scenario synthesizes three interconnected failure modes. **Primary suspect: fretting corrosion.** The hallmark is a system that passes static bench testing but fails under dynamic load, exactly the intermittent, vibration-dependent behavior described in the fretting progression (Section 10.1.8, Figure 10.3). Diagnostic steps: (a) Measure contact resistance across every mated connector pair with a milliohm meter *while the system is subject to vibration* (tap on connectors, flex cables). Resistance that fluctuates from milliohms to ohms on disturbance confirms fretting. (b) Check for I^2R heating: use a thermal camera to identify hot connectors under load. A connector running 10 A through a fretting contact at $500\text{ m}\Omega$ dissipates 5 W, enough to soften a nylon housing. (c) Inspect strain relief: if cables transmit vehicle vibration directly to the contact zone, fretting is accelerated regardless of plating. (d) Check connector specifications: are tin-plated contacts being used in a high-vibration application where gold plating is warranted? Is the contact normal force sufficient? (e) For the motor controller faults specifically, consider whether the power connector is derated for the actual ambient temperature inside the enclosure under operation. Resolution: replace affected contacts with gold-plated equivalents, add proper strain-relief clamps, install secondary locking features (CPA clips), and retest under dynamic load with milliohm monitoring.

10.7.4 Application Scenario Answers

11. Redesigning a student project from soldered connections to connectors:

Step 1: Identify every wire-to-wire and wire-to-board connection in the system. Create a connection table: source, destination, signal type (power/signal/data), current, voltage, mating cycles expected.

Step 2: Group connections by subsystem boundary. Each subsystem-to-subsystem boundary gets one multi-pin connector (e.g., sensor module connects to controller via one 4-pin JST-XH: VCC, GND, SDA, SCL).

Step 3: Select connector families. For low-current signals ($<1\text{ A}$): JST-XH (wire-to-board, 2.5 mm pitch, rated 3 A, keyed). For power ($>1\text{ A}$): XT30 (rated 30 A, gendered, compact)

or Anderson Powerpole (genderless, stackable). For board-to-board: standard 2.54 mm pin headers with locking housings.

Step 4: Create a wiring diagram showing every connector with pin assignments, wire colors, and wire gauges. This diagram is a CDR deliverable.

Step 5: Label every connector pair with matching alphanumeric codes (J1/P1, J2/P2, etc.) on both the connector and the wiring diagram.

Benefits: any subsystem can be disconnected, tested independently, and reconnected. Failed modules can be swapped without soldering. The system can be disassembled for transport and reassembled on-site.

Chapter 11

Power Supplies, Motors & Pumps

When to Read This Chapter

Sprint 2–3: Read when sizing motors, pumps, or power supplies for your design. The motor/driver quick-reference table and the belt conveyor worked example are the most immediately useful. **Sprint 3 (CDR):** Verify your power budget using the battery sizing section. **Sprint 4:** Return to the motor driver section when wiring actuators.

11.1 Power Supply Selection

11.1.1 What a Power Supply Does

A power supply converts electrical energy from a source, AC mains, battery, solar panel, or USB, into one or more regulated DC voltage rails that your circuits and actuators require. Every component in your design has a specific voltage and current requirement. A typical student project might need 3.3 V for the microcontroller, 5 V for sensors and logic, and 12 V or 24 V for motors and solenoids. The power supply must deliver all of these simultaneously with adequate regulation and current capacity.

11.1.2 Linear Regulators (LDO)

Linear regulators are the simplest power supply topology. They are essentially a controllable series resistance that drops the input voltage to the desired output, dissipating the excess as heat.

$$P_{\text{waste}} = (V_{\text{in}} - V_{\text{out}}) \times I_{\text{load}}, \quad \eta = \frac{V_{\text{out}}}{V_{\text{in}}} \times 100\% \quad (11.1)$$

An LM7805 producing 5 V from 12 V at 1 A wastes 7 W as heat with only 42% efficiency. A low-dropout (LDO) regulator reduces the minimum input-to-output voltage difference to as little as 200 mV, improving efficiency when the differential is small.

When to Use an LDO

Use LDOs when: (1) the voltage drop is small ($< 2\text{ V}$), (2) current is under $\sim 1\text{ A}$, (3) noise matters (analog sensors, audio, ADCs), or (4) simplicity matters (only 2 capacitors, no inductor needed). Common devices: LM7805 (5 V/1 A), AMS1117-3.3 (3.3 V/1 A), MCP1700 (3.3 V/250 mA, ultra-low quiescent current).

11.1.3 Switching Regulators (DC-DC Converters)

Switching regulators chop the input voltage at high frequency (100 kHz–2 MHz) and use an inductor to transfer energy efficiently, achieving 85–95% efficiency.

Buck (Step-Down): $V_{\text{out}} < V_{\text{in}}$

The workhorse of DC-DC conversion. A switching FET, inductor, and capacitor step voltage down. Typical efficiency 85–95%. Common devices: LM2596 (3 A), MP1584 (3 A), TPS54302 (3 A). Pre-built module boards are available for under \$2; recommended for student projects because the critical PCB layout is already done.

Example: 12 V battery $\rightarrow$ 5 V at 2 A. At 90% efficiency, waste heat is only 1.1 W compared to 14 W for a linear regulator.

Boost (Step-Up): $V_{\text{out}} > V_{\text{in}}$

Stores energy in an inductor during the switch-on phase, then releases it at higher voltage during switch-off. Typical efficiency 80–93%. Common devices: MT3608 (2 A), TPS61088 (10 A). Important limitation: most boost converters cannot fully disconnect the load from the input.

Buck-Boost: $V_{\text{out}} \geq V_{\text{in}}$

Handles input voltage that swings above and below the output; essential for battery applications. A LiPo cell ranges from 4.2 V (full) to 3.0 V (depleted). If your circuit needs 3.3 V, the input crosses the output during discharge. Only buck-boost works through this transition. Common: TPS63020 (3.6 A), LTC3780.

EMI from Switching Regulators

All switching regulators generate electromagnetic interference at their switching frequency. This can corrupt ADC readings, cause radio interference, and inject noise into audio circuits. Always follow the datasheet layout recommendations: keep switching traces short, use a continuous ground plane, and place input/output capacitors as close to the regulator as physically possible. When powering noise-sensitive circuits, add an LDO downstream of the switcher as a noise filter.

11.1.4 AC-DC Power Supplies

For systems that plug into AC mains (120/240 VAC), you need an AC-to-DC conversion stage with galvanic isolation to separate lethal mains voltage from safe low-voltage outputs.

Student Project Rule

Never design your own AC-DC converter. Always use a pre-certified wall adapter (UL/CE listed) that provides a regulated DC output. These are fully enclosed, safety-certified, and eliminate all mains-voltage hazards from your design. If a wall adapter cannot meet your requirements, use an enclosed industrial supply (Mean Well, TDK-Lambda) with proper enclosure design.

LiPo Battery Safety: This Is Not Optional

A lithium-polymer battery that is overcharged, over-discharged, punctured, or short-circuited can undergo thermal runaway: **fire and toxic gas release** within seconds. A LiPo fire in a campus lab is a safety emergency that can end careers and close facilities. **Mandatory rules for MEGN 301:**

- **Never** charge a LiPo without a functional Battery Management System (BMS) with cell-level overvoltage, undervoltage, and overcurrent protection.
- **Never** leave a charging LiPo unattended.
- **Always** charge and store LiPo batteries in a fireproof LiPo safety bag.
- **Never** use a battery with a puffy or swollen pouch; it is generating gas internally.
- Violation of these rules is grounds for immediate lab access revocation.

11.1.5 Key Power Supply Specifications

Specification	Description
Output Voltage	Nominal output with tolerance (e.g., $5.0\text{ V} \pm 2\%$). Must match each load rail.
Output Current	Max continuous delivery. Must exceed total load current with $\geq 20\%$ margin.
Regulation	Line regulation: output change when input varies. Load regulation: output change when load varies. Expressed as % or mV.
Ripple & Noise	Peak-to-peak AC on the DC output. Switchers: 20–100 mV. LDOs: $<1\text{ mV}$.
Efficiency	$P_{\text{out}}/P_{\text{in}} \times 100\%$. LDO: 40–80%. Buck: 85–95%. Determines waste heat and battery life.
Isolation & Safety	Galvanic isolation for AC-DC. Look for UL/CSA/CE marks.

Table 11.1: Key power supply specifications.

11.1.6 Power Supply Failure Modes

- **Overcurrent:** Motor inrush can be 6–10 $\times$ running current. Size for peak, not average.
- **Thermal failure:** Every supply has a derating curve; output current decreases with ambient temperature.
- **Input transients:** Voltage spikes from motor switching. Mitigate with input capacitors and TVS diodes.
- **Capacitor aging:** Electrolytic caps dry out, increasing ripple. Use 105 °C rated caps.

11.2 DC Motors**11.2.1 Motor Fundamentals**

Every DC motor operates on the Lorentz force: a current-carrying conductor in a magnetic field experiences a force. Two equations govern motor behavior:

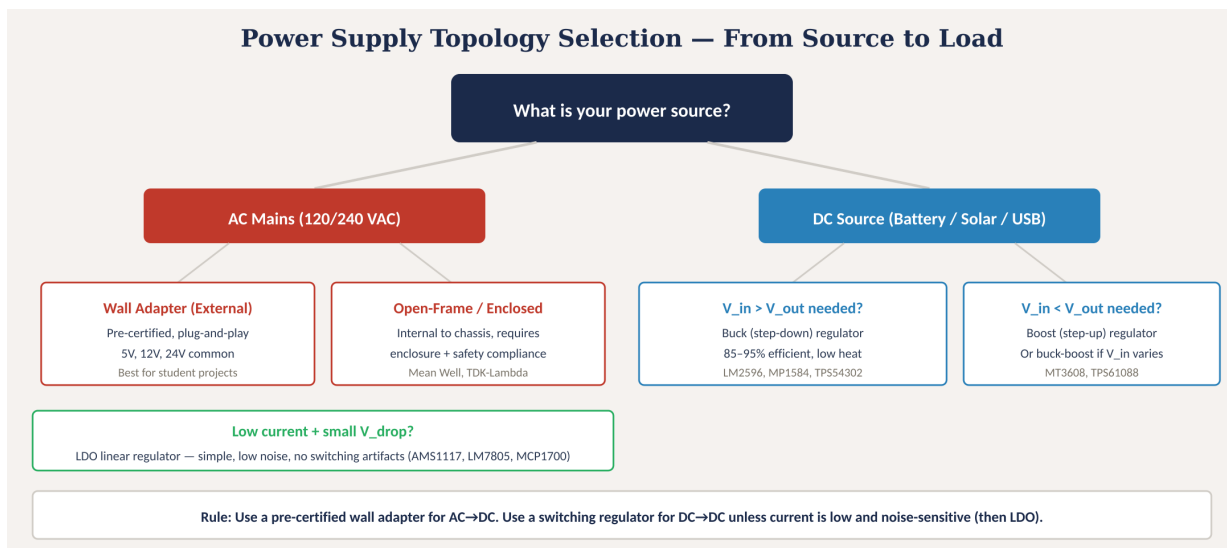


Figure 11.1: Power supply topology decision tree, from source type through voltage conversion to specific regulator selection.

$$\tau = K_t \times I \quad (\text{torque is proportional to current}) \quad (11.2)$$

$$V_{\text{back-EMF}} = K_e \times \omega \quad (\text{back-EMF is proportional to speed}) \quad (11.3)$$

At stall ($\omega = 0$), back-EMF is zero and current is limited only by winding resistance: $I_{\text{stall}} = V/R_{\text{winding}}$. This is typically 6–10× the rated running current and causes rapid overheating.

11.2.2 Brushed DC Motors

Simplest motor type. Mechanical commutation via carbon brushes on a copper commutator. Apply voltage, it spins; reverse polarity, it reverses. Speed control via PWM. The torque-speed curve is linear from stall torque at zero speed to no-load speed at zero torque.

Advantages: Cheap, simple to drive (just PWM + H-bridge), widely available.

Limitations: Brush wear limits lifetime (1,000–5,000 hours), EMI from brush arcing, moderate efficiency (70–85%).

11.2.3 Brushless DC (BLDC) Motors

Permanent magnets on the rotor, three-phase windings on the stator. An Electronic Speed Controller (ESC) sequentially energizes the stator phases. The motor *cannot function without its ESC*; they are an inseparable system.

Advantages: 85–95% efficiency, 10,000+ hour life, higher power density, flat torque curve in the mid-range, lower EMI.

Limitations: Requires ESC (adds cost and complexity), cannot be driven directly from DC, commutation tuning matters.

11.2.4 Stepper Motors

Rotate in discrete angular steps (typically $1.8^\circ = 200$ steps/rev). Open-loop positioning: count steps to know position without an encoder. NEMA 17 is the standard frame size for student projects. Driver boards (A4988, TMC2209) accept step/direction signals from any microcontroller.

Advantages: Precise positioning, high holding torque, no encoder needed.

Limitations: Torque drops rapidly with speed, runs hot (always energized), skips steps silently under overload.

11.2.5 Servo Motors

A motor + encoder + controller in a closed-loop system. Hobby servos (SG90, MG996R) are self-contained with PWM control. Industrial servos use BLDC motors with high-resolution encoders and dedicated controllers running PID loops at kHz rates.

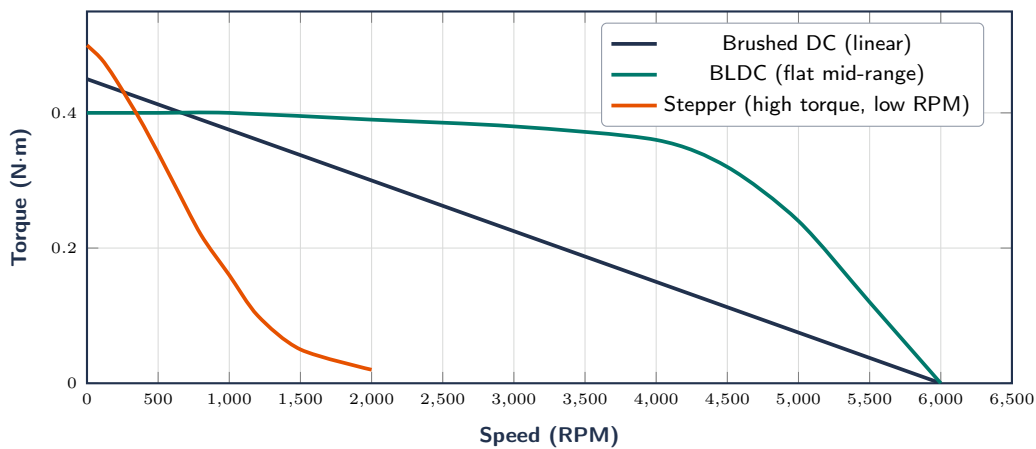


Figure 11.2: Torque–speed characteristics for brushed DC (linear), BLDC (flat mid-range), and stepper (high-torque at low RPM).

11.2.6 Key Motor Specifications

Specification	Description
Rated Voltage	Nominal operating voltage. Must match your supply rail.
No-Load Speed	Max speed with no load (RPM). Actual speed under load is lower.
Stall Torque	Max torque at zero speed. NOT your design point ; stall current overheats the motor.
Rated / Continuous Torque	Torque the motor delivers indefinitely. Your design ceiling. Derate to 50–75%.
Rated Current	Current at rated torque. Sizes your driver and power supply.
Torque Constant (K_t)	Torque per ampere (N·m/A). Calculate required current: $I = \tau_{\text{load}}/K_t$.

Table 11.2: Key motor specifications.

Never Design at Stall Torque

Stall current = $V_{\text{rated}}/R_{\text{winding}}$, typically 6–10× the rated running current. At stall, all electrical power converts to heat in the windings. A motor will overheat and fail within seconds to minutes. Always design at 50–75% of the *continuous* rated torque.

11.2.7 Motor Drive Electronics

Every motor type needs a matched driver circuit:

- **H-Bridge (Brushed DC):** Four transistors for bidirectional current. L298N (2A, legacy), BTS7960 (43A), DRV8871 (3.6A). **Must include flyback diodes**; back-EMF from inductive loads will destroy unprotected transistors.
- **ESC (BLDC):** Converts DC + throttle signal to three-phase AC. Drone ESCs are cheap and widely available. Use bidirectional or 4-quadrant ESC for reversing.
- **Stepper Driver:** Converts step + direction signals into motor phase currents. A4988 (2A), DRV8825 (2.5A), TMC2209 (2A, silent). **Set the current limit potentiometer before powering on.**
- **Servo Controller:** Hobby servos need only a PWM signal (1–2 ms pulse, 50 Hz). Industrial servos require dedicated controllers.

11.3 Pumps**11.3.1 Positive Displacement vs. Dynamic**

Positive displacement (PD) pumps trap a fixed volume per cycle and force it into the discharge. Flow is proportional to speed and nearly independent of pressure. Self-priming. Pulsating flow. Types: gear, diaphragm, peristaltic, piston, vane.

Dynamic (centrifugal) pumps impart kinetic energy to the fluid via a spinning impeller. Flow varies with discharge pressure (follows a pump curve). Smooth, continuous flow. *Not* self-priming. Best for high-volume, lower-pressure applications.

11.3.2 Pump Types**Gear Pumps**

Two meshing gears trap fluid between teeth and housing. Self-priming, handles viscous fluids (oil, glycol). Constant flow rate. Used in hydraulic systems and lubrication. Primary failure mode: wear from particle contamination. Always use inline filtration.

Diaphragm Pumps

A flexible diaphragm oscillates to create suction and discharge. Seal-less design isolates fluid from moving parts. Self-priming, handles corrosive and abrasive fluids. Pulsating flow requires dampeners.

Peristaltic Pumps

Rollers compress a flexible tube, pushing fluid forward. Zero contamination risk; fluid only contacts the tube. Easy maintenance: swap the tube. Gentle on shear-sensitive fluids. Limited to ~ 30 psi. Tube fatigue is the primary maintenance concern.

Centrifugal / Impeller Pumps

Spinning impeller accelerates fluid radially. Smooth, high-volume flow. Requires flooded suction (not self-priming). Poor with viscous fluids. Small 12V DC impeller pumps are widely available for student cooling and circulation projects.

Pump Type	Flow / Pressure	Strengths	Limitations	Best For
Diaphragm	Low-Med flow High pressure (60+ psi)	Self-priming Dry-run safe Pulsatile	Pulsating flow Vibration Diaphragm wear	Spraying Chemical dosing Pressure washing
Peristaltic	Low flow Low pressure (< 30 psi)	Zero contamination Easy maintenance Gentle on fluid	Tube fatigue Low pressure Pulsatile	Medical/bio Food-safe Shear-sensitive
Centrifugal	High flow Med pressure Smooth	Smooth flow High volume Simple design	Not self-priming Poor with viscous Cavitation risk	Cooling loops Circulation Water systems
Gear	Med flow High pressure (100+ psi)	Steady flow High pressure Self-priming	Wear particles Not for solids Noise	Hydraulics Oil transfer Fuel systems

Figure 11.3: Pump selection matrix; type, performance envelope, strengths, limitations, and best applications.

11.3.3 Key Pump Specifications

Specification	Description
Flow Rate	Volume per unit time (L/min, mL/min). Must meet demand with $\geq 25\%$ margin.
Discharge Pressure / Head	Max pressure (psi/bar) or total dynamic head (ft/m). Must exceed system losses.
Fluid Compatibility	Wetted materials must be compatible with the pumped fluid. Check seal charts.
Self-Priming	PD pumps: yes. Centrifugal: no (must be flooded).
Motor Integration	Many small pumps include integrated 12V/24V DC motors; specify the assembly.

Table 11.3: Key pump specifications.

11.3.4 Worked Example: Motor Selection for a Belt Conveyor

Requirements: Move 1 kg payload at 0.1 m/s on a horizontal belt conveyor. Belt drive with 50 mm diameter driven roller.

Step 1: Calculate load torque. Conveyor force = payload weight + belt friction = $(1.0 \times 9.81) + (1.0 \times 9.81 \times 0.3) = 12.75 \text{ N}$ (assuming $\mu = 0.3$ for belt on roller).

Torque at roller = $F \times r = 12.75 \times 0.025 = 0.319 \text{ N}\cdot\text{m}$.

Step 2: Calculate required speed. Roller angular velocity = $v/r = 0.1/0.025 = 4.0 \text{ rad/s} = 38.2 \text{ RPM}$.

Step 3: Select motor with margin. Required motor torque (with $2\times$ margin): $0.319 \times 2 = 0.638 \text{ N}\cdot\text{m}$ at 38 RPM. A typical NEMA 17 stepper (0.44 N·m rated) is insufficient. Options: (a) 12 V DC gearmotor with 50:1 ratio (Pololu 37D: 0.8 N·m at 50 RPM, \$25), or (b) NEMA 17 stepper with 5:1 planetary gearbox ($0.44 \times 5 \times 0.95 = 2.09 \text{ N}\cdot\text{m}$ at 40 RPM, \$40).

Step 4: Verify power budget. $P = \tau \times \omega = 0.319 \times 4.0 = 1.28 \text{ W}$ mechanical. At 70% motor efficiency: $P_{\text{electrical}} = 1.28/0.70 = 1.83 \text{ W}$. At 12 V: $I = 1.83/12 = 0.15 \text{ A}$ running. Stall current (Pololu 37D): 5.5 A. Power supply must handle 5.5 A inrush.

Step 5: Select driver. DC gearmotor: L298N H-bridge (2 A continuous, inadequate for 5.5 A stall). Better: BTS7960 (43 A, \$8). Include flyback diodes (integrated in BTS7960).

BOM entries: Motor: Pololu 4756 (12V 50:1 37D gearmotor), \$24.95. Driver: BTS7960 module, \$7.50. Mounting: Pololu 1084 bracket, \$7.49.

11.4 Selection and Design Practice

11.4.1 Unified Selection Framework

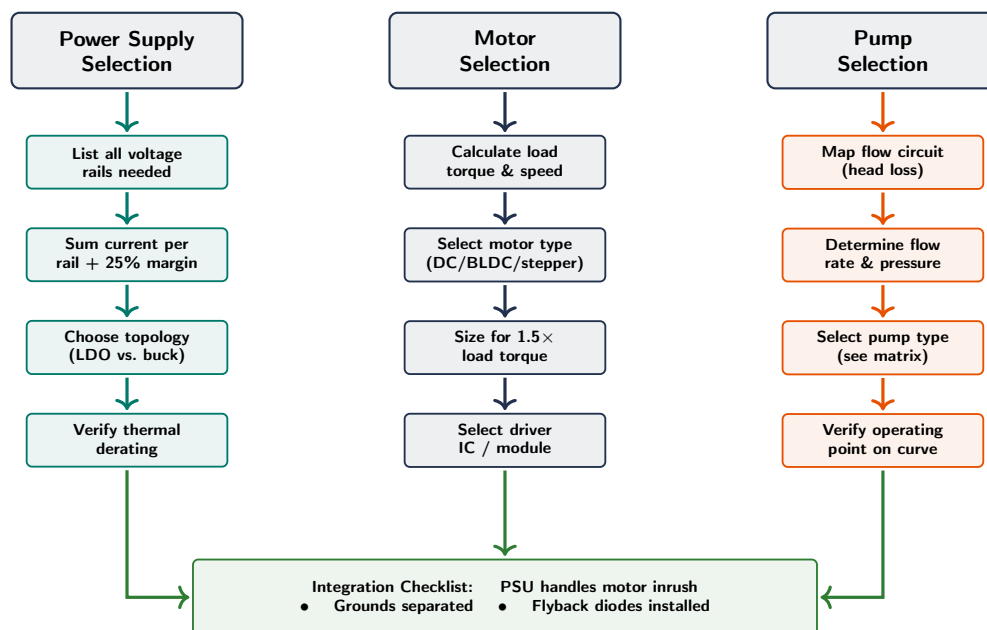


Figure 11.4: Parallel selection flows for power supply, motor, and pump with integration verification checklist.

11.4.2 Design Rules

1. **Build a power budget spreadsheet first.** List every component with its voltage rail, continuous current, and peak current. Sum the columns. Add 25% margin.
2. **Calculate mechanical load torque before selecting a motor.** $\tau = F \times r$ for linear-to-rotary. Include friction, inertia, and acceleration torque. Select a motor with rated torque $\geq 1.5 \times$ your load.
3. **For fluid systems, map your complete flow circuit,** pipe lengths, fittings, elevation, filters, to calculate total system head loss. Select a pump whose curve crosses your system curve with margin.
4. **Motor inrush is the #1 power supply problem.** A motor rated at 2 A running may draw 15 A at startup. Size your supply for inrush, or use soft-start circuits.
5. **Separate power and signal grounds.** Run motor power on dedicated wires back to the supply. Never share ground wires between motors and sensitive electronics.
6. **Document everything in your BOM:** part number, manufacturer, voltage, current, torque, flow rate, and procurement source.

Five Key Takeaways

1. Power supplies: use pre-certified wall adapters for AC-DC; switching regulators for efficient DC-DC; LDOs only for low-current, noise-sensitive loads.
2. Motors: match type to control need; brushed DC for simplicity, BLDC for efficiency, stepper for open-loop positioning, servo for closed-loop precision. Every motor needs a driver.
3. Pumps: positive displacement for precise flow and self-priming; centrifugal for high-volume smooth flow. Match wetted materials to your fluid.
4. Derating is non-negotiable: 20% current margin on power supplies, 25–50% torque margin on motors, 25% flow margin on pumps.
5. Integration: build a power budget first, size for inrush, separate power and signal grounds, document every selection in your BOM.

11.5 Artifact Handoff: What This Chapter Supports

Artifact	Description	Forward Reference
Power Budget	Voltage rails, current draws, margin analysis	Ch. 9 (protection), Ch. 5 (integration)
Motor Selection	Torque/speed calcs, driver selection, BOM entry	Ch. 11, Ch. 16 (p.163), BOM
Pump Selection	Flow/pressure calcs, wetted material verification	BOM, test plan

11.6 Power Supply Noise and Its Effect on Measurement

Every switching power supply generates output ripple at the switching frequency (typically 100 kHz–2 MHz) and its harmonics. This ripple propagates to every powered circuit, including sensitive analog signal conditioning.

Ripple specification: a “12 V, 5 A” switching supply may have 50–200 mV peak-to-peak ripple. For a 14-bit ADC with ± 10 V range ($\text{LSB} = 1.22$ mV), 50 mV of ripple is 41 LSBs of noise, potentially dominating the measurement uncertainty.

Mitigation: (1) use a linear post-regulator (LDO) between the switching supply and the analog circuits. The LDO’s PSRR (Power Supply Rejection Ratio) attenuates the ripple by 40–70 dB. A 50 mV ripple becomes < 0.05 mV after an LDO with 60 dB PSRR. (2) Add LC filtering: a 100 μH inductor followed by 100 μF capacitor provides ~ 40 dB attenuation at 100 kHz. (3) Physically separate the analog and digital power paths on the PCB, connecting them only at the power entry point.

11.7 Battery Selection for Portable Systems

Lithium Polymer (LiPo): 3.7 V nominal, 4.2 V charged, 3.0 V discharged. Energy density: 150–250 Wh/kg. Most common for student projects. Available in standard sizes from 500 mAh to 10,000 mAh. Requires BMS (Battery Management System) for charge control and overdischarge protection. **Never charge unattended; never charge above 4.2 V; never discharge below 3.0 V.**

Lithium Iron Phosphate (LFP/LiFePO₄): 3.2 V nominal. Lower energy density but inherently safer (no thermal runaway). Better cycle life (2000+ vs. 500 for LiPo). Preferred for applications where safety is paramount.

NiMH: 1.2 V nominal. Robust, safe, well-understood. Lower energy density than lithium. Good for applications where cost matters more than weight. Standard AA and AAA sizes simplify replacement.

Runtime estimation: $\text{runtime} = C_{\text{battery}}(\text{mAh}) / I_{\text{average}}(\text{mA})$ hours. A 2000 mAh battery powering a 200 mA average load: $2000 / 200 = 10$ hours. Derate by 20% for real-world conditions (self-discharge, temperature, aging): effective runtime ≈ 8 hours.

Quick Reference: Common Motor/Driver Combinations

Motor	Driver	Typical V/A	Notes
Pololu micro gearmotor	DRV8871	6–12 V / 3.6 A	Small robots, mechanisms
N20 gearmotor	TB6612FNG	3–12 V / 1.2 A	Dual channel, breadboard-friendly
NEMA 17 stepper	A4988 or DRV8825	8–35 V / 2 A	3D printer axes, linear stages
NEMA 23 stepper	TB6600	9–42 V / 4 A	CNC, heavier linear motion
12 V brushed DC	BTS7960	5–27 V / 43 A	High-current, bidirectional
Brushless (BLDC)	ESC (hobby-grade)	7–22 V / varies	Drones, pumps, fans
Small servo (SG90)	Direct PWM from MCU	5 V / 0.5 A	Position control, no driver IC
Large servo (MG996R)	Direct PWM + ext. supply	5–7 V / 2.5 A stall	Separate power rail from MCU

11.8 Practice Questions

11.8.1 Conceptual Questions

1. What is the fundamental difference between how a linear regulator and a switching regulator convert voltage?

Requirements Mapping: From Motor Curves to Shall-Statements

A motor's torque-speed curve directly generates testable requirements:

- “System shall move payload at 0.5 m/s” → Calculate wheel RPM from diameter, verify the motor's torque at that RPM exceeds load torque with $\geq 50\%$ margin.
- “Motor runs continuously” → MOT-REQ: Motor current at operating point shall be $\leq 75\%$ of continuous rated current. Calculate: $I_{req} = \tau_{load} / K_t$.
- “Battery life 4 hours” → PSU-REQ: Battery capacity $\geq I_{avg} \times 4 \times 1.3$ (aging margin).
- “Pump delivers 2 L/min at 10 psi” → PUMP-REQ: Pump shall deliver ≥ 2.5 L/min at ≥ 12 psi system head loss (25% margin).

If your requirement says “0.5 m/s” but the motor cannot produce the required torque at the corresponding RPM, the requirement and hardware are in conflict. The torque-speed curve resolves this *before* you order parts.

2. A 12 V motor has a winding resistance of 0.5Ω and a rated current of 4 A. What is the stall current? Why is this dangerous?
3. Explain why a BLDC motor cannot function without an ESC, while a brushed DC motor can operate with just a battery connection.
4. Your project needs to regulate a LiPo battery (3.0–4.2 V) down to 3.3 V. Why would a standard buck converter fail during battery discharge?

5. Compare the self-priming capability of gear pumps versus centrifugal pumps. How does this affect installation design?
6. Why must you include flyback diodes when driving a motor with an H-bridge? What happens without them?
7. A power supply is rated at 10 A at 25 °C. If your enclosure reaches 50 °C, can you still draw 10 A? Explain.
8. What is the primary advantage of a peristaltic pump over a gear pump for biomedical applications?

11.8.2 Application / Scenario Questions

9. Your design uses a 24 V wall adapter powering two DC motors (24 V, 3 A each running, 20 A stall each) and an Arduino (5 V, 200 mA). Design the power architecture: what regulators are needed, what is the minimum supply current rating, and how do you handle motor inrush?
10. You need to pump ethylene glycol coolant at 2 L/min through a closed cooling loop with 3 m of tubing, six fittings, and no elevation change. Which pump type do you recommend and why? What seal material?
11. Your team selects a NEMA 17 stepper for a linear stage. The required force is 20 N at 50 mm/s using a 10 mm lead screw. Calculate the required torque and verify whether a standard NEMA 17 (0.4 N·m rated) is adequate.
12. **Diagnostic scenario:** Your prototype's motor operates normally when powered from a bench supply but stutters and resets the microcontroller when powered from the project's 12 V battery pack. The battery is fully charged. What is the most likely cause, and how do you fix it?

11.8.3 Answers

1. A linear regulator dissipates excess voltage as heat (acts as a variable resistor). A switching regulator rapidly switches the input on/off and uses an inductor to transfer energy efficiently, achieving 85–95% efficiency versus 40–80% for linear.
2. $I_{\text{stall}} = V/R = 12/0.5 = 24 \text{ A}$; six times the rated current. At stall, all 288 W (24×12) converts to heat in the windings. The motor will overheat and fail within seconds.
3. A brushed DC motor has mechanical commutation (brushes and commutator) that automatically reverses current in the rotor windings. A BLDC motor's rotor has permanent magnets and the stator windings must be energized in a precise three-phase sequence by the ESC, which reads rotor position via Hall sensors or back-EMF sensing. Without the ESC, there is no commutation and no rotation.
4. A buck converter requires $V_{\text{in}} > V_{\text{out}}$ to regulate. When the LiPo discharges below 3.3 V (which happens as the cell depletes toward 3.0 V), the buck cannot maintain the output. A buck-boost converter is required because it handles the input crossing the output voltage.
5. Gear pumps are self-priming: they can draw fluid up from below the pump inlet by creating suction through the gear tooth spaces. Centrifugal pumps are not self-priming and require a flooded suction (pump inlet below fluid level). This means centrifugal pumps must be installed below the tank or use a foot valve and check valve to maintain prime.

6. A motor is an inductive load. When current is suddenly switched off, the collapsing magnetic field generates a voltage spike (back-EMF) that can be many times the supply voltage. Without flyback diodes, this spike destroys the H-bridge transistors. Flyback diodes clamp the spike by providing a current path for the decaying magnetic energy.
7. No. Every power supply has a thermal derating curve. At 50 °C, the supply has less thermal headroom to dissipate heat. A typical derating might reduce capacity to 70–80% of the 25 °C rating, meaning 7–8 A. Always check the datasheet derating curve.
8. The fluid in a peristaltic pump only contacts the inside of a disposable tube; no seals, rotors, or metal surfaces touch the fluid. This eliminates contamination risk and allows easy sterilization by swapping the tube. A gear pump has fluid contact with gears, housing, and shaft seals, making sterilization far more difficult.
9. Architecture: wall adapter provides 24 V rail directly to motors via H-bridges. A buck converter (e.g., LM2596) steps 24 V down to 5 V for the Arduino. Minimum supply current: continuous = $2 \times 3 + 0.2 = 6.2 \text{ A}$, but inrush = $2 \times 20 = 40 \text{ A}$ simultaneously. A 10 A supply with bulk capacitance ($\geq 4700 \mu\text{F}$) on the motor rail to absorb inrush, plus staggered motor startup in firmware (start one motor, wait 500 ms, start the second), is the practical solution.
10. A small DC centrifugal impeller pump is appropriate: the loop is closed (no priming issue if pump is at the bottom), flow is moderate, ethylene glycol is low viscosity, and smooth non-pulsating flow is preferred for cooling. Seal material: Viton (FKM) or EPDM; both compatible with ethylene glycol at moderate temperatures.
11. Torque = force $\times$ lead / ($2\pi \times$ efficiency). $\tau = 20 \times 0.010 / (2\pi \times 0.9) = 0.035 \text{ N}\cdot\text{m}$. The NEMA 17 at 0.4 N·m rated torque provides $0.4 / 0.035 = 11.4\times$ margin. However, at the required speed: $50 \text{ mm/s} / 10 \text{ mm/rev} = 5 \text{ rev/s} = 300 \text{ RPM}$. Check the stepper's torque-speed curve at 300 RPM; torque typically drops to 40–60% of holding torque, giving 0.16–0.24 N·m. Still adequate ($4.6\text{--}6.9\times$ margin), but verify on the actual datasheet curve.
12. Most likely cause: the battery cannot supply the motor inrush current. Unlike a bench supply (which can deliver virtually unlimited current), a battery has internal resistance. During motor startup, the inrush current (6–10 $\times$ running) causes a voltage drop across the battery's internal resistance, drooping the 12 V rail. If it drops below the buck converter's minimum input or the microcontroller's brown-out threshold ($\sim 2.7 \text{ V}$ for most AVR/ARM), the MCU resets. Fix: add a large bulk capacitor ($\geq 4700 \mu\text{F}$) on the battery rail close to the motor driver, use a battery with lower internal resistance (higher C-rating), implement soft-start in firmware, and add a separate regulated supply for the MCU with adequate input capacitance.

Chapter 12

Power Architecture & Circuit Safety

When to Read This Chapter

Sprint 3: Read the full chapter before CDR. The power budget, grounding strategy, and protection circuit sections directly determine whether your system will work reliably. **Sprint 4–5:** Return to the battery integration and EMI sections when debugging integration failures. **Related:** Circuit fundamentals in Chapter 13 (p. 138); motor/pump sizing in Chapter 11 (p. 112); PCB implementation in Chapter 18 (p. 186); debugging ground loops in Chapter 20 (p. 210).

Every electromechanical system needs a deliberately designed power architecture. A common student failure mode is treating power as an afterthought: “I’ll just wire everything to the battery.” This leads to brown-outs, ground loops, component damage, and debugging sessions that consume entire sprints.

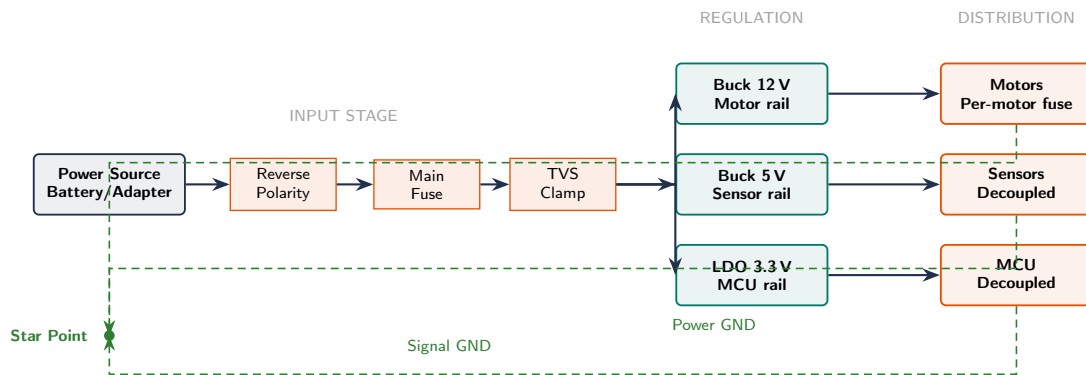


Figure 12.1: Typical student project power architecture showing three stages: input protection (reverse polarity, fuse, TVS), regulation (cascaded converters from highest to lowest voltage), and distribution (per-load protection). Ground returns converge at a single star point near the source.

12.1 Power Architecture

12.1.1 System-Level Design

A well-designed power system (Figure 12.1) flows from source through three functional stages before reaching the loads. The **input stage** is where power enters through a single, well-defined connector. This is where you place first-line protections: a keyed connector (prevents miswiring), reverse polarity protection, an input fuse, and a TVS diode for transient suppression. The **regulation stage** converts raw input voltage into the specific rails your system needs, commonly

cascading from the highest voltage down (e.g., 24 V input → 12 V motor rail → 5 V sensor rail → 3.3 V MCU rail). The **distribution stage** delivers regulated power to each load through individual protection, per-rail fuses or electronic current limiters that prevent a fault in one subsystem from cascading.

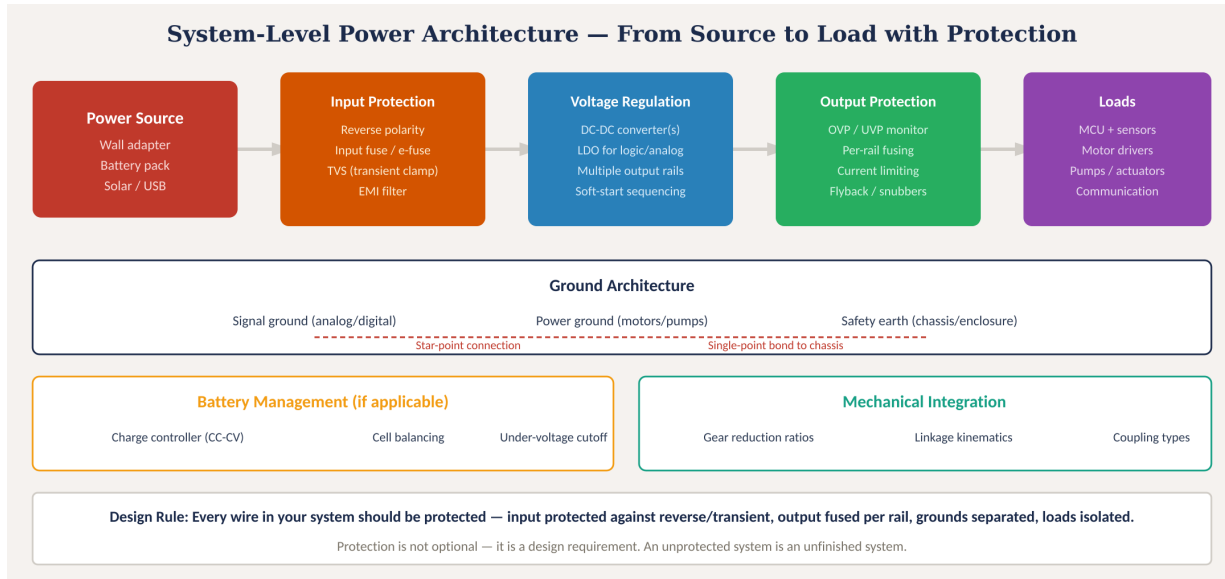


Figure 12.2: System-level power architecture from source to load with protection stages, ground architecture, battery management, and mechanical integration.

12.1.2 Grounding Strategy

Ground is not a magical zero-volt reference; it is a real conductor with real resistance and inductance. When a motor draws 5 A through a ground trace with 50 mΩ resistance, there is a 250 mV drop along that trace. If your ADC's ground reference shares that path, your sensor readings are corrupted.

Star-Point Grounding

Run separate ground conductors from each subsystem back to a single point near the power supply. Motor ground, sensor ground, and MCU ground each get their own conductor. They meet at *one star point*. This prevents motor return current from flowing through sensor ground paths. See also Chapter 20 (p. 210, Figure 20.3) for diagnosing ground-loop faults.

Three Ground Domains

- **Signal ground:** analog sensors, ADCs, digital logic. Low current, noise-sensitive.
- **Power ground:** motors, pumps, solenoids, heaters. High current, noisy.
- **Safety earth:** metal enclosure, chassis. Fault-current path for AC-powered systems.

The #1 Debugging Time Sink

Poor grounding is the single most common cause of difficult-to-diagnose intermittent failures in student projects. If your sensor readings are noisy, your MCU resets when a motor starts, or your serial communication drops characters; check your grounding first.

AC Mains Safety

Use a Class II (double-insulated) wall adapter whenever possible; no earth ground connection is needed. If you must use an internal open-frame supply, bond the metal enclosure to earth ground through a 3-prong inlet with proper strain relief. **Never defeat a ground prong.** AC mains voltage (120 V/240 V) is lethal. All AC wiring must use appropriately rated wire, fuses, and enclosures. If your project requires AC mains wiring, consult with the instructor before proceeding.

Signal Isolation

Use optocouplers or digital isolators (Si8620, ADUM1201) to pass signals between high-voltage and low-voltage domains. Essential when controlling AC loads (relays, triacs) from a microcontroller.

12.1.3 Battery Integration

Lithium cells (LiPo, Li-ion, LiFePO₄) store enormous energy density and can cause fire or explosion if mishandled. A Battery Management System (BMS) is **mandatory**:

- Overcharge cutoff: 4.2 V/cell maximum.
- Deep-discharge cutoff: 3.0 V/cell minimum (irreversible damage below this).
- Overcurrent / short-circuit shutoff.
- Cell balancing for multi-cell packs.

Common modules: TP4056 (single-cell charger, CC-CV profile), BQ25895 (buck-boost charger IC), BQ76940 (multi-cell BMS). During development, always store and charge lithium batteries in a fireproof LiPo-safe bag. **Never charge unattended without a BMS.**

DC Generation Sources (Solar, Alternator)

Solar panels produce variable voltage; use an MPPT charge controller to extract maximum power. Always include a blocking diode to prevent battery discharge through the panel at night. For alternators: rectify the AC output, regulate with a charge controller, and add TVS/MOV protection against load-dump voltage spikes if the battery disconnects while the alternator is running.

12.1.4 Power Budget Spreadsheet: How to Build One

The power budget is a spreadsheet with one row per component and the following columns:

The peak column sizes your regulators and fuses. The average column sizes your battery. Add 25% margin to both. If the SIM7000G's 2 A peak on the 4 V rail shares a regulator with the ESP32, that regulator must handle 2.24 A peak; a common sizing mistake.

Component	Rail	I_{typ} (mA)	I_{peak} (mA)	Duty (%)	I_{avg} (mA)
ESP32 (active)	3.3V	80	240	10	8.0
ESP32 (deep sleep)	3.3V	0.01	—	90	0.009
DS18B20 sensor	3.3V	1.5	1.5	100	1.5
SIM7000G (TX)	4.0V	200	2000	1	2.0
Relay coil	12V	80	80	5	4.0
3.3V rail total					9.5
3.3V rail peak					241.5
12V rail total					4.0

Table 12.1: GreenShield power budget example.

12.2 Protection Circuits

The philosophy is **defense in depth**: layer multiple protections so that each catches what the previous one missed. No single device guards against all threats.

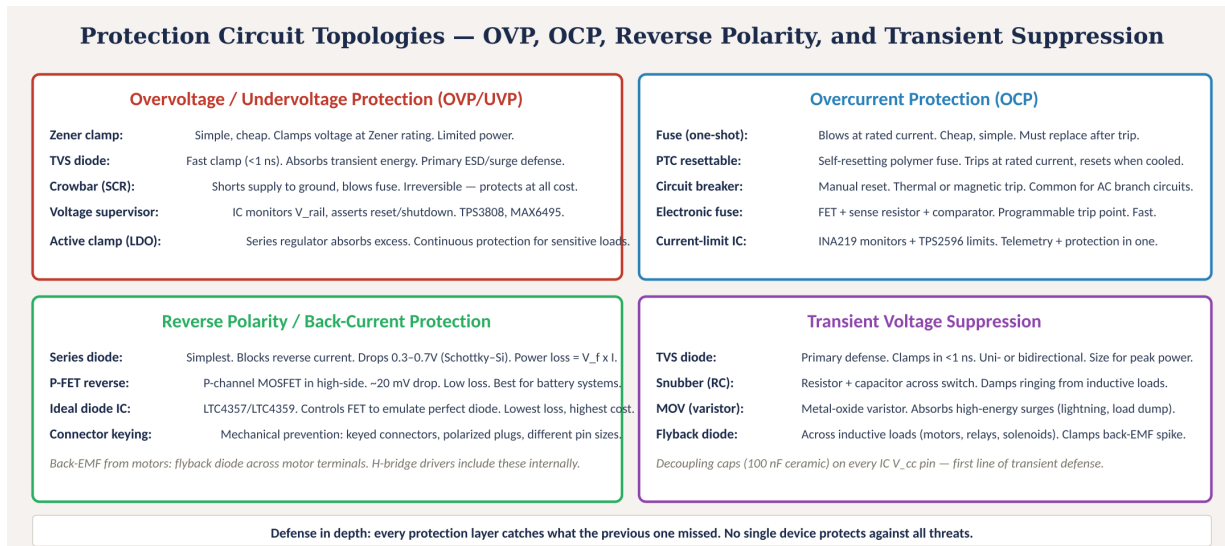


Figure 12.3: Four protection categories: OVP/UVLP, OCP, reverse polarity/back-current, and transient suppression with specific device options.

12.2.1 Overvoltage and Undervoltage Protection (OVP / UVLP)

Overvoltage pushes components beyond absolute maximum ratings; gate oxide breakdown, latch-up, permanent damage. **Undervoltage** causes unpredictable behavior: brown-outs, corrupted flash memory, and erratic execution.

Zener Clamp

A Zener diode across the supply rail conducts when voltage exceeds its breakdown value, clamping the rail. Simple and cheap but limited by power dissipation ($P = (V_{\text{spike}} - V_Z) \times I$). Select V_Z slightly above your nominal rail (e.g., 5.6 V Zener for a 5 V rail).

TVS Diode

A Transient Voltage Suppressor; essentially a heavy-duty bidirectional Zener optimized for high peak power. Responds in <1 ns. The primary defense against ESD, inductive spikes, and power supply transients. **Every power input in a professional design has a TVS diode.**

Selection: standoff voltage $\geq$ operating voltage; clamping voltage $<$ component absolute maximum rating. Common series: SMAJ, SMBJ, 1.5KE.

Voltage Supervisor IC

Monitors the supply rail and asserts a reset or shutdown signal when voltage is out of range. Does not clamp; it shuts the system down gracefully. Essential for microcontroller brown-out protection. Common: TPS3808 (adjustable OVP/UVP), MAX6495 (OVP with FET control), TLV803 (simple reset).

Crowbar Circuit

An SCR (thyristor) fires when voltage exceeds a threshold, shorting the supply to ground and blowing the upstream fuse. **Irreversible**; the fuse must be replaced. Used as a last resort when the overvoltage source has enough sustained energy to overwhelm a clamp.

12.2.2 Overcurrent Protection (OCP)

Every power rail must have current limiting. A short circuit without protection delivers the full energy of the supply into the fault.

Fuses (One-Shot)

A fusible element melts at its rated current. **Fast-blow** for sensitive electronics; **slow-blow** (time-delay) for motor circuits where inrush exceeds steady-state. The fuse must blow *before* the wire or component is damaged. Must be accessible for replacement.

PTC Resettable Fuses

Polymer PTC (positive temperature coefficient) devices. Resistance increases dramatically at the trip point, limiting current. Self-resets when cooled. Slower than traditional fuses. Best for USB ports and sensor circuits where the fault is intermittent.

Electronic Fuses (eFuse)

A MOSFET in series with the load, controlled by a current-sense amplifier. Programmable trip point, fast response, automatic retry or latched shutdown. Provides current *limiting* (not just cutting); can hold current at a safe level during motor inrush. Common: TPS2596, LTC4368, MAX14690.

Circuit Breakers

Thermal or magnetic trip mechanism with manual reset. Common for AC branch circuits (15 A, 20 A per NEC). In DC systems, used for battery main disconnects. **Must be DC-rated**; an AC breaker may not safely interrupt DC arcs because DC does not cross zero.

Fuse + Wire Sizing Rule

The fuse must blow before the wire overheats. This means the fuse rating must be *below* the wire's ampacity. Size them together using NEC Table 310.16 or AWG ampacity charts: the fuse protects the wire, the wire carries the current, and the load defines the requirement.

12.2.3 Reverse Polarity and Back-Current Protection

A battery connected backwards forward-biases body diodes of every downstream MOSFET and IC, often destroying the entire board in milliseconds.

Series Diode

Simplest approach. A Schottky diode ($\sim 0.3\text{ V}$ drop) or silicon diode ($\sim 0.7\text{ V}$ drop) in the positive rail blocks reverse current. Power loss = $V_f \times I_{\text{load}}$. Best for low-current ($< 2\text{ A}$) circuits where the voltage drop is acceptable.

P-Channel MOSFET

A P-FET in the high-side power path conducts when polarity is correct (body diode initially conducts, pulling gate low, fully enhancing the FET) and blocks when reversed. Voltage drop = $I \times R_{DS(\text{on})}$, typically $10\text{--}50\text{ m}\Omega$. At 5 A with $20\text{ m}\Omega$: only 0.1 V drop, 0.5 W loss. **Best for battery-powered systems.** Common: Si2301, IRLML6402, DMG2307L.

Ideal Diode IC

An IC controlling an external N-FET to emulate a perfect diode with $\sim 20\text{ mV}$ forward drop. Lowest loss, fastest reverse blocking, highest cost. LTC4357, LTC4359, MAX40200. Also used for OR-ing redundant supplies and solar/battery combining.

Mechanical Prevention

Keyed connectors, polarized plugs, and different-sized pins physically prevent reverse connection. The first line of defense. Always use polarized connectors for DC power.

12.2.4 Transient Voltage Suppression

Every time you switch an inductive load, motor, relay, solenoid, you create a voltage transient. The physics: $V = -L \times dI/dt$. Current cannot change instantaneously in an inductor, so the collapsing magnetic field generates a spike that can reach hundreds of volts from a 12 V motor.

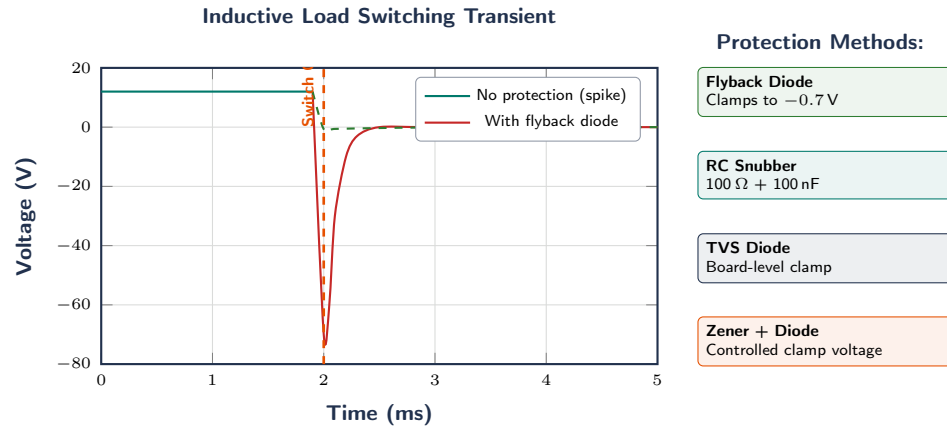


Figure 12.4: Back-EMF generation from inductive load switching and four protection solutions: flyback diode, RC snubber, TVS clamp, and Zener+diode.

Flyback Diode (Freewheeling Diode)

The single most important protection for inductive loads. A diode across the motor, relay, or solenoid (cathode to V+) provides a current path when the switch opens. Without it, the collapsing field generates a spike that destroys the switching device. Use a fast-recovery or Schottky diode. H-bridge driver ICs include these internally, but if you drive a relay or solenoid with a bare MOSFET, you **must** add one externally.

RC Snubber

A resistor-capacitor network across a switch or relay contact. The capacitor absorbs the initial spike; the resistor dissipates energy and damps oscillation. Typical starting values: $100\ \Omega + 100\ \text{nF}$ for relay contacts. Tune by observing the transient on an oscilloscope.

TVS Diode (Board-Level)

Place a TVS across each power rail at the board edge where external cables connect. Absorbs energy from ESD, cable transients, and motor switching spikes propagating through the power bus.

Decoupling Capacitors

100 nF ceramic capacitors on **every** IC V_{CC} pin, as close to the pin as physically possible. These absorb high-frequency transients before they reach the IC. For power regulators, add bulk capacitance (10–100 μF) per the datasheet. **Decoupling is not optional.**

12.3 Electromechanical Integration

12.3.1 Gear Reduction

Motors run efficiently at high speed and low torque. Loads typically need the opposite. Gearboxes trade speed for torque:

$$\tau_{\text{out}} = \tau_{\text{motor}} \times N \times \eta_{\text{gear}} \quad (12.1)$$

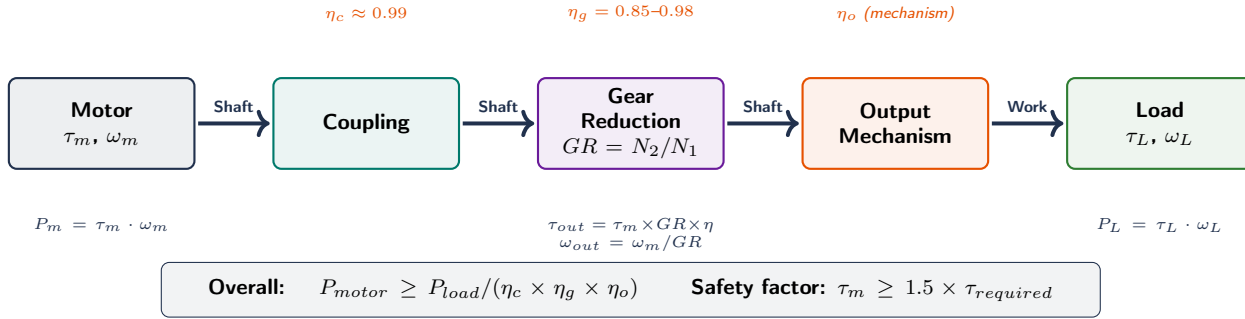


Figure 12.5: Motor-to-load power flow through coupling, gear reduction, and output mechanism with key sizing equations.

where N is the gear ratio (ω_{in}/ω_{out}) and η is the gearbox efficiency.

Type	Efficiency	Characteristics
Spur	95–98%	Simplest, cheapest. External parallel-axis mesh. Available in stock gear-motors.
Planetary	85–95%/stage	Compact, coaxial, high torque density. 3:1 to 100:1 per stage.
Worm	40–90%	Very high ratios (20:1–300:1) in one stage. Self-locking at high ratios.

Table 12.2: Gear reduction types and efficiencies.

12.3.2 Output Mechanisms

- **Shaft couplings:** rigid (zero misalignment tolerance) or flexible (jaw, Oldham, bellows, beam). Beam couplings are the standard for stepper motors.
- **Lead/ball screws:** convert rotary to linear. $F = 2\pi\tau\eta/\text{lead}$. Lead screws are self-locking; ball screws are not.
- **Timing belt drives** (GT2, HTD): positive engagement, no slip. Standard for CNC and 3D printers.
- **Linkages** (4-bar, crank-slider, cam): complex output paths. Torque varies through the cycle; size the motor for peak, not average.

Integration Rule

Start from the load and work backward: calculate the force/torque and speed the load requires → select the mechanism and gear ratio → calculate the motor torque → select the motor → calculate the current draw → size the power supply. The power check: $P_{mech} = \tau\omega = Fv$ must be $\leq P_{elec} \times \eta_{total}$.

12.4 Safety Design Checklist

Every item must be addressed before commissioning:

1. Input has reverse polarity protection (P-FET or series diode).
2. Input has a fuse or electronic current limiter rated for expected peak current.
3. TVS diode on every power input where external cables connect.
4. Voltage supervisor on MCU power rail (brown-out reset).
5. Flyback diode on every inductive load not protected by the driver IC.
6. Per-rail fusing; each output rail has its own current protection.
7. 100 nF decoupling capacitor on every IC V_{CC} pin.
8. Signal and power grounds separated, joined at a single star point.
9. Enclosure bonded to earth ground (if using internal AC-DC supply).
10. All wire gauges sized for their maximum current with appropriate derating.

Requirements Mapping: From Protection Specs to Shall-Statements

Protection circuit design generates system requirements:

- “System survives reverse battery” → SYS-REQ: The power input shall include reverse-polarity protection limiting leakage to <1 mA.
- “Motor inrush must not reset MCU” → PWR-REQ: The MCU supply rail shall maintain ≥ 2.7 V during motor inrush transients up to 15 A for 100 ms.
- “ESD at user connectors” → SYS-REQ: User-accessible interfaces shall withstand ± 8 kV contact discharge per IEC 61000-4-2.

Build your power budget spreadsheet *first*, then derive protection requirements from worst-case currents and voltages.

12.5 Grounding Architecture

The grounding strategy is the single most impactful decision for system noise performance:

Single-point (star) ground: all ground returns converge at one physical point. Prevents ground loops (current from one subsystem flowing through another’s ground return and creating noise). The star point is typically at the power entry connector. This is the standard for mixed analog/digital systems.

Multi-point ground: each subsystem connects to the nearest ground plane point. Used in high-frequency systems (>1 MHz) where the inductance of long ground wires to the star point would cause resonance. For MEGN 301 projects operating below 1 MHz: use star grounding.

Split ground plane: on the PCB, use a continuous ground plane under the analog section and a separate continuous ground plane under the digital section, connected at a single point (the star bridge). This prevents digital switching current from flowing under the analog circuits, which would inject noise.

12.6 Electromagnetic Compatibility (EMC) Basics

Every MEGN 301 system is both an EMI source (motors, switching regulators, digital clocks radiate) and an EMI victim (sensors, ADCs, communication lines are susceptible). Basic EMC design rules:

Minimize loop areas: the magnetic field radiated by a current loop is proportional to the loop area. Keep the current path and its return path as close together as possible. A ground plane provides the minimum-area return path for every trace above it.

Filter at boundaries: when a signal enters or leaves the system enclosure, it should pass through a filter (RC low-pass for analog signals, ferrite bead for digital signals, TVS diode for ESD protection). The enclosure boundary is where external EMI enters and internal EMI escapes.

Separate noisy and sensitive circuits: physically separate motor drivers, switching regulators, and high-speed digital circuits from analog sensors and amplifiers. On a PCB: place them on opposite sides. In an enclosure: use separate compartments or shielding barriers.

Decouple every IC: 100 nF ceramic capacitor within 5 mm of every IC power pin. This provides local charge storage for fast switching transients, preventing them from propagating through the power bus to other ICs.

Quick Reference: Common Protection Components

Component	Example	Use Case
TVS diode	SMBJ12A (12 V standoff)	ESD and transient voltage clamping on power rails
Schottky diode	SS34 (3 A, 40 V)	Reverse polarity protection (series); flyback (parallel)
P-FET reverse pol.	IRLML6402	Low-loss reverse polarity; $R_{DS(on)} < 50 \text{ m}\Omega$
PTC fuse (resettable)	MF-R050 (500 mA)	Signal line overcurrent; auto-resets after fault clears
Blade/glass fuse	ATC (automotive)	Main power rail; sized at 125% of max continuous current
eFuse IC	TPS2596	Programmable current limit, soft-start, fault reporting
Varistor (MOV)	14D471K (300 V)	AC mains transient suppression
Ferrite bead	BLM18PG221	High-frequency noise filtering on DC power lines

12.7 Practice Questions

12.7.1 Conceptual Questions

1. Explain why star-point grounding prevents motor noise from corrupting ADC readings. What happens if motor current returns through a shared ground trace?
2. A TVS diode has a standoff voltage of 12 V and a clamping voltage of 19.9 V at 10 A. Is this suitable for protecting a 12 V circuit whose components have a 20 V absolute maximum rating?
3. Compare a series Schottky diode versus a P-channel MOSFET for reverse polarity protection at 5 A. Calculate the power loss for each (Schottky $V_f = 0.3 \text{ V}$, P-FET $R_{DS(on)} = 20 \text{ m}\Omega$).

4. Why must a DC circuit breaker be specifically rated for DC voltage? What is the hazard of using an AC-rated breaker on a DC circuit?
5. A relay coil has 50 mH inductance and draws 100 mA at 12 V. If the switch opens in $1\text{ }\mu\text{s}$, estimate the back-EMF spike voltage ($V = L \times dI/dt$). Why does this destroy a 60 V MOSFET?
6. What is the difference between a crowbar circuit and a TVS diode clamp? When would you use each?
7. Explain why decoupling capacitors must be placed as close to the IC power pin as possible, not just “somewhere on the same board.”

12.7.2 Application / Scenario Questions

8. Your robot uses a 4S LiPo battery (14.8 V nominal, 16.8 V max charge, 12.0 V cutoff) powering two DC motors (5 A each continuous, 30 A each stall), a Raspberry Pi (5 V, 2.5 A), and three servos (6 V, 1 A each peak). Design the complete power architecture: draw a block diagram showing every rail, regulator, and protection device. Specify part numbers where possible.
9. Your project team’s prototype works perfectly on the bench but fails intermittently in the field. Symptoms: the motor runs fine, but the microcontroller occasionally resets, and analog sensor readings drift by 200 mV when the motor switches direction. Diagnose the root cause and propose a fix using concepts from this module.
10. A classmate argues that a flyback diode is unnecessary because “the motor driver IC already handles back-EMF.” Under what conditions is the classmate correct? Under what conditions is the classmate dangerously wrong?
11. Design a protection scheme for a solar panel (18 V open-circuit, 5 A short-circuit) charging a 3S LiPo (11.1 V nominal) that also powers a 12 V water pump. Include reverse-current blocking, charge control, and load protection.

12.7.3 Answers

1. In a shared ground, motor current (I_M) flows through the same trace as sensor return current. The trace has resistance R , so the sensor sees a ground voltage offset of $V_{\text{noise}} = I_M \times R$. With star-point grounding, motor current returns through its own dedicated conductor, so the sensor ground conductor carries only the small sensor current. The noise voltage on the sensor ground drops from $I_M \times R$ to $I_{\text{sensor}} \times R$, which is typically orders of magnitude smaller.
2. Yes; the standoff voltage (12 V) is at or above the operating voltage, so the TVS does not conduct during normal operation. The clamping voltage (19.9 V) is below the 20 V absolute maximum, so the TVS clamps before the components are damaged. However, the margin is very tight (0.1 V). A TVS with a 16.7 V or lower clamping voltage would provide better margin. Always verify the clamping voltage at the expected peak current, not just the rated value.
3. Schottky: $P = V_f \times I = 0.3 \times 5 = 1.5\text{ W}$. P-FET: $P = I^2 \times R_{DS} = 25 \times 0.020 = 0.5\text{ W}$. The P-FET dissipates one-third the power. At 5 A, the Schottky also drops the voltage rail by 0.3 V, which may push the rail below the minimum operating voltage of downstream components. The P-FET drops only 0.1 V. For battery-powered systems where efficiency and voltage headroom matter, the P-FET is strongly preferred.

4. AC current crosses zero 120 times per second (at 60 Hz), which naturally extinguishes any arc at the breaker contacts. DC current never crosses zero, so a DC arc can sustain indefinitely. An AC-rated breaker's contacts and arc chute may not be designed to extinguish a sustained DC arc, potentially causing the breaker to fail to interrupt the circuit, or worse, catch fire internally.
5. $V = L \times dI/dt = 0.050 \times (0.100/0.000001) = 5000 \text{ V}$. This 5 kV spike (in theory) far exceeds the 60 V MOSFET's drain-source breakdown voltage. In practice, the MOSFET avalanches and the spike is clamped at the breakdown voltage, but the energy dissipated during avalanche can destroy the device. A flyback diode clamps the spike to approximately $V_{\text{supply}} + V_f \approx 12.7 \text{ V}$, well within the MOSFET's rating.
6. A TVS diode *clamps* the voltage; it conducts just enough to limit the voltage and absorbs the transient energy, then stops conducting when the voltage returns to normal. Operation is non-destructive. A crowbar circuit *shorts* the supply to ground via an SCR, deliberately blowing the upstream fuse. It is destructive (fuse replacement required) but provides guaranteed protection when the overvoltage has enough sustained energy to overwhelm a TVS. Use TVS for transient spikes (ESD, inductive switching); use crowbar for sustained overvoltage from a failed regulator.
7. Decoupling caps work by providing a low-impedance path for high-frequency transients to bypass the IC rather than entering through the power pins. The trace between the capacitor and the IC pin has inductance proportional to its length. A long trace adds enough inductance to negate the capacitor's effectiveness at high frequencies. Placing the cap within a few millimeters of the pin minimizes this parasitic inductance and keeps the bypass impedance low at the frequencies where transients occur.
8. Architecture: Battery $\rightarrow$ main fuse (80 A, covers dual motor stall) $\rightarrow$ reverse polarity P-FET (IRLML6402 or similar) $\rightarrow$ TVS diode (18 V standoff) $\rightarrow$ power distribution bus. Motor rail: direct from battery through individual 40 A slow-blow fuses per motor, H-bridge drivers with integrated flyback diodes. 5 V rail: buck converter (e.g., LM2596 or MP1584, 3A rated) from battery bus, with 100 μF bulk cap + 100 nF ceramic decoupling. 6 V servo rail: buck converter (adjustable output set to 6 V, 5A rated, TPS54560) with per-servo PTC fuses. BMS: 4S BMS module with 12.0 V cutoff, 16.8 V overcharge protection, and cell balancing. Grounds: star-point near main fuse; separate motor ground bus and logic ground bus.
9. Root cause: ground-coupled motor noise. When the motor reverses direction via the H-bridge, the current transient creates a voltage spike on the shared ground that (a) momentarily shifts the MCU's ground reference, triggering a brown-out reset, and (b) offsets the ADC's reference voltage by 200 mV, corrupting readings. Fix: (a) separate motor power ground and MCU signal ground, joining at a star point near the battery; (b) add a voltage supervisor (TPS3808) on the MCU rail to hold the MCU in clean reset during transients rather than allowing corrupted execution; (c) add bulk capacitance (470 μF) on the MCU's supply rail to absorb the transient; (d) add a TVS diode on the motor power bus.
10. The classmate is correct when using a properly rated H-bridge driver IC (like L298N, DRV8871, BTS7960) that includes integrated flyback/freewheeling diodes across the output MOSFETs. These diodes handle the back-EMF from the motor. The classmate is **dangerously wrong** when: (a) driving a relay, solenoid, or valve directly from a bare MOSFET or BJT; these loads have no integrated protection; (b) the motor's back-EMF exceeds the driver's clamping capability; or (c) using a driver module where the "protection diodes" are actually for the logic

section, not the motor outputs. **Rule:** verify in the driver's datasheet that flyback diodes are explicitly rated for the motor's inductive energy.

11. Solar input: blocking diode (Schottky, e.g., MBR2045) prevents battery discharge through panel at night. MPPT charge controller (e.g., Genasun GV-5 or Victron 75/15) sized for 3S LiPo with 12.6 V charge termination. Reverse polarity: P-FET on both the solar input and the battery connector. TVS diode (18 V standoff) on the solar input for transient protection. Load: battery feeds pump through a slow-blow fuse (3 A for a 2 A pump). Flyback diode across pump motor terminals. Voltage supervisor monitors battery voltage and disables pump via a load switch when battery drops below 10.5 V to protect cells. The charge controller's MPPT algorithm ensures maximum solar harvest even as panel voltage varies with sun angle and temperature.

Chapter 13

Electronics Fundamentals, Safety & Component Selection

This chapter provides the foundational circuit theory, electrical safety practices, and practical component selection knowledge that supports every technical chapter in Part II. If you are already comfortable with Ohm's Law and Kirchhoff's Laws, use this chapter as a quick reference. If these concepts are new, work through the examples before tackling amplifier circuits, power architecture, or PCB design.

When to Read This Chapter

Sprint 3: Review Ohm's Law, KVL/KCL, and the voltage divider sections before designing circuits. Students who took MEGN 300 will find this material familiar; use it as a refresher and quick reference. **Sprint 4–5:** Return to the component selection and safety sections when building and debugging. **Related:** Power architecture and protection in Chapter 12 (p. 125); PCB layout in Chapter 18 (p. 186); debugging in Chapter 20 (p. 210).

13.1 Voltage, Current, Resistance, and Power

Voltage (V , measured in volts) is the electrical potential difference between two points; it is the “pressure” that pushes charge through a circuit. **Current** (I , measured in amperes) is the rate of charge flow: $I = dQ/dt$. Conventional current flows from higher to lower potential. **Resistance** (R , measured in ohms, Ω) quantifies how strongly a material opposes current flow. **Power** (P , measured in watts) is the rate of energy transfer: $P = VI$. In a resistor, $P = I^2R = V^2/R$.

Ohm's Law

The voltage across a resistor is proportional to the current through it:

$$V = IR \quad \Leftrightarrow \quad I = \frac{V}{R} \quad \Leftrightarrow \quad R = \frac{V}{I} \quad (13.1)$$

This is the single most-used equation in electronics. It applies to any purely resistive element: resistors, wire, heating elements, and the on-resistance of a MOSFET switch.

Conductance ($G = 1/R$, measured in siemens, S) is the reciprocal of resistance and is convenient when combining parallel resistances: $G_{\text{total}} = G_1 + G_2 + \dots$. **Energy** (E , measured in joules) is power integrated over time: $E = P \cdot t$. Battery capacity in milliamp-hours (mAh) converts to energy as $E = \text{mAh} \times V \times 3.6$ (in joules).

13.2 Kirchhoff's Laws

13.2.1 Kirchhoff's Current Law (KCL)

The algebraic sum of currents entering a node is zero:

$$\sum I_{\text{in}} = \sum I_{\text{out}} \quad (13.2)$$

Current is conserved. Whatever flows into a junction must flow out.

13.2.2 Kirchhoff's Voltage Law (KVL)

The algebraic sum of voltages around any closed loop is zero:

$$\sum V_{\text{loop}} = 0 \quad (13.3)$$

Energy is conserved. Voltage rises (sources) must equal voltage drops (resistors, loads) around any closed path.

KVL Applied: Series Resistor Circuit

A 12 V supply drives $R_1 = 4\text{ k}\Omega$ and $R_2 = 8\text{ k}\Omega$ in series. KVL: $V_s - V_{R1} - V_{R2} = 0$. Current: $I = 12/12,000 = 1\text{ mA}$. Then $V_{R1} = 4\text{ V}$ and $V_{R2} = 8\text{ V}$. Check: $12 - 4 - 8 = 0$. KVL satisfied.

13.3 Series and Parallel Circuits

13.3.1 Series Combination

Components in series carry the same current. Resistances add directly:

$$R_{\text{series}} = R_1 + R_2 + \cdots + R_n \quad (13.4)$$

13.3.2 Parallel Combination

Components in parallel share the same voltage. Reciprocals of resistance add:

$$\frac{1}{R_{\text{parallel}}} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_n} \quad (13.5)$$

For two resistors: $R_{\text{parallel}} = R_1 R_2 / (R_1 + R_2)$. The total is always less than the smallest individual resistor.

13.4 Voltage Dividers

Two series resistors connected across a source produce an output voltage at their junction:

$$V_{\text{out}} = V_{\text{in}} \cdot \frac{R_2}{R_1 + R_2} \quad (13.6)$$

Applications in MEGN 301: scaling sensor outputs to match ADC input ranges, creating reference voltages, and biasing amplifier circuits.

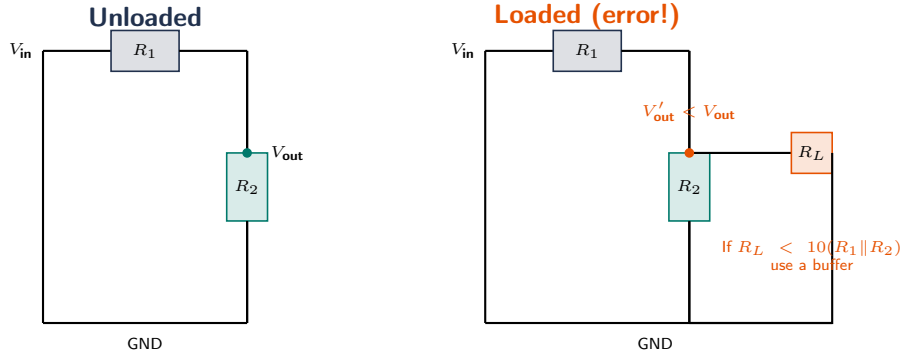


Figure 13.1: Voltage divider unloaded (left) vs. loaded (right). When a load R_L is connected, it appears in parallel with R_2 , pulling V_{out} below the ideal value. Rule of thumb: $R_L \geq 10 \times (R_1 \parallel R_2)$ for <10% loading error. If this condition cannot be met, insert an op-amp voltage follower between the divider and the load.

Loading a Voltage Divider

A voltage divider only produces the calculated voltage if the load draws negligible current compared to the divider current. Rule of thumb: the load resistance should be at least $10 \times$ the divider output resistance ($R_1 \parallel R_2$). If not, use a voltage follower (op-amp buffer) to isolate the divider from the load.

13.5 Thevenin Equivalent Circuits

Any linear circuit can be reduced to a single voltage source (V_{Th}) in series with a single resistance (R_{Th}). V_{Th} is the open-circuit voltage at the output terminals; R_{Th} is the resistance seen from those terminals with all independent sources zeroed. This matters for instrumentation because every sensor has an output impedance. Connecting a low-impedance load causes a voltage drop across R_{Th} , producing a loading error (the same phenomenon shown for voltage dividers in Figure 13.1). If $R_{Th} = 5 \text{ k}\Omega$ and the ADC input impedance is $1 \text{ M}\Omega$, the loading error is 0.5%; if the ADC impedance is only $10 \text{ k}\Omega$, the error jumps to 33%.

13.6 Capacitors and Inductors

A **capacitor** stores energy in an electric field: $Q = CV$, $I = C \cdot dV/dt$, $E = \frac{1}{2}CV^2$. It opposes changes in voltage, passes AC and blocks DC. Impedance: $Z_C = 1/(2\pi fC)$, decreasing with frequency. Types: ceramic (pF to μF , low ESR, ideal for decoupling), electrolytic (μF to mF, polarized, bulk storage), tantalum (compact, voltage-sensitive).

An **inductor** stores energy in a magnetic field: $V = L \cdot dI/dt$, $E = \frac{1}{2}LI^2$. It opposes changes in current, passes DC and blocks AC. Impedance: $Z_L = 2\pi fL$, increasing with frequency. Motor windings, solenoid coils, and relay coils are all inductors, which is why they produce voltage spikes when switched (Chapter 12).

13.7 RC and RL Time Constants

An RC circuit has time constant $\tau = RC$. The step response charges to 63.2% after 1τ and 99.3% after 5τ :

$$V_C(t) = V_{\text{final}} \left(1 - e^{-t/\tau}\right) \text{ (charging),} \quad V_C(t) = V_0 \cdot e^{-t/\tau} \text{ (discharging)} \quad (13.7)$$

An RL circuit has time constant $\tau = L/R$, with analogous exponential current response. Applications: low-pass input filtering before an ADC ($f_c = 1/(2\pi RC)$), switch debouncing ($\tau \approx 10$ ms), and understanding relay pull-in timing.

13.8 AC Fundamentals: Impedance, RMS, and Decibels

In AC circuits, resistance generalizes to **impedance** (Z), which has magnitude and phase. For a resistor: $Z_R = R$. For a capacitor: $Z_C = 1/(j\omega C)$. For an inductor: $Z_L = j\omega L$. Ohm's Law applies: $V = IZ$.

RMS values represent the equivalent DC value delivering the same power: $V_{\text{RMS}} = V_{\text{peak}}/\sqrt{2} \approx 0.707 \cdot V_{\text{peak}}$ for sinusoidal signals. US household power: $V_{\text{peak}} = 170$ V, $V_{\text{RMS}} = 120$ V.

Decibels: Gain (dB) = $20 \log_{10}(V_{\text{out}}/V_{\text{in}})$. Key values: +6 dB $\approx$ doubling voltage, -3 dB = half-power point (filter cutoff), -20 dB = voltage reduced by 10 $\times$. Decibels are additive through a signal chain.

13.9 Electrical and Electronics Safety

Electrical hazards in MEGN 301 projects range from minor (component damage from static discharge) to severe (shock, burns, or fire from mishandled power systems). The following practices are mandatory for all lab and project work.

13.9.1 Voltage and Current Hazards

The danger of electrical shock depends on current through the body, not voltage alone, but higher voltage drives more current through the body's resistance (~ 1 k Ω hand-to-hand, wet conditions). Thresholds: 1 mA is perceptible, 10–20 mA causes involuntary muscle contraction ("can't let go"), 75–100 mA across the heart is potentially fatal. At 120 VAC across 1 k Ω body resistance: $I = 120$ mA, well above the lethal threshold.

Safe voltage thresholds: below 30 VDC or 25 VAC RMS is generally considered safe for dry skin contact (IEC 61010). Between 30 V and 48 V, exercise caution: use insulated tools, avoid working on live circuits, and keep one hand in your pocket (the one-hand rule prevents hand-to-hand current paths across the heart). Above 48 V (including all AC mains), only qualified personnel should work on live circuits; de-energize and lock out/tag out before touching.

13.9.2 Benchtop Power Supply Safety

Always start with current limiting set to the minimum expected value before turning on the output. Increase slowly while monitoring. This prevents damage to your circuit and limits energy delivery during a fault. When using multiple supply channels, be aware that some supplies share a common ground between channels; connecting them in series to create higher voltages may short through the chassis. Verify isolation before stacking. After powering down, discharge any capacitors in your

circuit before touching. Large electrolytics ($100\text{ }\mu\text{F}+$ at $24\text{ V}+$) store enough energy to produce a painful spark and weld probe tips.

13.9.3 Multimeter Safety

Current-Mode Dead-Short Trap

If the multimeter is left in current mode (probes in the current jacks, near-zero internal impedance) and connected in parallel across a voltage source, the meter becomes a short circuit. This blows the meter's internal fuse (best case) or damages the meter and circuit (worst case). After every current measurement, move the red probe back to the voltage jack and switch the dial to voltage mode. This is the single most common equipment damage mode in the lab.

When measuring resistance, always de-energize the circuit first. The DMM's ohms mode injects a small current and measures the resulting voltage; if external voltage is present, the reading is meaningless and the meter may be damaged. For continuity checks (beep mode), the same rule applies.

13.9.4 Oscilloscope Ground Clip Hazard

The oscilloscope probe's ground clip is connected directly to earth ground through the scope chassis and the AC power cord. If you clip the ground lead to a node that is not at circuit ground, you create a short circuit through the earth ground path. This can destroy components, blow fuses, or trip breakers. The ground clip must always connect to circuit ground and nowhere else. To measure a floating voltage (voltage across a component not referenced to ground), use the math function to subtract two channels, or use a differential probe.

13.9.5 Static Discharge (ESD) Protection

CMOS components (including the ESP32, most op-amps, and all modern ICs) are sensitive to electrostatic discharge. A human body can accumulate $3,000\text{--}25,000\text{ V}$ of static charge; discharging through an IC pin destroys the gate oxide. Mitigation: wear a grounded wrist strap when handling bare ICs and PCBs, store ICs in anti-static bags or foam until ready for assembly, touch a grounded metal surface before picking up components, and avoid working on carpet or synthetic surfaces that generate static.

13.9.6 Lithium Battery Safety

Lithium-ion and lithium-polymer cells are used in many MEGN 301 projects. They offer high energy density but present fire and explosion risks if mishandled. Mandatory practices: never short-circuit a lithium cell (even momentarily), never charge above the rated voltage (typically 4.2 V per cell; overcharge causes thermal runaway and fire), never discharge below the minimum voltage (typically 3.0 V ; deep discharge causes internal copper dendrite growth that can short the cell on the next charge cycle), always use a proper battery management system (BMS) or charge controller IC, store cells at $3.7\text{--}3.8\text{ V}$ per cell (approximately 40–60% charge) for long-term storage, and never puncture, crush, or expose cells to temperatures above 60°C .

Thermal Runaway

If a lithium cell is swelling, hissing, smoking, or feels abnormally hot: do not touch it, clear the area, place it in a fire-safe container (metal bucket with sand if available), and notify lab staff. Lithium fires cannot be extinguished with water; use a Class D extinguisher or dry sand. Include lithium battery handling in your project risk register and have a response plan documented before bringing cells into the lab.

13.9.7 Fire Safety for Electronics

Electronics fires most commonly result from: overcurrent through undersized wire (wire heats, insulation melts, fire starts), short circuits with no fuse protection (uncontrolled energy release), lithium battery failure (thermal runaway), and component overvoltage causing arcing. Prevention: fuse every power rail at or below the wire’s current rating, use wire rated for the voltage and current, inspect solder joints for bridges before powering up, and keep a fire extinguisher (CO₂ or dry chemical, suitable for Class C electrical fires) accessible in the workspace.

13.10 Resistor Color Code

Through-hole resistors use colored bands to encode their value. The standard 4-band code: first two bands are significant digits, the third is the multiplier, and the fourth is tolerance.

Color	Digit	Multiplier	Tolerance
Black	0	×1	—
Brown	1	×10	±1%
Red	2	×100	±2%
Orange	3	×1,000	—
Yellow	4	×10,000	—
Green	5	×100,000	±0.5%
Blue	6	×1,000,000	±0.25%
Violet	7	×10,000,000	±0.1%
Gray	8	—	±0.05%
White	9	—	—
Gold	—	×0.1	±5%
Silver	—	×0.01	±10%

5-band resistors (precision, 1% or better) use three significant digits, a multiplier, and a tolerance band.

Reading Resistor Color Codes

4-band: Brown–Black–Orange–Gold = $10 \times 1,000 = 10\text{ k}\Omega$, ±5% (range: 9.5 kΩ to 10.5 kΩ).
4-band: Red–Violet–Red–Gold = $27 \times 100 = 2.7\text{ k}\Omega$, ±5%.
5-band: Brown–Red–Black–Brown–Brown = $120 \times 10 = 1.2\text{ k}\Omega$, ±1%.
Mnemonic (digit order 0–9): “Better Be Right Or Your Great Big Venture Goes Wrong.”

SMD Resistor Markings

Surface-mount resistors use a numeric code. The 3-digit code (e.g., “103”) uses the first two digits as significant figures and the third as the number of zeros: $103 = 10 \times 10^3 = 10 \text{ k}\Omega$. The 4-digit code (e.g., “4702”) uses three significant digits: $470 \times 10^2 = 47 \text{ k}\Omega$. “R” indicates a decimal point: “4R7” = 4.7Ω .

13.11 Component Selection Reference

13.11.1 Resistor Tolerance and Standard Values

Resistors come in standard value series spaced logarithmically. E12 (10% tolerance): 12 values per decade (10, 12, 15, 18, 22, 27, 33, 39, 47, 56, 68, 82). E24 (5%): 24 values. E96 (1%): 96 values. When designing a divider or filter, choose the nearest standard value and verify the circuit meets specifications at the tolerance extremes.

13.11.2 Power Rating

Every resistor has a maximum power dissipation. Standard through-hole: $1/4 \text{ W}$. Verify $P = I^2 R$ or $P = V^2/R$ at worst-case conditions. Exceeding the rating causes the resistor to overheat, change value, and eventually fail.

13.11.3 Capacitor Voltage Rating

Every capacitor has a maximum working voltage. Exceeding it causes dielectric breakdown. Derate by 50%: use a 25 V cap on a 12 V rail. Electrolytic capacitors are polarized; reversing polarity causes catastrophic failure.

13.11.4 Wire Gauge Reference (AWG)

AWG	Diameter (mm)	Max Current (A)	Typical Use
30	0.25	0.5	Wire-wrap, fine signal wiring
28	0.32	0.8	Ribbon cable, breadboard jumpers
24	0.51	2.1	Standard signal and sensor wiring
22	0.64	3.0	General purpose hookup wire
20	0.81	5.0	Power wiring, LED strips
18	1.02	7.5	Motor power, moderate loads
16	1.29	10	High-current motor, battery leads
14	1.63	15	High-power supply, automotive

Current ratings are approximate for chassis wiring in open air. Bundled wires should be derated by 20–40%. Use stranded wire for any connection subject to movement or vibration; solid wire work-hardens and breaks after repeated flexing.

13.11.5 SI Prefixes

Prefix	Symbol	Factor	Common Usage
pico	p	10^{-12}	Capacitance (pF)
nano	n	10^{-9}	Capacitance (nF), time (ns)
micro	μ	10^{-6}	Voltage (μ V), current (μ A)
milli	m	10^{-3}	Voltage (mV), current (mA)
—	—	10^0	Base unit (V, A, Ω , Hz)
kilo	k	10^3	Resistance ($k\Omega$), frequency (kHz)
mega	M	10^6	Resistance ($M\Omega$), frequency (MHz)
giga	G	10^9	Frequency (GHz)

Case Sensitivity

Lowercase “m” means milli (10^{-3}). Uppercase “M” means mega (10^6). Confusing them changes a value by a factor of one billion. A 4.7 m Ω current-sense shunt is nine orders of magnitude different from a 4.7 M Ω input bias resistor. Always verify case when reading datasheets.

13.11.6 Common Schematic Symbols

Passive: resistor (zigzag or rectangle), capacitor (two parallel lines; curved plate for polarized electrolytic), inductor (series of loops), fuse (thin line with break). **Semiconductors:** diode (triangle into bar), LED (diode with arrows), Zener (bent bar ends), NPN BJT (emitter arrow away from base), N-MOSFET (arrow inward on body diode). **Sources:** DC voltage (circle with +/–), AC (circle with sine wave), ground (three horizontal lines). **Active:** op-amp (triangle with +/– inputs).

13.12 Grounding and Ground Loops

Ground is the 0 V reference node. In practice, ground conductors carry return currents and have nonzero resistance, creating small voltage differences between ground points in different parts of the circuit. A ground loop occurs when multiple ground connections form a closed loop that picks up magnetically induced noise from nearby motors, transformers, or switching supplies.

Star Grounding

Connect all subsystem grounds to a single point near the power supply rather than daisy-chaining. Keep signal ground (sensors, ADC) and power ground (motors, relays) on separate conductors that meet only at the star point. This prevents motor return currents from corrupting sensor measurements.

13.13 Multimeter and Oscilloscope Quick Reference

Voltage: probes in parallel across the component. DMM presents >1 M Ω input impedance. **Current:** meter in series (break the circuit). Red probe must be in the current jack. **Resistance:** de-energize the circuit first; DMM injects a test current. **Continuity:** beeps if <50 Ω ; circuit must be de-energized.

Oscilloscope: the ground clip connects directly to earth ground through the power cord. Never clip it to anything other than circuit ground or you will create a short circuit through the earth path. To measure floating voltages, use channel math (Ch1 – Ch2) or a differential probe.

13.14 Breadboard Conventions

The two long rails along the edges are bus strips (power and ground). Central rows of five-hole groups connect horizontally within each group but not across the center channel, which is sized to straddle DIP IC packages. Best practices: red rail for $V+$, blue for ground, short flat jumper wires color-coded by function, ICs straddling the center channel. Limitations: contact resistance 0.01–0.1 Ω per connection (adds up), stray capacitance 2–5 pF between rows (limits to <1 MHz), and connections loosen over time. Never use a breadboard as a final deliverable at FDR.

Requirements Mapping: Electrical Safety and Component Selection

“All power rails shall be fused at or below the wire current rating per AWG specifications.” “The system shall include a physical power disconnect that de-energizes all actuators independently of firmware.” “Resistors in signal conditioning paths shall be 1% tolerance (E96 series) or better.” “Decoupling capacitors shall have a voltage rating at least $1.5\times$ the rail voltage.” “All lithium battery installations shall include a battery management system with overvoltage, undervoltage, and overcurrent protection.”

Quick Reference: Essential Electronics Values

Parameter	Value / Rule
Ohm's Law	$V = IR, P = VI = I^2R = V^2/R$
Resistors in series	$R_{total} = R_1 + R_2 + \dots$
Resistors in parallel	$1/R_{total} = 1/R_1 + 1/R_2 + \dots$
Voltage divider	$V_{out} = V_{in} \cdot R_2/(R_1 + R_2)$; load $R_L \geq 10(R_1 \parallel R_2)$
RC time constant	$\tau = RC$; 63% at 1τ , 99% at 5τ
Capacitor impedance	$Z_C = 1/(2\pi fC)$; lower at higher frequency
Decoupling cap	100 nF ceramic ≤ 5 mm from every IC power pin
Pull-up resistor	4.7 k Ω for I2C; 10 k Ω for general GPIO
LED current-limiting	$R = (V_{supply} - V_{LED})/I_{LED}$; typical: 220 Ω at 5 V, 68 Ω at 3.3 V
Wire current (chassis)	22 AWG: 3 A; 18 AWG: 7 A; 14 AWG: 15 A
Safe voltage threshold	<30 VDC / 25 VAC RMS (IEC 61010)

13.15 Practice Questions

1. A 9 V battery drives $R_1 = 1 \text{ k}\Omega$ in series with the parallel combination of $R_2 = 2 \text{ k}\Omega$ and $R_3 = 3 \text{ k}\Omega$. Find the total current and voltage across each element.
2. Identify the value and tolerance of a resistor with bands: Green–Blue–Red–Gold. Then identify: Orange–Orange–Brown–Brown–Brown (5-band).
3. Your sensor has a Thevenin output of $V_{Th} = 2.5 \text{ V}$ and $R_{Th} = 5 \text{ k}\Omega$. What voltage does an ADC with 1 M Ω input impedance read? What about 10 k Ω input impedance? What is the fix?

13.15.1 Answers

1. $R_{2\parallel 3} = (2 \times 3)/(2 + 3) = 1.2 \text{ k}\Omega$. $R_{\text{total}} = 1.0 + 1.2 = 2.2 \text{ k}\Omega$. $I = 9/2200 = 4.09 \text{ mA}$. $V_{R1} = 4.09 \text{ V}$. $V_{R2\parallel 3} = 4.91 \text{ V}$. KVL check: $4.09 + 4.91 = 9.0 \text{ V}$.
2. Green–Blue–Red–Gold: $56 \times 100 = 5.6 \text{ k}\Omega$, $\pm 5\%$. Orange–Orange–Brown–Brown–Brown: $331 \times 10 = 3.31 \text{ k}\Omega$, $\pm 1\%$.
3. With $1 \text{ M}\Omega$: $V = 2.5 \times 10^6/(5000 + 10^6) = 2.488 \text{ V}$ (0.5% error, negligible). With $10 \text{ k}\Omega$: $V = 2.5 \times 10,000/(5000 + 10,000) = 1.667 \text{ V}$ (33% error, unacceptable). Fix: add a unity-gain buffer (voltage follower op-amp) between the sensor and the ADC to present high impedance to the sensor and low impedance to the ADC.

Chapter 14

Principles of Sensors and Actuators

When to Read This Chapter

Sprint 2–3: Read when selecting sensors and actuators for your design. The six-criteria selection framework and the common sensor types quick-reference table are your starting points. **Sprint 4:** Return to the actuator driver section before wiring motors, solenoids, or heaters.

This chapter covers the physical principles of transduction: how sensors convert physical quantities into electrical signals and how actuators convert electrical commands into physical action. Understanding these principles allows you to select the right component for a given requirement, predict its limitations, and design the signal conditioning or driver circuit that bridges it to your controller.

14.1 Transduction: Converting Between Physical Domains

A **transducer** converts energy from one domain to another. A **sensor** (input transducer) converts a physical quantity (temperature, pressure, strain, displacement, flow, light, acceleration) into an electrical signal. An **actuator** (output transducer) converts an electrical command into physical action (force, torque, displacement, heat, flow). The choice of transduction mechanism determines sensitivity, range, linearity, bandwidth, and failure modes.

14.2 Sensor Classification by Transduction Mechanism

Resistive: the measurand changes resistance. Examples: strain gauges (piezoresistive effect), RTDs (temperature coefficient of platinum), thermistors (semiconductor oxide), potentiometers (wiper position). Output is measured with a Wheatstone bridge or voltage divider.

Capacitive: the measurand changes capacitance by varying plate area, gap, or dielectric. Examples: MEMS accelerometers (proof mass changes gap), humidity sensors (dielectric absorbs moisture), touchscreens (finger changes fringe capacitance). Measured with oscillator circuits or capacitance-to-digital converters.

Inductive/Electromagnetic: the measurand changes inductance or generates voltage via Faraday's law. Examples: LVDTs (displacement changes transformer coupling), tachometers (rotation generates AC voltage), eddy current proximity sensors.

Piezoelectric: mechanical stress generates charge (direct piezoelectric effect). Examples: dynamic force/pressure sensors (quartz, PZT), accelerometers, ultrasonic transducers. Critical limitation: piezoelectric sensors measure only *dynamic* quantities; charge leaks for static loads.

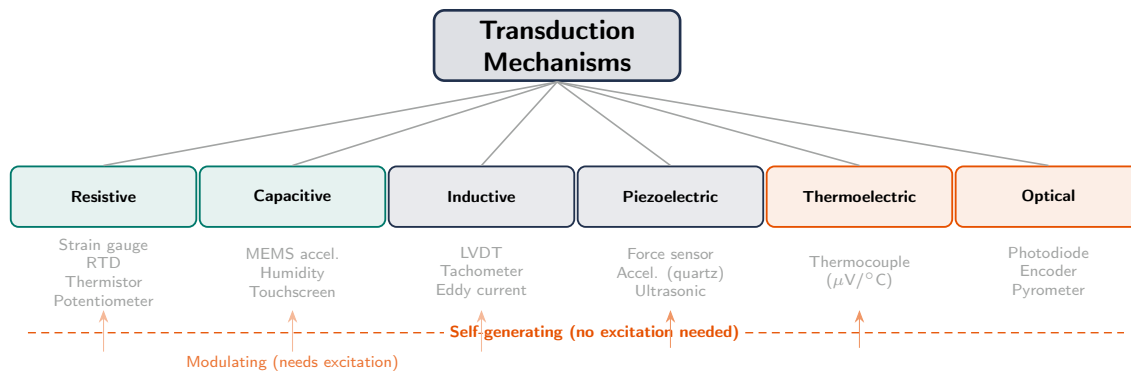


Figure 14.1: Sensor classification by transduction mechanism with examples and output type. Self-generating sensors (piezoelectric, thermoelectric) produce signal energy from the measurand without external excitation. Modulating sensors (resistive, capacitive, inductive) require an excitation source whose output the measurand modifies.

Thermoelectric: temperature difference at a junction of dissimilar metals generates voltage (Seebeck effect). The thermocouple produces $\sim 10\text{--}60\ \mu\text{V}/^\circ\text{C}$ depending on type, requiring amplification and cold-junction compensation.

Optical/Photoelectric: light intensity, wavelength, or interference pattern encodes the measurand. Examples: photodiodes (light to current), fiber-optic sensors (strain changes transmission), pyrometers (infrared radiation indicates temperature without contact), encoders (slotted disc produces digital pulses).

14.3 Sensor Output Types

Figure 14.1 above classifies sensors by their transduction mechanism. A complementary classification distinguishes sensors by their output type: **Analog output:** continuous voltage or current proportional to measurand (thermocouples, pressure transducers, 4–20 mA loop sensors). Requires ADC for digital processing. **Digital output:** digital word or pulse train directly readable by MCU (encoders, I2C/SPI sensors like BME280, DS18B20). **Self-generating:** produces energy from measurand without excitation (thermocouples, piezoelectric). **Modulating:** requires external excitation; measurand modifies the excitation (strain gauges need bridge voltage, RTDs need current source, LVDTs need AC drive).

14.4 Key Sensor Specifications

When selecting a sensor, evaluate these against your system requirements:

Range: min/max measurable values. Operating beyond range produces saturated or damaged output. **Sensitivity:** $S = \Delta V_{\text{out}}/\Delta x$ (mV/°C, mV/V per microstrain, etc.). Higher sensitivity means better SNR. **Resolution:** smallest detectable change, limited by noise or quantization. **Accuracy:** closeness to true value ($\pm\%$ FS or $\pm$ absolute units). **Linearity:** deviation from straight-line input-output relationship ($\pm\%$ FS). **Response time / Bandwidth:** how fast the sensor tracks changes; must exceed $5\times$ the signal frequency. **Repeatability:** consistency of successive readings under identical conditions. **Hysteresis:** output difference for same input

on increasing vs. decreasing sweeps. **Output impedance:** affects loading when connected to measurement circuits.

Sensor Selection Decision Framework

(1) What physical quantity? (2) What range and accuracy does the requirement specify? (3) What bandwidth (fastest change to capture)? (4) What output type does your DAQ/MCU accept? (5) What is the operating environment (temperature, moisture, vibration)? (6) Budget? Match these six criteria to candidate sensors, then prefer the simplest signal conditioning path.

14.5 Common Sensor Types for MEGN 301 Projects

14.5.1 Temperature

Thermocouples (Type K, J, T): self-generating, wide range, fast response, rugged. μV -level output requires amplification. **RTDs** (Pt100): excellent accuracy ($\pm 0.1^\circ\text{C}$), highly linear, stable. Slower, requires bridge excitation. **Thermistors** (NTC): very high sensitivity ($10\times \text{TC}$), small, but highly nonlinear and limited range. **IC sensors** (DS18B20, TMP36): DS18B20 is digital (1-Wire); TMP36 is analog ($10\text{ mV}/^\circ\text{C}$ output). Convenient, limited accuracy ($\pm 0.5^\circ\text{C}$).

14.5.2 Strain, Force, and Pressure

Foil strain gauges: $\sim 2\text{ mV/V}$ at full scale, require Wheatstone bridge and instrumentation amplifier. **Load cells:** factory-calibrated strain gauge assemblies for direct force/weight measurement. **Piezoelectric:** dynamic force only, very high bandwidth ($> 100\text{ kHz}$). **Piezoresistive pressure sensors:** strain gauges on silicon diaphragm; available in gauge, differential, and absolute configurations.

14.5.3 Motion and Position

Potentiometers: simple analog position, mechanical wear limits life. **Encoders:** incremental (pulse count for position, frequency for speed) or absolute (unique code per angle). **MEMS accelerometers** (ADXL345, MPU6050): digital I2C/SPI output for acceleration, tilt, vibration. **Ultrasonic** (HC-SR04): 2 cm–4 m non-contact range, $\pm 3\text{ mm}$.

14.5.4 Flow

Differential pressure (orifice plate, Venturi, Pitot): pressure drop proportional to v^2 . **Turbine:** rotor frequency proportional to flow. **Ultrasonic transit-time:** non-invasive clamp-on. **Electromagnetic:** Faraday's law, no moving parts, requires conductive fluid.

14.6 Actuator Principles

14.6.1 Electromagnetic Actuators

DC motors (brushed, BLDC): torque proportional to current, speed proportional to voltage minus back-EMF. Controlled via PWM + H-bridge (Chapter 12). **Stepper motors:** discrete steps (typically $1.8^\circ/\text{step}$), open-loop positioning, no encoder required. **Servo motors:** integrated

motor + feedback + controller; accept position/velocity commands. **Solenoids:** linear on/off force from energized coil; fast (<10 ms), inductive (flyback diode mandatory per Chapter 12).

14.6.2 Thermal Actuators

Resistive heaters: $P = I^2 R$, controlled via PWM through SSR or MOSFET. **Peltier devices:** semiconductor heat pump; can heat or cool depending on current direction; low efficiency.

14.6.3 Fluid Actuators

Pumps: positive-displacement (gear, peristaltic) for precise metering; centrifugal for high flow. **Valves:** solenoid for on/off; proportional valves for variable flow. **Cylinders:** hydraulic (high force, 100–350 bar) or pneumatic (lower force, 4–8 bar).

Every Actuator Needs a Driver

No motor, solenoid, or heater connects directly to a microcontroller GPIO pin. Motors need H-bridges or ESCs. Solenoids need MOSFETs with flyback diodes. Heaters need SSRs or MOSFETs with current limiting. Chapter 11 (Power Supplies, Motors & Pumps) covers selection; Chapter 12 (Power Architecture) covers driver circuits and protection.

Requirements Mapping: Sensors and Actuators

“The temperature sensor shall measure 0–200 °C with ± 1.0 °C accuracy and response time <2.0 s.” “The drive motor shall produce ≥ 0.5 N·m continuous torque at 500 RPM with $\pm 5\%$ speed regulation.” “The flow sensor shall measure 0–10 L/min with $\pm 3\%$ FS accuracy and a pulse output compatible with the ESP32 GPIO.” Every sensor and actuator specification traces to a testable system requirement.

Quick Reference: Common Sensor Types

Measurand	Sensor	Output	Range	Example Parts
Temperature	DS18B20	1-Wire digital	–55–125 °C	Waterproof probe, ± 0.5 °C
Temperature	BME280	I2C/SPI	–40–85 °C	Also humidity + pressure
Temperature	Type K TC	Analog (mV)	–200–1250 °C	Use MAX31855 amplifier
Proximity	HC-SR04	Pulse (echo)	2–400 cm	Ultrasonic; 3 mm resolution
Proximity	VL53L0X	I2C	3–200 cm	ToF laser; 1 mm resolution
Force/Weight	HX711 + load cell	Proprietary serial	0–200 kg	24-bit ADC built-in
Accel/Gyro	MPU-6050	I2C	± 16 g / $\pm 2000^\circ$ /s	6-axis IMU
Current	ACS712	Analog	$\pm 5/20/30$ A	Hall-effect, isolated
Pressure	BMP390	I2C/SPI	300–1250 hPa	Barometric; ± 0.5 hPa
Flow	YF-S201	Pulse	1–30 L/min	Hall-effect turbine

14.7 Practice Questions

1. You need to measure temperature inside a running engine (range 200–800 °C, accuracy $\pm 2^\circ\text{C}$, response $< 1\text{ s}$). Which sensor type and why? What signal conditioning?
2. Explain why a piezoelectric force sensor cannot measure the static weight of an object, even though it easily detects the impact when the object is dropped.
3. A stepper motor has 200 steps/rev driven at 400 pulses/s. What is the shaft speed in RPM?

14.7.1 Answers

1. Type K thermocouple. Range covers 200–800 °C (RTDs max at $\sim 850^\circ\text{C}$ but cost more). Type K accuracy is $\pm 2.2^\circ\text{C}$ or 0.75%. Exposed-junction bead: $\tau < 0.5\text{ s}$ in moving gas. Signal conditioning: thermocouple amplifier with cold-junction compensation (AD8495 analog or MAX31855 digital).
2. Piezoelectric sensors generate charge from stress ($Q = d \times F$). For static loads, charge is generated at application but leaks away through crystal insulation resistance and amplifier input impedance ($\tau = R \times C$, typically seconds). After 5τ , output decays to zero. Dynamic forces continuously generate new charge, maintaining output.
3. $400/200 = 2\text{ rev/s} = 120\text{ RPM}$.

Chapter 15

Mechanical Fasteners & Bearings

When to Read This Chapter

Sprint 3 (CDR): Read when specifying fasteners in your BOM and engineering drawings. The common metric fastener sizes table is your primary reference. **Sprint 4–5:** Return to the threaded inserts section when mounting bearings or components in 3D-printed parts. The bearing selection framework applies when your project includes rotating shafts.

Every mechanical system is held together by fasteners and supported by bearings. Selecting the wrong bolt grade, under-torquing a critical joint, or choosing a bearing that cannot handle the load will cause your system to fail; often catastrophically and without warning. This chapter covers the fastener and bearing knowledge you need to specify these components correctly in your BOM and engineering drawings.

15.1 Threaded Fasteners

15.1.1 Bolt and Screw Anatomy

A threaded fastener converts rotational torque into axial clamping force (preload) that holds a joint together. Key parameters: **nominal diameter** (major diameter, e.g., M6 = 6 mm), **pitch** (distance between threads; metric specifies pitch in mm; imperial uses TPI), **thread length**, and **head type** (hex, socket, flat, pan, button).

Bolts pass through clearance holes and clamp with a nut. **Screws** thread into a tapped hole. **Machine screws** use fine threads in tapped metal. **Self-tapping screws** form their own threads in plastic or sheet metal.

15.1.2 Thread Standards

Metric and Imperial Do Not Mix

M6×1.0 and 1/4-20 UNC look nearly identical (6.0 mm vs. 6.35 mm diameter). They will partially engage but strip immediately under load. Always verify with a thread gauge or package markings.

Unified (inch): UNC (Coarse) for general assembly, fewer turns, tolerant of damage. UNF (Fine) for vibration-sensitive and precision applications. **Metric (ISO):** Designated $Md \times p$ (e.g., M6×1.0). Coarse pitch is the default. Fine pitch for vibration resistance. Metric is the global standard, use for new designs unless interfacing with existing inch hardware.

15.1.3 Bolt Grades and Strength

SAE (Inch)				ISO (Metric)		
Grade	Head Mark	UTS	Use	Class	UTS (MPa)	Yield (MPa)
Grade 2	No marks	74 ksi (510 MPa)	General purpose	4.6	400	240 (4 × 6 × 10)
Grade 5	3 radial lines	120 ksi (830 MPa)	Structural (most common)	8.8	800	640 (8 × 8 × 10)
Grade 8	6 radial lines	150 ksi (1040 MPa)	High strength critical joints	10.9	1040	940
				12.9	1220	1100

Torque-Preload: $T = K \cdot F \cdot d$ $K = 0.20$ (dry) $K = 0.15$ (lubricated) $K = 0.12$ (anti-seize)
90% of torque is lost to friction; only 10% generates clamping force.

Figure 15.1: SAE and ISO bolt grade markings with strength data and torque-preload relationship.

SAE (inch): Grade 2 (no marks, 74 ksi, general purpose), Grade 5 (3 radial lines, 120 ksi, most common structural), Grade 8 (6 radial lines, 150 ksi, high strength). **ISO (metric):** Class 4.6 ($\approx$ Gr. 2), 8.8 ($\approx$ Gr. 5), 10.9 ($\approx$ Gr. 8), 12.9 (socket heads, 1220 MPa). The first digit $\times 100$ gives the UTS in MPa; the product of digits gives yield.

Specify the Grade

Never list just “M6 hex bolt” in your BOM. Always specify: M6 $\times$ 1.0 $\times$ 25, Class 8.8, zinc plated. Missing the grade means the vendor ships 4.6 at half the strength.

15.1.4 Torque and Preload

The purpose of tightening a bolt is to generate **preload** (clamping force). The torque-preload relationship:

$$T = K \cdot F \cdot d \quad (15.1)$$

where T = applied torque, K = nut factor (0.20 dry steel, 0.15 lubricated, 0.12 anti-seize), F = desired preload (typically 75% of proof load), d = nominal diameter. **90% of applied torque is lost to friction; only 10% generates clamping force.**

15.1.5 Locking Methods

Vibration loosens fasteners. Four reliable methods:

- **Nyloc nut:** nylon insert deforms around threads. One-time use per manufacturer guidance. Max 120°C.
- **Thread-locker:** Loctite 242 (blue, removable) for general use; 271 (red, permanent) for critical joints.

- **Nord-Lock washers:** wedge-locking. Positive mechanical lock independent of friction. Gold standard.
- **Safety wire:** drilled bolt heads wired in pairs. Aerospace and motorsport.

Split Lock Washers Are Ineffective

Despite ubiquity, split lock washers have been shown in NASA and Junker vibration testing to provide no meaningful locking action. They flatten at low preloads and do not prevent loosening under transverse vibration. Use nyloc nuts, thread-locker (Loctite 242 medium-strength for removable, 271 for permanent), or Nordlock wedge-lock washers instead.

15.1.6 Non-Threaded Fasteners

Rivets: permanent fasteners for sheet materials. Blind (pop) rivets install from one side. **Pins:** dowel (precision alignment), roll/spring (self-retaining), clevis (quick connect/disconnect for linkages). **Retaining rings:** external E-clips on shafts, internal in bores; inexpensive axial retention for bearings and gears. **Snap-fits:** integral plastic features; cantilever design with strain $\leq 2\%$ for ABS, $\leq 4\%$ for PP.

MEGN 301 Sourcing

McMaster-Carr ships next-day to Golden and stocks every fastener here. Bolt Depot and Albany County Fasteners for bulk. For 3D-printed snap-fits, prototype in PLA, final in PETG or nylon.

15.2 Bearings

15.2.1 What Bearings Do

Bearings support a rotating or sliding shaft while minimizing friction. Every bearing balances: **load capacity**, **speed**, **precision**, and **life**.

15.2.2 Bearing Types

Deep groove ball (6000 series): default for students. Radial + light axial, high speed, low cost. **Angular contact:** combined radial + axial; installed in pairs. **Tapered roller:** heavy combined loads (wheel bearings). **Needle roller:** compact, high radial, zero axial. **Thrust:** axial only; always pair with a radial bearing. **Sleeve/bushing:** no rolling elements, low cost, quiet. Igus iglide® polymer bushings are excellent for 3D-printed housings.

15.2.3 Bearing Selection

Five steps: (1) **Load type and magnitude;** calculate equivalent dynamic load P . (2) **Speed;** check limiting RPM. (3) **Space;** bore and OD constraints. (4) **Life;** $L_{10} = (C/P)^p \times 10^6 / (60n)$ hours. (5) **Mounting;** press fit, retaining ring, adhesive.

Bearing Sizing for Student Projects

(1) Select shaft diameter. (2) Look up matching bearing (8 mm shaft $\rightarrow$ 608-2RS, \$2). (3) Verify $C \geq 3 \times \text{max load}$. (4) Order 2RS (sealed) variant. Done.

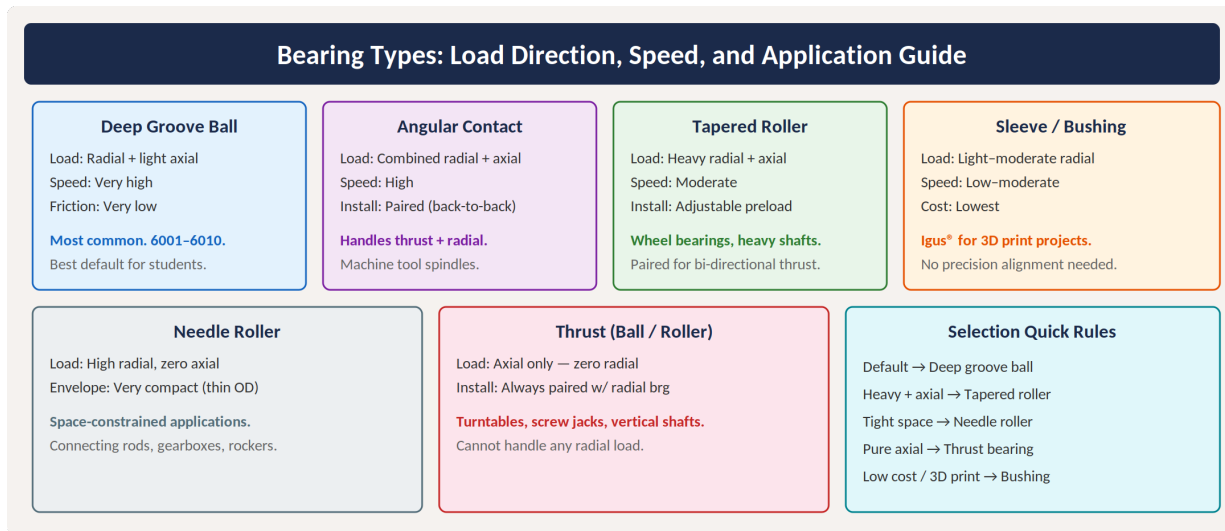


Figure 15.2: Six bearing types with load direction, speed capability, and application guidance.

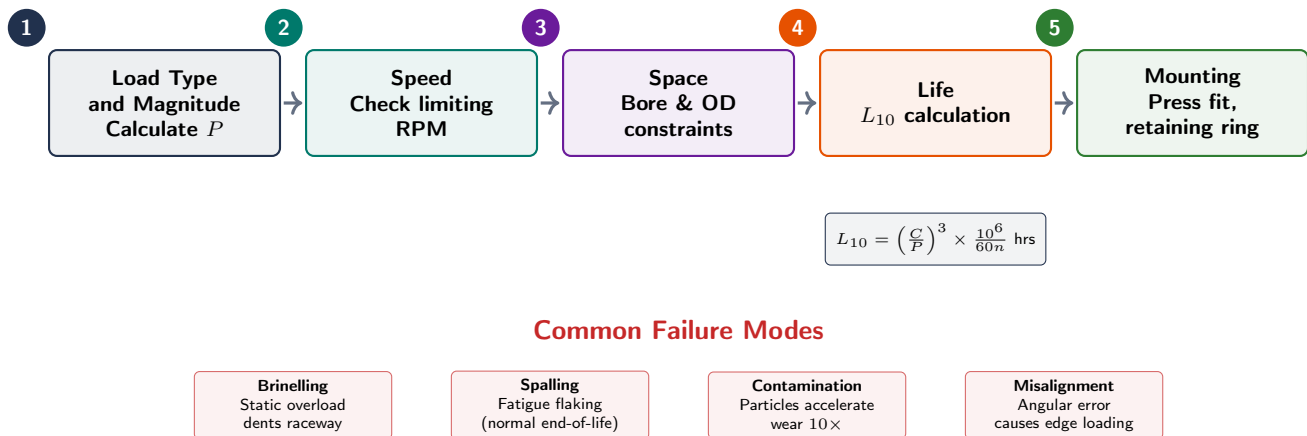


Figure 15.3: Five-step bearing selection framework with common failure modes.

15.2.4 Mounting Bearings

Press fit: bore interference fit on shaft (0.005–0.015 mm). Use an arbor press; press only on the fitted ring.

Mounting Bearings in 3D Prints

FDM tolerance ± 0.2 – 0.3 mm. For bearing pockets: design 0.1–0.2 mm undersize and ream to fit, or 0.05 mm oversize with retaining compound (Loctite 638). Always test-fit with a prototype print.

Never Hammer a Bearing

Striking a bearing transmits shock through rolling elements, creating brinelling dents. These cause vibration, noise, and premature failure. Always use an arbor press or soft-jaw vise.

15.2.5 Failure Modes

Brinelling: static overload dents raceway. **Spalling:** fatigue flaking (normal end-of-life). **Contamination:** particles accelerate wear 10 $\times$; use sealed (2RS) bearings. **Misalignment:** angular error causes edge loading.

15.2.6 Thread Engagement Table

Material	Min Engagement	Preferred	Notes
Steel	$1.0d$	$1.5d$	Standard for structural joints
Aluminum	$1.5d$	$2.0d$	Softer threads require more engagement
Cast iron	$1.5d$	$2.0d$	Brittle; threads can crack if overtorqued
Plastic (ABS)	$2.0d$	$2.5d$	Use coarse thread; consider threaded insert
Brass	$1.5d$	$2.0d$	Avoid for high-torque applications

Table 15.1: Minimum thread engagement by material.

15.2.7 Threaded Inserts for 3D-Printed Parts

Direct threading into FDM plastic is unreliable; the threads strip under moderate torque. Three better options: **Heat-set brass inserts** (pressed in with a soldering iron; strongest, most professional), **press-fit inserts** (pushed into undersized holes; adequate for light loads), and **captive nuts** (hex nut trapped in a designed pocket during printing). Heat-set inserts (McMaster #94180A series) are the recommended solution for MEGN 301 projects.

15.2.8 Bolt Selection Checklist

1. Determine joint loads (tension, shear, or combined).
2. Select thread standard (metric for new designs).
3. Select grade/class based on load with $\geq 2\times$ safety factor.
4. Size diameter from shear/tension calculations.
5. Determine length: grip length + nut + washer + 2–3 exposed threads.

6. Specify head type based on tool access and flush requirements.
7. Specify locking method based on vibration environment.
8. Calculate and specify torque value.
9. Add all details to BOM entry.

15.3 Design Rules of Thumb

1. Thread engagement: min $1.5d$ in steel, $2d$ in aluminum, $2.5d$ in plastic.
2. Bolt spacing: min $3d$ center-to-center, $2d$ from edge.
3. Inner ring press fit to shaft; outer ring sliding fit to housing (rotating-shaft case).
4. Always specify grade, material, finish, and torque in assembly documentation.

Requirements Mapping: From Fastener Specs to Shall-Statements

Mechanical joining generates system requirements:

- “Vibration environment (field deployment)” → MECH-REQ: All structural fasteners shall use positive locking (nyloc, thread-locker, or Nord-Lock). Split washers shall not be used.
- “Enclosure opens for maintenance” → MECH-REQ: Enclosure fasteners shall be captive (retained in the cover after removal) and require only a #2 Phillips driver.
- “Shaft supports 50 N radial load at 1500 RPM” → BRG-REQ: Shaft bearing shall have $L_{10} \geq 5000$ hrs at the specified load and speed. Bearing type: deep-groove ball, sealed (2RS).

15.4 Threaded Fastener Design

15.4.1 Bolt Sizing

The primary design criterion for a bolted joint is the clamping force. The bolt must provide sufficient clamping force to prevent joint separation under the maximum expected external load, plus a safety factor. The clamping force is related to the tightening torque by:

$$T = K \cdot d \cdot F_i \quad (15.2)$$

where T is the torque, K is the nut factor (approximately 0.2 for dry steel, 0.15 for lubricated), d is the nominal bolt diameter, and F_i is the initial clamping force (preload). For the joint to remain clamped under external load F_{ext} : $F_i > F_{\text{ext}}/(1 - C)$ where C is the joint stiffness ratio (typically 0.1–0.3 for metal-on-metal joints).

15.4.2 Thread Engagement

The minimum thread engagement length to develop full bolt strength: for steel bolts in steel: $L_e \geq 1.0d$ (one diameter). For steel bolts in aluminum: $L_e \geq 1.5d$ (the aluminum threads are weaker). For steel bolts in plastic: $L_e \geq 2.5d$ or use threaded inserts (Heli-Coils).

15.4.3 Thread Locking

Under vibration, threaded fasteners can loosen. Prevention methods: (1) **Nylon-insert lock nuts** (Nyloc): a nylon collar in the nut provides friction that resists loosening. Effective, low cost, one-time use per manufacturer guidance. (2) **Thread-locking adhesive** (Loctite): blue (242) for medium strength (removable with hand tools), red (271) for permanent (requires heat to remove). (3) **Split lock washers**: limited effectiveness; largely replaced by Nordlock wedge-locking washers. (4) **Safety wire**: for high-vibration aerospace applications (drilling the bolt head and wiring to an adjacent bolt).

15.5 Bearing Selection and Installation

15.5.1 Bearing Sizing

Bearings are sized by their rated life, which depends on the applied load:

$$L_{10} = \left(\frac{C}{P}\right)^3 \quad (\text{ball bearings}) \quad L_{10} = \left(\frac{C}{P}\right)^{10/3} \quad (\text{roller bearings}) \quad (15.3)$$

where L_{10} is the life in millions of revolutions (90% survival probability), C is the basic dynamic load rating (from the catalog), and P is the equivalent dynamic load. For a 608 bearing ($C = 3.45$ kN) under 100 N radial load: $L_{10} = (3450/100)^3 = 41,063$ million revolutions. At 1000 RPM: this is 683,000 hours, far exceeding any project lifetime. Bearings in MEGN 301 projects are almost never load-limited; select based on size, availability, and cost.

15.5.2 Bearing Installation

Press the bearing onto the shaft by applying force to the inner ring (not the outer ring). For housing installation: press on the outer ring. Never apply force through the rolling elements (this creates dents in the raceways, called brinelling, which causes noise and premature failure). Use an arbor press with a properly sized tube that contacts only the ring being pressed. For light press fits: use a dead-blow hammer with a bearing installation tool. Apply a thin film of oil to the shaft and housing bore before installation.

15.6 Design for Assembly (DfA)

Minimize part count: every part that can be eliminated saves assembly time, reduces potential failure points, and simplifies inventory. Challenge every part: can it be combined with an adjacent part? Can its function be achieved by a feature of another part?

Design for top-down assembly: parts should assemble from one direction (typically gravity-assisted, top-down). Avoid requiring the assembler to flip, rotate, or hold the assembly in unstable orientations.

Self-locating features: use chamfers, tapers, and guide pins to align parts during assembly. A part that can only be assembled one way (poka-yoke) eliminates assembly errors.

Standard hardware: use the minimum number of fastener types and sizes. Ideally, one M3 screw size for all internal fasteners and one M5 for external/structural fasteners. This minimizes the tool kit and eliminates wrong-fastener errors.

15.7 Fastener Material Selection

Low-carbon steel (Grade 2/SAE J429): the most common and cheapest fastener material. Tensile strength ~ 510 MPa. Adequate for most non-critical applications. Usually zinc-plated for corrosion resistance (adequate for indoor use; inadequate for outdoor or wet environments).

Medium-carbon steel (Grade 5/SAE J429): quenched and tempered. Tensile strength ~ 830 MPa. Standard for automotive and structural applications. Three radial marks on the hex head identify Grade 5.

Alloy steel (Grade 8/SAE J429): highest standard grade. Tensile strength ~ 1040 MPa. Six radial marks. Use when high clamping force is needed in a small bolt (e.g., mounting a motor that generates high vibration).

Stainless steel (A2/304 and A4/316): corrosion resistant but lower strength than Grade 5 (~ 700 MPa for A2). Non-magnetic (mostly). Use in wet environments, food equipment, and medical devices. Galling risk: stainless-on-stainless threads can cold-weld during tightening. Always use anti-seize compound (nickel-based: Permatex Nickel Anti-Seize or Loctite LB 8023) on stainless fasteners.

Nylon and other polymers: electrically insulating, non-marring, lightweight. Very low strength (~ 70 MPa). Use for PCB mounting, panel retention, and applications where electrical isolation is required.

15.8 Metric vs. Imperial Fasteners

MEGN 301 uses metric fasteners exclusively (M2, M2.5, M3, M4, M5, M6, M8). Key dimensions:

- M3: 3.0 mm diameter, 0.5 mm pitch, 2.5 mm hex socket (most common for electronics/small mechanisms)
- M4: 4.0 mm, 0.7 mm pitch, 3.0 mm hex
- M5: 5.0 mm, 0.8 mm pitch, 4.0 mm hex (structural connections)
- M6: 6.0 mm, 1.0 mm pitch, 5.0 mm hex

Never mix metric and imperial fasteners in the same assembly. They are not interchangeable: an M6 bolt (6.0×1.0 mm) will cross-thread into a 1/4-20 UNC hole (6.35×1.27 mm), destroying both threads.

15.9 Threaded Insert Technology

When bolting into soft materials (3D-printed plastic, wood, aluminum sheet), direct threading provides insufficient strength and wear resistance. Threaded inserts solve this:

Heat-set inserts (brass, knurled): heated with a soldering iron and pressed into a slightly undersized hole in thermoplastic. The plastic melts around the knurls and re-solidifies, creating a permanent, high-pullout-strength metal thread. Standard for 3D-printed enclosures. Use an insert tip on the soldering iron for consistent installation.

Heli-Coil inserts (stainless steel wire coils): installed into a tapped hole of the correct Heli-Coil drill and tap size. Create a wear-resistant thread in aluminum, magnesium, and other soft metals. Standard for aerospace and automotive repair of stripped threads.

Press-in inserts (brass, steel): pressed into a hole with an arbor press. The knurled exterior grips the surrounding material. Used in plastic injection-molded parts and machined assemblies.

15.10 Adhesive Bonding

When mechanical fasteners are impractical (thin materials, aesthetic requirements, vibration-sensitive joints), adhesives provide an alternative:

Cyanoacrylate (CA, “super glue”): fast cure (seconds), high shear strength (~ 20 MPa), brittle. Good for: bonding small plastic parts, tacking components during assembly, and strain gauge installation. Poor gap-filling; requires intimate contact.

Epoxy (two-part): slow cure (5 minutes to 24 hours depending on formulation), very high strength (~ 35 MPa shear, ~ 70 MPa tensile), excellent gap-filling, chemical resistant. Standard structural adhesive. 5-minute epoxy is convenient but weaker than 24-hour types.

Silicone (RTV): flexible, waterproof, temperature resistant (-60 to $+200$ °C). Low strength (~ 2 MPa). Use for sealing, vibration isolation, and potting electronics. Not a structural adhesive.

Thread-locking adhesive (Loctite): applied to bolt threads to prevent loosening. Blue (242): medium strength, removable with hand tools. Red (271): permanent, requires heat (~ 250 °C) to remove. Purple (222): low strength, for small screws (M2–M6) that must be removable.

Adhesive Surface Preparation

Adhesive strength depends critically on surface cleanliness. Always clean bonding surfaces with isopropanol (IPA) before applying adhesive. For metals: abrade lightly with 320-grit sandpaper, then clean with IPA. For plastics: IPA wipe only (abrasion can create micro-cracks that weaken the joint). For 3D-printed PLA: adhesion is poor on smooth surfaces; lightly sand to create mechanical interlock.

Quick Reference: Common Metric Fastener Sizes

Size	Pitch	Clearance	Tap Drill	Torque ^a	Typical Use
M2	0.4 mm	2.4 mm	1.6 mm	0.2 N·m	Electronics, PCB standoffs
M3	0.5 mm	3.4 mm	2.5 mm	0.6 N·m	Small brackets, sensor mounts
M4	0.7 mm	4.5 mm	3.3 mm	1.5 N·m	General assembly, motor mounts
M5	0.8 mm	5.5 mm	4.2 mm	3.0 N·m	Medium structural, T-slot extrusion
M6	1.0 mm	6.6 mm	5.0 mm	5.5 N·m	Heavy structural, brackets
M8	1.25 mm	9.0 mm	6.8 mm	14 N·m	High-load joints, flanges

^aClass 8.8 bolt, dry ($K = 0.20$). Lubricated: multiply by 0.75.

15.11 Practice Questions

1. Your design uses an M8×1.25 bolt in tapped aluminum. Minimum thread engagement depth? If limited to 10 mm, is this sufficient?
2. A joint experiences severe vibration. Compare nyloc, Loctite 242, and Nord-Lock. Recommend one.
3. Your BOM lists “1/4-20 bolt, qty 12” with no other info. What is wrong? Rewrite it properly.
4. Select a bearing for an 8 mm shaft at 3000 RPM with 50 N radial load. Specify designation, type, expected life.

5. A teammate suggests press-fitting bearings into PLA housings. Identify two problems and propose solutions.
6. Belt-driven stage idler pulley bearings grind after one week. Likely causes and diagnosis?
7. An M6 Class 8.8 bolt has proof load ≈ 16.3 kN. Calculate torque for 75% proof load preload ($K = 0.20$).

15.11.1 Answers

1. Min = 2×8 mm = 16 mm for aluminum. 10 mm = $1.25d$; insufficient. Use through-bolt with nut, Heli-Coil insert, or thicker boss.
2. Nord-Lock for severe vibration with frequent disassembly. Loctite 271 (red) for permanent assembly. Nyloc for general use.
3. Missing: grade, head type, length, material/finish, torque. Correct: “1/4-20 UNC $\times$ 1.00 in hex cap, SAE Grade 5, zinc, qty 12. Torque: 8.0 ft-lb with Loctite 242.”
4. 608-2RS: 8 mm bore, 22 mm OD. $C = 3.45$ kN, $P = 0.05$ kN. $L_{10} = (3.45/0.05)^3 \times 10^6 / (60 \times 3000) = 918,000$ hrs.
5. PLA: low HDT ($\sim 60^\circ\text{C}$), creeps under load $\rightarrow$ pocket enlarges. Tolerance ± 0.2 mm makes press fit unreliable. Fix: use PETG/nylon, ream pocket, or use retaining compound.
6. Contamination, misalignment, or open (unsealed) bearings. Diagnosis: spin by hand (gritty = contamination), check housing alignment, inspect seals. Replace with 2RS type.
7. $F = 0.75 \times 16,300 = 12,225$ N. $T = 0.20 \times 12,225 \times 0.006 = 14.7$ N·m.

Chapter 16

Mechanical Power Transmission

When to Read This Chapter

Sprint 2–3: Read the motor-to-load matching section when sizing your drivetrain. **Sprint 3 (CDR):** Use the transmission components quick-reference table to specify parts for your BOM. The 3D-printed gear section applies if you are prototyping custom gears.

Power transmission converts motor output (torque and speed) into the motion your system needs. Every element, belt, chain, gear, shaft, linkage, lead screw, trades speed for torque (or vice versa), and every one introduces losses. Selecting the right transmission is the bridge between your electrical power budget and the mechanical work your system must perform.

16.1 Fundamental Relationships

Three equations govern all power transmission:

$$\text{Gear ratio: } GR = N_2/N_1 = \omega_1/\omega_2 = \tau_2/\tau_1 \quad (16.1)$$

$$\text{Power: } P = \tau \times \omega \quad (16.2)$$

$$\text{Efficiency: } \eta = P_{\text{out}}/P_{\text{in}} \quad (16.3)$$

Speed down → **torque up** by the gear ratio; power constant minus losses. Multi-stage efficiency compounds: $0.95^3 = 85.7\%$.

16.2 Belt and Pulley Drives

16.2.1 V-Belts

Wedge cross-section grips by friction. Allows slip (protects downstream). 90–95% efficient. Requires tension adjustment.

16.2.2 Timing Belts (Synchronous)

Toothed belts mesh with toothed pulleys; no slip, positive drive. 97–99% efficient.

GT2 Timing Belts for Student Projects

GT2 profile (2 mm pitch, 6 mm width) is the standard for student projects, 3D printers, and hobby CNC. \$3–8/meter on Amazon. Pulleys 16–60 tooth for \$2–5. Always use flanged pulleys to prevent belt tracking.

16.2.3 Flat and Round Belts

Flat belts on crowned pulleys for long center distances. Round belts (O-ring) for low-load, compact drives.

16.3 Chain Drives

Roller chain: positive engagement, 95–98% efficient, high loads and shock tolerance. #25 (1/4 in pitch) for student projects. Minimum 17 teeth on small sprocket.

Chain Requires Lubrication

Dry chain wears at 10× the rate of lubricated chain. Apply chain lubricant (not WD-40) every 20–50 operating hours. A worn chain skips teeth; sudden and dangerous.

16.4 Gear Trains

Type	Efficiency	Characteristics	Limitations	Recommendation
Spur Gear	94–98% per mesh	Simple, cheap Straight teeth High capacity	Noisy at speed No axial load capacity	Default for student projects
Timing Belt	97–99%	No slip No lubrication Quiet	Limited torque Tension needed Belt wear	GT2 for light linear stages
Worm Gear	40–90%	High ratio (5:1–100:1) Self-locking	Low efficiency Heat generation One-way drive	Position hold without power
Planetary	95–97% per stage	Compact Coaxial High torque	Complex Expensive Backlash	Space-constrained high-ratio apps
Roller Chain	95–98%	Positive drive High torque Shock tolerant	Needs lube Noisy Heavy	#25 chain for high-load apps

Figure 16.1: Power transmission methods with efficiency, characteristics, and recommendations.

Spur: straight teeth, 94–98%/mesh, noisy at speed. **Helical**: angled teeth, quieter, generates axial thrust. **Bevel**: intersecting shafts (90°). **Worm**: high ratio single-stage (5:1–100:1), 40–90% efficient, **self-locking** when lead angle $< \sim 6^\circ$. **Planetary**: compact, coaxial, 95–97%/stage.

Gear Selection for Student Projects

ServoCity (servocity.com) sells matched spur gear sets with standardized shafts and hubs. McMaster has individual gears in any size. For 3D-printed gears: PETG or nylon, module ≥ 1.5 , add 0.1–0.2 mm backlash.

16.4.1 Gear Train Design Process

1. Calculate required output torque and speed from the load.
2. Calculate gear ratio: $GR = \omega_{motor}/\omega_{load}$.
3. Select gear type based on noise, efficiency, and axis orientation requirements.
4. If $GR > 10$: use multi-stage (two spur stages) or single-stage worm/planetary.
5. Size gears for tooth bending stress and surface contact stress (Lewis equation for bending: $\sigma = W_t/(b \cdot m \cdot Y)$).
6. For student projects: select stock gears from ServoCity or McMaster and verify that the rated torque exceeds your load with $\geq 2\times$ margin.
7. Specify mounting: shaft diameter, key or set screw, bearing supports, and housing.

16.4.2 Backlash and Precision

Backlash is the angular play between mating gear teeth (or belt-pulley engagement). It causes positioning error and control-loop instability. Standard spur gears have 0.05–0.10 mm backlash. For precision applications: use anti-backlash gears (split gear with spring preload), ball screws (inherently low backlash), or harmonic drives (zero backlash). For most MEGN 301 projects, standard backlash is acceptable.

16.5 Shafts and Couplings

16.5.1 Shaft Design

Shafts transmit torque and support rotating components. Default: 1045 steel. Size for torque with safety factor. Keyways add $\sim 25\%$ stress concentration.

16.5.2 Shaft-Hub Connections

Keyway: rectangular key transmits torque; standard sizes from ANSI B17.1. **Set screw**: clamps hub to flat; low torque, adequate for light student loads. **D-shaft**: flat on shaft engages matching hub flat; common on stepper motors.

16.5.3 Couplings

Rigid: zero misalignment tolerance. **Flexible**: jaw coupling (elastomer spider, shock absorption), beam coupling (zero backlash), Oldham (parallel offset), universal joint (large angles).

16.6 Linear Motion

16.6.1 Lead Screws

Acme-thread, 25–50% efficient, **self-locking**. T8×8 (8 mm dia, 8 mm lead) is the student default (\$3–8 with brass nut).

16.6.2 Ball Screws

Recirculating balls, 85–95% efficient, precise, **not self-locking**. More expensive.

16.6.3 Rack and Pinion

Unlimited travel, moderate precision. Force = torque / pinion pitch radius.

16.6.4 Belt-Driven Linear Stages

Timing belt on two pulleys, carriage on linear rails. Fast, low friction. GT2 + MGN12 rail for student projects.

16.7 Linkages

Four-bar: most versatile planar mechanism. Grashof condition: $S + L \leq P + Q$ for full rotation.

Slider-crank: rotary $\leftrightarrow$ linear (pistons, presses). Stroke = $2 \times$ crank radius. **Cam-follower:** programmed motion profiles from cam shape.

16.8 Selection Framework

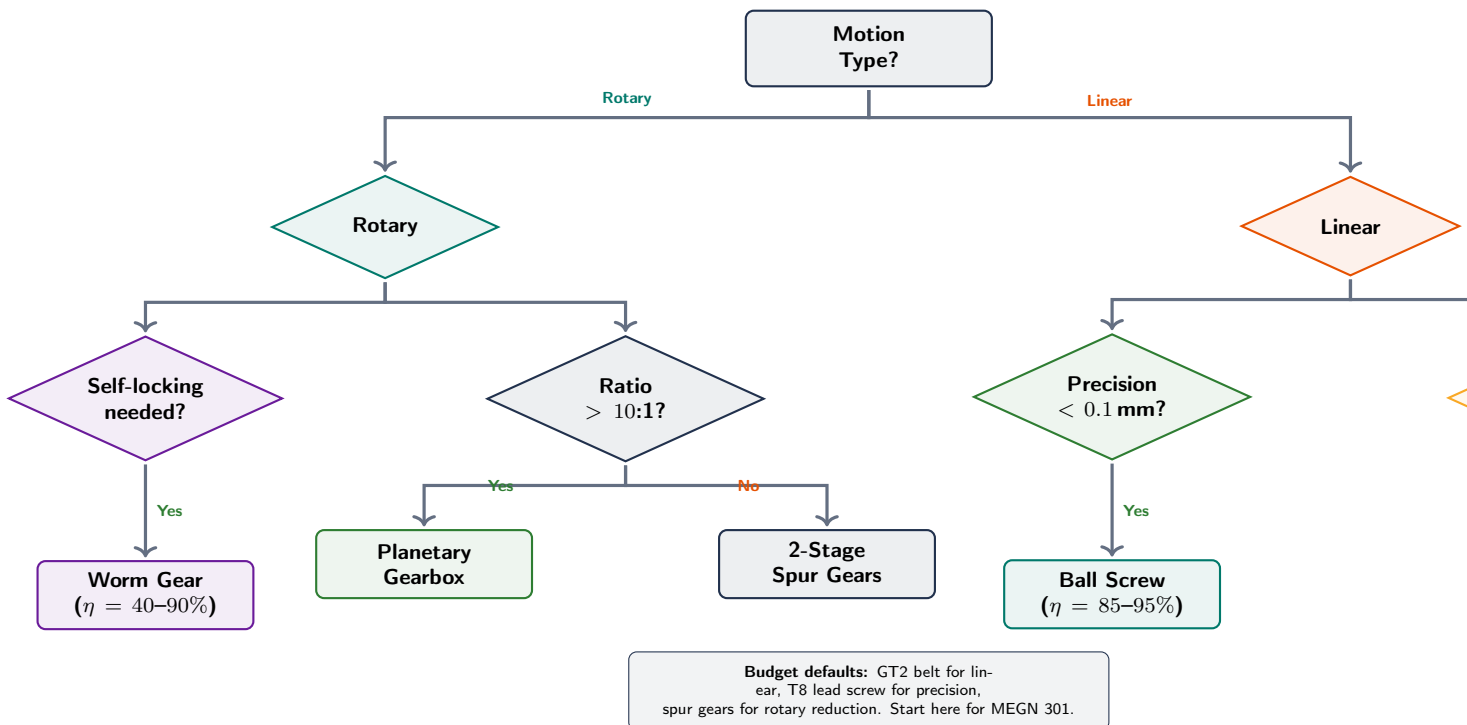


Figure 16.2: Power transmission selection: motion type, precision, self-locking, ratio, budget.

Five Key Takeaways

1. Speed down = torque up. Calculate output torque before selecting transmission.
2. GT2 belts and T8 lead screws are student defaults. Start there.
3. Efficiency compounds multiplicatively. Minimize stages.

4. Self-locking (lead screws, worms) holds position without power. Non-self-locking (ball screws, spur gears) needs a brake.
5. Every rotating joint needs a bearing. Every coupling needs defined misalignment tolerance.

Requirements Mapping: From Transmission Specs to Shall-Statements

Power transmission design generates system requirements:

- “Linear stage positions to ± 0.1 mm” $\rightarrow$ MECH-REQ: Lead screw shall have ≤ 0.05 mm backlash. Anti-backlash nut required.
- “System holds position when unpowered” $\rightarrow$ MECH-REQ: Transmission shall be self-locking (lead screw or worm gear with lead angle $< 6^\circ$).
- “Gear ratio 10:1 in minimum space” $\rightarrow$ MECH-REQ: Use planetary gearbox (≤ 40 mm diameter) with $\geq 95\%$ efficiency per stage.

Calculate the required output torque and speed *from requirements* before selecting any transmission hardware.

16.9 Power Transmission Fundamentals

Mechanical power transmission converts the motor’s output (speed and torque at the motor shaft) into the required speed and torque at the load. The fundamental relationship:

$$P = T \cdot \omega \quad (16.4)$$

where P is power (W), T is torque (N·m), and ω is angular velocity (rad/s). Power is conserved (minus friction losses), so reducing speed increases torque proportionally: $T_{\text{out}} = T_{\text{in}} \times (N_{\text{in}}/N_{\text{out}}) \times \eta$ where η is the efficiency.

16.10 Belt and Chain Drives

Timing belts (synchronous belts): positive engagement (no slip). Tooth profiles: GT2 (2 mm pitch, standard for 3D printers and small mechanisms), HTD (higher torque capacity for larger systems). Advantages: precise speed ratio, no lubrication, quiet. Sizing: select belt width for the required torque; the belt manufacturer’s catalog provides power ratings per unit width for each tooth profile.

V-belts: friction drive (slight slip under load). Simple, tolerant of misalignment, absorbs vibration. Used in industrial machinery, HVAC, and automotive accessories. Not suitable for precise positioning.

Roller chain: positive engagement, high torque capacity. Standard sizes: #25 (1/4-inch pitch), #35 (3/8-inch), #40 (1/2-inch). Requires lubrication and tensioning. Noisy at high speed. Used in bicycles, motorcycles, and industrial conveyors.

16.11 Lead Screws and Linear Motion

Converting rotary motion to linear motion:

Lead screws: a threaded rod turns in a nut, translating rotary motion to linear motion. The lead l (linear distance per revolution) determines the speed and force: $v = l \cdot n$ (speed),

$F = 2\pi T / (l \cdot \eta)$ (force). Standard trapezoidal (ACME) thread: $\eta \approx 40\%$. Ball screws (recirculating ball bearings in the nut): $\eta \approx 90\%$, but $10\times$ the cost.

Rack and pinion: a spur gear (pinion) meshes with a linear gear (rack). Linear speed $= \pi d n$ where d is the pinion pitch diameter. Used in CNC machines, steering systems, and linear actuators.

16.12 Coupling Selection

Couplings connect two shafts and transmit torque. They must accommodate misalignment between the shafts:

Rigid couplings: no misalignment tolerance. Used only when shafts are precisely aligned (lathe spindles, precision instruments). Types: sleeve, clamp, keyed.

Flexible couplings: accommodate angular, parallel, and axial misalignment. Types: jaw (elastomer insert absorbs vibration; most common in student projects), beam (single-piece helical cut; allows angular and axial flex), Oldham (sliding disk; accommodates parallel offset), and universal joint (large angular misalignment for non-parallel shafts).

Selection criteria: (1) torque rating $\geq 2\times$ the maximum torque (safety factor for shock loads), (2) bore size matches both shafts, (3) misalignment tolerance $\geq$ the expected misalignment (typically $\pm 1^\circ$ angular and ± 0.5 mm parallel for student assemblies), (4) set screw or clamp style (clamp is preferred; set screws can loosen and damage the shaft).

16.13 Efficiency and Heat Generation

Every transmission element converts some input power to heat through friction. Typical efficiencies: spur gears 95–98% per stage, timing belt 95–98%, roller chain 95–98%, V-belt 90–95%, worm gear 40–90% (depending on ratio), lead screw 30–50% (ACME), ball screw 85–95%.

For a two-stage spur gear train driving a lead screw: $\eta_{\text{total}} = 0.97 \times 0.97 \times 0.40 = 0.376$ (37.6%). A 10 W motor delivers only 3.76 W to the load; 6.24 W is dissipated as heat in the gear teeth and screw thread. This heat must be managed (lubrication, heat sinking, duty cycle limits) or the mechanism will overheat and seize.

16.14 Gear Design for 3D-Printed Prototypes

3D printing enables rapid prototyping of custom gears, but the material properties and dimensional accuracy impose limitations:

FDM (PLA, ABS, PETG): layer lines on tooth flanks increase friction and noise. Typical tolerance: ± 0.2 mm, which limits the minimum module to $m \geq 1.5$ mm (smaller teeth do not print accurately). Tooth strength is limited by the layer adhesion (Z-direction strength is 30–70% of XY strength). Print gears with teeth oriented in the XY plane for maximum strength. Expect 50–100 hours of operation before significant wear; suitable for proof-of-concept but not production.

SLA (resin): much finer detail (tolerance ± 0.05 mm), enabling module ≥ 0.5 mm. Smoother tooth surfaces reduce friction. However, standard SLA resins are brittle; use tough or engineering resins for functional gears. UV degradation limits outdoor use.

Design tips: (1) increase backlash to $1.5\times$ the standard value to accommodate print tolerance, (2) add a 0.5 mm fillet at the tooth root to reduce stress concentration (which FDM's layer structure amplifies), (3) print a test gear pair and verify meshing before printing the full mechanism.

16.15 Motor-to-Load Matching

The motor must be selected to match the load's torque and speed requirements through the transmission. The design procedure:

Step 1: determine the load requirements at the output shaft. What torque (T_{load}) and speed (n_{load}) does the application need?

Step 2: select a gear ratio. $i = n_{\text{motor}}/n_{\text{load}}$. The motor's rated speed is typically 3000–12,000 RPM; the load speed is often 10–500 RPM.

Step 3: calculate the motor torque requirement. $T_{\text{motor}} = T_{\text{load}}/(i \times \eta)$ where η is the transmission efficiency.

Step 4: select a motor whose rated torque exceeds T_{motor} with a safety factor of ≥ 1.5 . Also verify that the motor's rated speed matches $n_{\text{load}} \times i$.

Step 5: verify the power. $P_{\text{motor}} = T_{\text{motor}} \times \omega_{\text{motor}} \geq T_{\text{load}} \times \omega_{\text{load}}/\eta$.

GS GreenShield: Motor Selection for a Pump Drive

The pump requires $T_{\text{load}} = 0.3 \text{ N}\cdot\text{m}$ at $n_{\text{load}} = 300 \text{ RPM}$. Using a 2-stage spur gear reduction with $i = 10$ and $\eta = 0.94$: $T_{\text{motor}} = 0.3/(10 \times 0.94) = 0.032 \text{ N}\cdot\text{m}$ at $n_{\text{motor}} = 3000 \text{ RPM}$. Power: $P = 0.3 \times 300 \times 2\pi/60 = 9.4 \text{ W}$. Select a motor rated for $\geq 0.048 \text{ N}\cdot\text{m}$ ($1.5\times$ safety factor) at 3000 RPM, with $\geq 15 \text{ W}$ power rating. A typical 12 V, 380-size brushed DC motor meets these requirements.

16.16 Linear Actuator Design

For applications requiring linear force (clamping, pressing, positioning), convert rotary motor output to linear motion:

Lead screw actuator: motor $\rightarrow$ coupling $\rightarrow$ lead screw $\rightarrow$ nut. Linear speed $v = l \times n$ (lead $\times$ RPM). Force $F = 2\pi T_{\text{motor}} \times i/l \times \eta$. Self-locking when the lead angle $< \arctan(\mu)$ (typically $< 5^\circ$), meaning the load cannot back-drive the motor. This is useful for holding position without power.

Belt-driven linear stage: motor $\rightarrow$ timing pulley $\rightarrow$ belt $\rightarrow$ carriage. Linear speed $v = \pi d n$ (pulley diameter $\times$ RPM). Not self-locking; the carriage moves freely when the motor is unpowered (use a brake or worm gear if holding is needed). Very fast: typical belt-driven CNC axes achieve 200–500 mm/s.

Rack and pinion: motor $\rightarrow$ pinion gear $\rightarrow$ rack. Linear speed $v = \pi d n$. Compact, high force, bidirectional. Common in CNC routers and industrial automation.

16.17 Vibration and Noise in Power Transmission

Gear noise arises from tooth meshing: each tooth engagement creates a brief impulse that excites structural resonances. Noise is proportional to speed, load, and gear quality. Mitigation: (1) use helical gears instead of spur (gradual engagement reduces impact), (2) increase gear quality grade (AGMA 8 or higher), (3) use plastic or composite gears for light loads (inherent damping reduces noise), (4) enclose the gear train in a housing with vibration-damping mounts.

Belt drives are inherently quieter than gears (flexible belt absorbs vibration). Chain drives are the noisiest (metal-on-metal impact at each link engagement). For noise-sensitive applications

(laboratory instruments, medical devices, consumer products): use timing belts or direct drive (no transmission, oversized motor at the load speed).

Requirements Mapping: Power Transmission

From technical specs to “shall” statements:

A timing belt drive using a GT2 belt with a 20-tooth pulley ($d = 12.73 \text{ mm}$) at 3000 RPM produces linear speed $v = \pi \times 0.01273 \times 3000/60 = 2.0 \text{ m/s}$. This becomes: “SYS-SPD-001: The linear stage shall achieve a travel speed of $2.0 \pm 0.1 \text{ m/s}$. *Verification: Test: measure travel time over 500 mm using a stopwatch ($N = 10$ trials).*”

A two-stage gear train ($i = 20$, $\eta = 0.94$) with a motor rated at $0.05 \text{ N}\cdot\text{m}$ delivers $0.05 \times 20 \times 0.94 = 0.94 \text{ N}\cdot\text{m}$ at the output. This becomes: “SYS-TRQ-001: The output shaft shall deliver $\geq 0.9 \text{ N}\cdot\text{m}$ at 150 RPM. *Verification: Test: measure stall torque with a spring scale at a known radius.*”

Quick Reference: Common Transmission Components for Student Projects

Component	Spec	Source	Notes
GT2 belt (6 mm)	2 mm pitch	Amazon, ServoCity	Standard for light linear stages
GT2 pulley (20T)	5 mm bore	Amazon, ServoCity	$d_{\text{pitch}} = 12.73 \text{ mm}$
T8×8 lead screw	8 mm lead	Amazon	26.7 mm/s at 200 RPM
T8×2 lead screw	2 mm lead	Amazon, McMaster	6.67 mm/s at 200 RPM; finer control
608 bearing (8 mm ID)	ABEC-1	McMaster, Amazon	Skate bearing; ubiquitous, cheap
6001 bearing (12 mm ID)	ABEC-1	McMaster	Common for NEMA 17 shafts
Oldham coupling	5–8 mm bore	ServoCity	Misalignment-tolerant shaft coupling
Flexible jaw coupling	5–8 mm bore	McMaster	Dampens vibration; easy install

16.18 Practice Questions

1. A NEMA 17 stepper produces $0.4 \text{ N}\cdot\text{m}$ at 200 RPM. You need $2.0 \text{ N}\cdot\text{m}$ at 40 RPM. What gear ratio is required? What efficiency must the transmission achieve?
2. Compare GT2 timing belt vs. roller chain for a 3:1 reduction on a small conveyor. Which do you choose and why?
3. Your linear stage uses a T8×8 lead screw driven by a stepper at 200 RPM. Calculate the linear speed in mm/s. If the stage must move 200 mm, how long does the motion take?
4. A worm gear set has a 30:1 ratio and 60% efficiency. Input power is 50 W. What is the output power? Is this set self-locking?
5. Your project needs a mechanism that converts continuous motor rotation into a back-and-forth linear stroke of 50 mm. Propose two different mechanism solutions and compare them.
6. A 4-bar linkage has link lengths 40, 100, 70, and 80 mm. Check the Grashof condition. Which link(s) can make a full revolution?

7. Your belt-driven linear stage stalls under load. The stepper motor skips steps but does not overheat. Diagnose the root cause and propose fixes.

16.18.1 Answers

1. $GR = 2.0/0.4 = 5 : 1$ (or $200/40 = 5 : 1$). Check: $P_{in} = 0.4 \times (200 \times 2\pi/60) = 8.38 \text{ W}$. $P_{out} = 2.0 \times (40 \times 2\pi/60) = 8.38 \text{ W}$ for 100% efficiency. With real losses, the transmission must be $\geq 100\%$ to deliver full torque; meaning the motor is marginally sized. Derate: need $\geq 85\%$ efficient transmission (GT2 belt: 97%, spur gear: 94% work; worm at 60% does not).
2. GT2 belt: no lubrication, clean, quiet, 97% efficient, \$10 total. Chain: requires lube, messy, noisier, 96% efficient, \$15. For a small indoor conveyor, GT2 wins on cleanliness, cost, and simplicity. Chain is better if the conveyor sees high shock loads or outdoor contamination.
3. Linear speed = lead $\times$ RPM / 60 = $8 \text{ mm} \times 200/60 = 26.7 \text{ mm/s}$. Time = $200/26.7 = 7.5 \text{ s}$.
4. $P_{out} = 50 \times 0.60 = 30 \text{ W}$. At 30:1 ratio, the lead angle is very low ($\ll 6^\circ$), so yes; self-locking.
5. (a) Slider-crank: crank radius = 25 mm, continuous rotation $\rightarrow$ 50 mm stroke. Simple, constant stroke. (b) Belt-driven stage with limit switches: motor reverses at each end. Variable stroke and speed via firmware. Slider-crank is simpler mechanically; belt stage is more flexible.
6. $S = 40, L = 100, P = 70, Q = 80$. $S + L = 140$. $P + Q = 150$. $140 \leq 150$. Grashof satisfied. The shortest link (40 mm) can make full revolution. If grounded, it is the crank.
7. Motor torque is sufficient (no overheating) but the belt/pulley system cannot transmit the load. Causes: belt tension too low (belt slips on pulley teeth), pulley set screw loose (pulley spins on shaft), or load exceeds belt rating. Fixes: increase belt tension, verify set screws, check that belt width and tooth count match the load.

Chapter 17

Designing for Injection Molding

When to Read This Chapter

Sprint 3 (CDR): Read the six DFM rules even if you are 3D printing your prototype. Good DFM practice (uniform walls, draft angles, radii) also produces better 3D prints and demonstrates design maturity at CDR. **Advanced/Entrepreneurial:** The tooling economics and mold classification sections are relevant if your team is considering a path from prototype to production.

17.1 Injection Molding Fundamentals

17.1.1 Process Overview

Injection molding produces plastic parts by forcing molten thermoplastic into a precision steel mold under high pressure. The machine has two units: the **injection unit** (barrel, screw, heater bands, nozzle) melts pellets and injects them, and the **clamping unit** (platens, tie bars, hydraulic or toggle mechanism) holds the mold closed. Machines are rated by clamp force (tons) and shot size (oz or g).

The mold has two halves: the **cavity** (A-side, stationary, forms the outside/cosmetic surface) and the **core** (B-side, moving, forms the inside surface). The **parting line** is where the halves meet. Inside the mold, a **runner system** distributes melt from the nozzle to the gate, **cooling channels** extract heat, and **ejector pins** push the part out.

17.1.2 The Four-Phase Cycle

1. **Clamping:** Mold halves close under force. Required clamp force:

$$F_{\text{clamp}} = P_{\text{cavity}} \times A_{\text{projected}} \quad (17.1)$$

Typical cavity pressures: 2–5 tons/in². A part with 10 in² projected area needs 20–50 tons of clamp force.

2. **Injection:** The screw pushes molten plastic into the cavity at 10,000–30,000 psi. Fill time: 0.5–5 seconds. A **pack-and-hold** phase follows to compensate for volumetric shrinkage as the plastic cools. Without adequate packing, parts have sink marks and are undersized. Melt temperatures range from 200 °C (PE) to 350 °C (PC, nylon).
3. **Cooling:** Plastic solidifies in the mold. This is the **longest phase**: 50–80% of total cycle time. Cooling water flowing through channels in the mold steel extracts heat. **Uniform cooling = uniform shrinkage = minimal warpage**. Mold temperatures: 40–120 °C depending

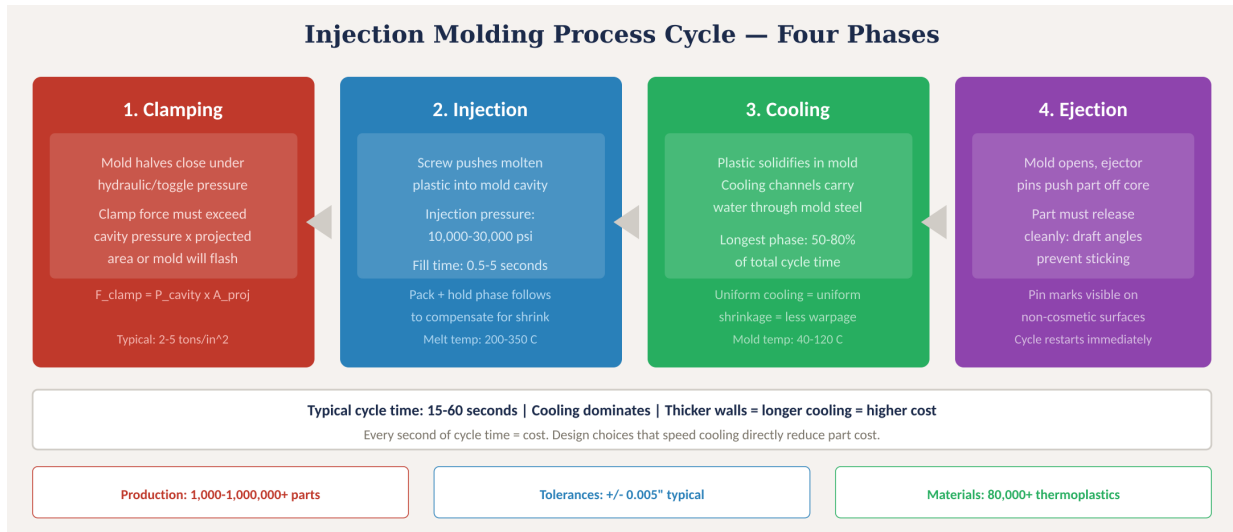


Figure 17.1: Four-phase injection molding cycle with process parameters, timing, and production capabilities.

on material. Cooling time scales with the *square* of wall thickness; doubling wall thickness quadruples cooling time.

4. **Ejection:** Mold opens, ejector pins push part off the core. Part must have **draft angles** so it releases cleanly. Pin marks are visible on non-cosmetic (B-side) surfaces. The cycle restarts immediately.

Cycle Time = Cost

Typical total cycle: 15–60 seconds. Every additional second of cycle time increases part cost. Wall thickness is the primary driver; every millimeter of additional wall can add 5–15 seconds of cooling per cycle. Design the thinnest wall that meets structural and flow requirements.

17.1.3 Material Selection

Selection criteria: Match the *cheapest* material that satisfies all of the following; mechanical loads (tensile, impact, flex), thermal environment (HDT, continuous use temperature, UL94 flammability), chemical exposure (solvents, UV, oils), aesthetics (gloss, texture, transparency, paintability), and processing (MFI, shrinkage, moisture sensitivity). Glass-fiber reinforcement increases stiffness 2–3× but reduces impact and abrades the mold.

Shrinkage Matters

When hot plastic cools it contracts. The mold cavity is cut *larger* than the finished part by the shrinkage factor. High-shrinkage materials (PP at 1–2.5%) require more generous tolerances than low-shrinkage materials (ABS at 0.5–0.7%). Crystalline polymers (PP, nylon, POM) shrink more than amorphous ones (ABS, PC, PS).

Material	Trade Names	Key Properties	Shrink	Common Uses
ABS	Cycolac	Good impact, easy to mold/paint, 80–100 °C HDT (varies by grade and test load, ASTM D648)	0.5–0.7%	Enclosures, LEGO
PC	Lexan	High impact, optical clarity, 130 °C, UV-sensitive	0.5–0.7%	Lenses, phone cases
PP	Pro-fax	Chemical resistant, living hinge, low cost	1.0–2.5%	Containers, caps
Nylon (PA)	Zytel	High strength/stiffness, absorbs moisture	0.8–1.5%	Gears, connectors
POM	Delrin	Low friction, dimensional stability, fatigue	1.8–2.5%	Gears, zippers
PC/ABS	Bayblend	PC toughness + ABS processability	0.5–0.7%	Laptops, power tools
PBT	Valox	Good electrical, chemical resistant	1.5–2.0%	Electrical connectors
TPE/TPU	Santoprene	Flexible, rubber-like, overmoldable	0.5–2.0%	Grips, seals

Table 17.1: Common injection molding thermoplastics.

17.2 Design for Manufacturing (DFM) Rules

17.2.1 Rule 1: Uniform Wall Thickness

The single most important design parameter. Wall thickness drives cycle time (cooling $\propto t^2$), material usage, shrinkage uniformity, and defect susceptibility.

Recommended ranges: ABS: 1.5–3.5 mm. PC: 1.5–3.5 mm. PP: 0.8–3.0 mm. Nylon: 1.0–3.0 mm. POM: 1.0–3.0 mm.

Uniformity: Variation <10% of nominal. Where transitions are unavoidable, use gradual tapers at a 3:1 length-to-thickness-change ratio. Abrupt steps cause differential shrinkage, sink marks, and stress concentrations.

Coring out: Replace solid thick sections with thin uniform shells reinforced by ribs. A ribbed 2 mm wall is stiffer than a solid 4 mm wall, cools in one-quarter the time, and uses half the material.

17.2.2 Rule 2: Draft Angles

Draft is the slight taper on walls parallel to the mold opening direction that allows the part to release.

- Minimum: 0.5° per side (tight tolerance applications).
- Standard: 1°–2° per side (recommended).
- Textured surfaces: add 1° per 0.001 in (0.025 mm) of texture depth.
- Core side often needs more draft than cavity side (part shrinks onto core).

Zero-Draft = Stuck Parts

Zero-draft vertical walls are the #1 student DFM error. Without draft, the part grips the core, ejector pins push through the wall, and the mold wears rapidly. Apply draft to *all* surfaces parallel to the pull direction; including internal ribs, bosses, and walls.

17.2.3 Rule 3: Ribs and Bosses

Ribs add stiffness without mass; bosses provide screw/insert mounting points.

Six Critical DFM Rules for Injection Molding

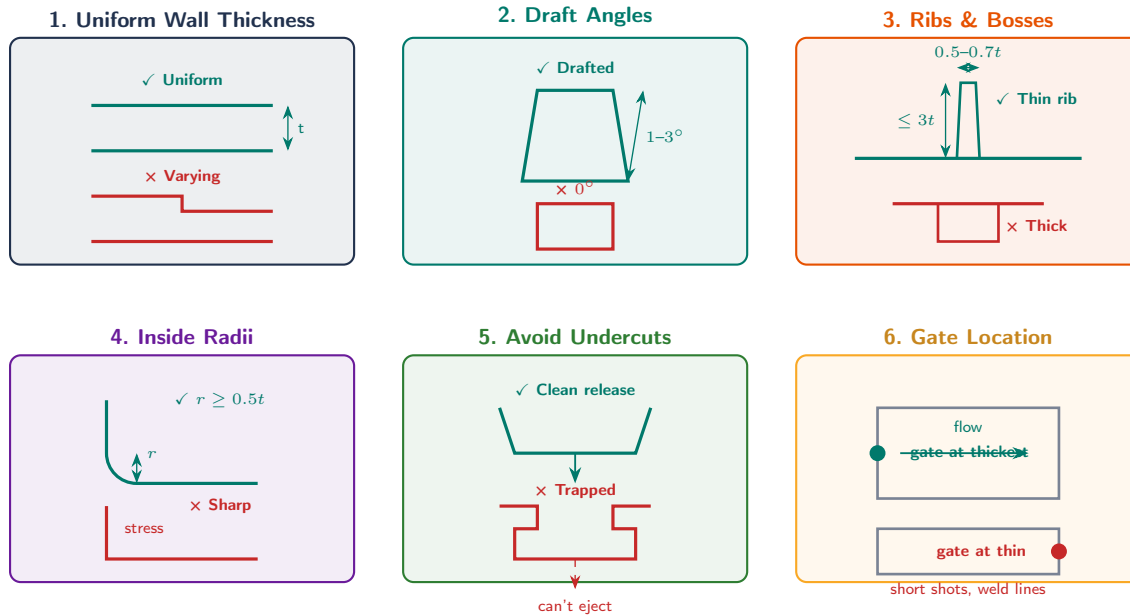


Figure 17.2: Six critical DFM geometry rules: wall thickness, draft angles, ribs/bosses, radii, undercuts, and gate location.

Rib Proportions

- Thickness: 50–70% of adjoining wall (thicker → sink marks on opposite surface).
- Height: max $3 \times$ wall thickness (taller → fill and ejection problems).
- Draft: min 0.5° per side (1° preferred).
- Spacing: min $2 \times$ wall thickness between ribs.
- Base radius: $0.25\text{--}0.5 \times$ wall thickness.
- Multiple thin ribs outperform one thick rib.

Boss Proportions

- OD: $2.0\text{--}2.5 \times$ screw/insert hole ID.
- Wall: $0.6 \times$ nominal wall thickness.
- Freestanding bosses: connect to nearest wall with a rib.
- Height: max $2.5 \times$ OD.
- For heat-set brass inserts: hole ID per insert manufacturer specification.

17.2.4 Rule 4: Radii and Fillets

Sharp inside corners create stress concentrations (up to $3 \times$ at zero radius), impede melt flow, and cause sink marks.

- Inside radius: min $0.5 \times$ wall ($1 \times$ preferred).
- Outside radius: inside radius + wall thickness.
- This maintains uniform wall thickness through the corner.
- Apply radii to *every* inside corner without exception.

17.2.5 Rule 5: Undercuts

Any feature preventing straight-pull mold separation; side holes, snap hooks, internal threads, louvers. Each undercut requires a side action (cam, lifter, or collapsing core), adding \$5,000–\$25,000 to tooling cost.

Always redesign first: Can a pass-through hole replace a blind side hole? Can a snap-fit be oriented along the pull direction? Can an internal thread become a heat-set insert installed after molding?

17.2.6 Rule 6: Gate Location

The gate is where plastic enters the cavity.

- **Place at the thickest section** so melt flows from thick to thin (ensures adequate packing).
- **Never gate on a cosmetic surface;** the gate vestige is always visible.
- Gate types: edge gate (simple, visible mark), sub gate (auto-trimmed), pin gate (small mark, 3-plate mold), hot runner (no runner waste, highest tooling cost).
- Flow simulation (Moldflow) predicts fill pattern and weld line locations.

Quick Reference: Six DFM Rules at a Glance

#	Rule	Key Values	Consequence of Violation
1	Uniform wall	ABS/PC: 1.5–3.5 mm; variation <10%	Sink marks, warpage
2	Draft angles	$\geq 1^\circ$ /side; $+1^\circ$ per 0.001" texture	Part stuck in mold
3	Ribs & bosses	Rib = 50–70% of wall; height $\leq 3 \times$ wall	Sink marks, voids
4	Radii	Inside $\geq 0.5 \times$ wall; outside = inside + wall	Stress cracks
5	Undercuts	Avoid or use side-actions/collapsing cores	Mold cost doubles
6	Gate location	Thickest section; flow toward thin; balanced	Short shots, weld lines

17.3 Common Defects

Every defect has a root cause in part geometry, material, or process. Good design eliminates most defects before the mold is ever cut.

Defect	What You See	Root Cause	Design Fix
Sink marks	Dimples opposite thick sections	Thick areas cool slowly, interior shrinks	Uniform walls; core out; rib = $0.6 \times$ wall
Warpage	Part bows/twists after ejection	Non-uniform cooling $\rightarrow$ differential shrinkage	Uniform walls; symmetric cooling; balanced gate
Short shot	Incomplete part	Inadequate venting, thin walls, low pressure	Vent at end of fill; increase thin walls
Flash	Thin film at parting line	Low clamp force, worn mold, high pressure	Minimize projected area; mold maintenance
Weld lines	Visible line where fronts meet	Flow splits around holes/cores	Relocate gate; increase melt temperature
Jetting	Squiggly pattern near gate	Melt shoots across cavity before wall contact	Redirect gate so melt hits a wall immediately

Table 17.2: Common defects, root causes, and design fixes.

Common Injection Molding Defects — Root Causes and Design Fixes			
Defect	What You See	Root Cause	Design Fix
Sink Marks	Dimples/depressions on surface opposite thick sections	Thick sections cool slower; interior shrinks, pulling surface inward	Uniform wall thickness; core out thick areas; rib thickness = 0.6x wall
Warpage	Part bows, twists, or distorts after ejection	Non-uniform cooling causes differential shrinkage	Uniform walls; symmetric cooling; balanced gate location
Short Shot	Part incomplete — cavity not fully filled	Insufficient material, inadequate venting, or walls too thin to fill	Increase wall thickness in thin areas; add vents at end of fill; relocate gate
Flash	Thin film of plastic at parting line or around inserts	Clamp force too low; worn mold surfaces; excessive injection pressure	Minimize projected area; check parting line placement; mold maint.
Weld / Knit Lines	Visible line where two flow fronts meet (cosmetic + weak point)	Flow splits around holes/cores and rejoins on far side	Move gate to push weld line to non-critical area; increase wall temp
Jetting	Snake-like squiggly pattern on part surface near gate	Melt shoots across cavity before contacting a wall to form a front	Redirect gate so melt hits a wall immediately (tab gate or fan gate)

Figure 17.3: Six common injection molding defects with visual description, root cause, and design fix.

Weld Line Strength

Weld lines retain only 50–90% of base material strength. If a weld line falls on a load-bearing feature, relocate the gate or add ribs to reinforce. Flow simulation is the primary tool for predicting and relocating weld lines before cutting the mold.

17.3.1 Worked DFM Example: Sensor Enclosure

Requirements: outdoor IP65 enclosure for GreenShield controller, ABS material, holds PCB (60×40 mm) and two cable glands.

Design decisions:

- **Wall thickness:** 2.0 mm uniform (ABS range: 1.5–3.5 mm). Adequate for IP65 with gasket groove.
- **Draft:** 2° on all external walls. C-1 matte finish; no additional texture draft needed.
- **Ribs:** four internal ribs at 1.2 mm thick (0.6 × 2.0), 6.0 mm tall (3 × 2.0), connecting the PCB mounting bosses to the walls. Base radius 0.5 mm.
- **Bosses:** four PCB mounting bosses, OD = 6.0 mm (2.5 × 2.4 mm screw), wall = 1.2 mm, height = 10 mm. Each connected to nearest wall by a rib.
- **Radii:** all inside corners 1.0 mm (0.5 × 2.0 mm wall). Outside corners 3.0 mm (inside + wall).
- **Sealing:** 1.5 mm wide × 1.0 mm deep O-ring groove around the lid perimeter. Silicone O-ring (cord dia. 1.5 mm, compressed to 75% = 1.0 mm groove depth).
- **Gate:** edge gate on the bottom face (non-cosmetic), at the thickest section near a rib intersection.
- **Parting line:** at the lid/base junction (natural split).

17.4 Tooling and Economics

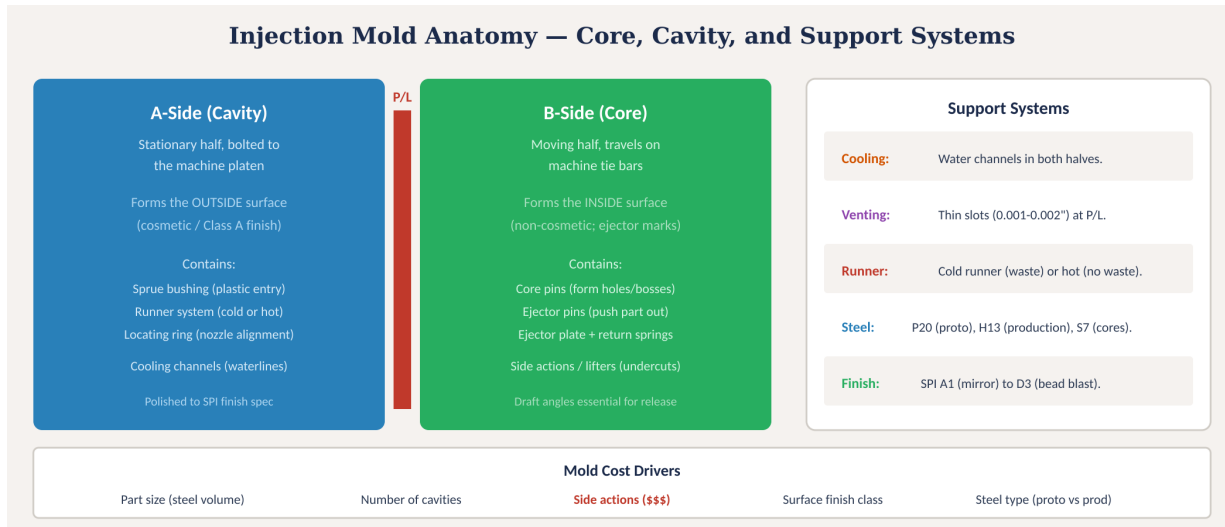


Figure 17.4: Mold anatomy: A-side (cavity), B-side (core), parting line, support systems, and cost drivers.

17.4.1 Mold Construction

The **A-side** (cavity) forms the cosmetic outside surface. It contains the sprue bushing, runner system, and locating ring. Surfaces are polished to SPI specifications (A-1 mirror to D-3 bead blast). The **B-side** (core) forms the inside surface and contains core pins, ejector pins, ejector plates, and side actions. The parting line between the halves leaves a witness mark on the part.

Support systems: cooling channels (water lines in both halves; conformal cooling can reduce cycle time 20–40%), venting (thin slots 0.001–0.002 in at parting line), runner system (cold runner produces waste; hot runner eliminates waste but costs more), and ejection system (pins, sleeves, blades, stripper plates).

17.4.2 Mold Classifications (SPI)

Class	Steel	Life (cycles)	Cost Range	Use
105 (Prototype)	Aluminum / soft steel	<500	\$5K–\$15K	Design validation, pilot runs
104 (Low Prod.)	P20 pre-hardened	<100K	\$15K–\$50K	Low volume, bridge tooling
103 (Med. Prod.)	H13 hardened	<500K	\$30K–\$80K	Medium-volume production
101 (High Prod.)	Premium hardened	1M+	\$50K–\$200K+	High-volume, long-life production

Table 17.3: SPI mold classifications.

Rapid Tooling: The Bridge Between 3D Printing and Production

For post-graduation startups or senior design continuations, **Class 105 prototype molds** (aluminum, \$5K–15K) offer a middle path. Services like Protolabs and Xometry CNC-machine aluminum molds in 10–15 business days for short runs (50–500 parts). This lets you validate DFM, test real injection-molded material properties, and produce enough units for beta testing without committing to a \$50K production mold. Design your MEGN 301 parts with full DFM rules even when 3D printing; if the design is DFM-compliant, the transition to rapid tooling requires only mold design, not part redesign.

$$C_{\text{part}} = C_{\text{material}} + C_{\text{cycle}} + C_{\text{tooling, amortized}} \quad (17.2)$$

where $C_{\text{material}} = \text{weight} \times \text{resin cost}$, $C_{\text{cycle}} = \text{cycle time} \times \text{machine rate}/3600$, and $C_{\text{tooling}} = \text{mold cost}/\text{expected lifetime parts}$.

Example (small ABS enclosure, 30 g): Material: \$0.06. Cycle (25 s at \$60/hr): \$0.42. Tooling (\$25K amortized over 50K parts): \$0.50. **Total: \$0.98/part**. At 500K parts, tooling drops to \$0.05/part, total \$0.53/part.

Injection molding breaks even versus 3D printing at approximately 100–500 parts depending on part size and complexity. Below that, 3D printing is cheaper; above it, injection molding wins dramatically.

17.5 Warpage

Warpage is dimensional distortion that occurs after ejection. It is caused by differential shrinkage: when different regions of the part cool at different rates or have different molecular orientations, they shrink by different amounts, creating internal stresses that bow or twist the part.

17.5.1 Causes of Warpage

Differential cooling: one side of the wall cools faster (e.g., mold steel temperature differs between A-side and B-side). The slow-cooling side shrinks more after ejection, bowing the part toward it. **Fiber orientation:** glass-filled materials shrink less in the flow direction than across it; 0.3% vs. 0.8% creates a 0.5% differential. **Gate location:** determines flow pattern and therefore orientation pattern. A poorly placed gate creates asymmetric flow that maximizes differential shrinkage.

17.5.2 Designing to Minimize Warpage

- Maintain uniform wall thickness ($\pm 10\%$).
- Equalize cooling on both sides (symmetric cooling channels).
- Use gating that produces symmetric flow.
- For glass-filled materials, gate at center to produce radial (symmetric) fiber orientation.
- Add ribs to resist bowing (ribs on the concave side of expected warp).

17.6 Surface Finish

17.6.1 SPI (PIA) Finish Standards

Surface finish is specified using SPI (Society of the Plastics Industry, now PIA) standards:

Grade	Finish	Ra (μm)	Method and Application
A-1	Mirror	0.012	Diamond buff. Optical lenses, light pipes.
A-2	High gloss	0.025	Diamond buff. Cosmetic consumer products.
B-1	Semi-gloss	0.05	600 grit paper. General cosmetic surfaces.
C-1	Matte	0.35	600 stone. Hides sink marks, fingerprints.
D-1	Textured	1.0	Dry blast. Grips, non-slip, industrial.
D-3	Rough texture	3.2	Coarse blast. Heavily textured panels.

Table 17.4: SPI/PIA mold surface finish standards.

Higher polish = higher mold cost. C-1 matte is the sweet spot for most student and industrial enclosures; hides minor defects while maintaining a professional appearance.

17.6.2 Textured Surfaces and Draft

Texture creates an effective undercut. Add 1° of additional draft per 0.001 in (0.025 mm) of texture depth. A D-1 texture ($1.0 \mu\text{m}$ Ra, ~ 0.001 in depth) requires 1° extra draft beyond the baseline. Failing to add texture draft causes drag marks during ejection.

17.7 Insert Molding and Overmolding

17.7.1 Insert Molding

A pre-formed component (metal insert, PCB, bushing) is placed in the mold before injection. Plastic flows around and bonds to the insert. Use when you need: threaded metal inserts (stronger than threading into plastic), electrical contacts molded into a housing, or reinforcing metal structures embedded in plastic.

Design rules: provide ≥ 1.5 mm of plastic around the insert, add knurling or grooves for mechanical interlock, preheat metal inserts to reduce thermal shock stress, and ensure the insert is fixtured precisely.

17.7.2 Overmolding

A second material is molded over a previously molded substrate. Two-shot or two-step process. Common application: soft TPE grip over rigid ABS body (power tools, toothbrushes, medical devices). The two materials must be chemically compatible for adhesion. Common pairs: ABS/TPE, PC/TPE, nylon/TPE.

17.8 Living Hinges and Snap-Fit Clips

17.8.1 Living Hinges

A living hinge is a thin, flexible web of plastic that connects two rigid sections, allowing them to flex repeatedly without breaking. **Polypropylene (PP)** is the only commonly injection-molded material suitable for living hinges; its molecular structure allows millions of flex cycles without fatigue. Design rules: hinge thickness 0.25–0.50 mm, web length 1.0–1.5 mm, flex immediately after molding (orients molecules across the hinge, increasing life by $10\times$), and gate so flow is perpendicular to the hinge line.

Material Matters

ABS, PC, and nylon all crack at a living hinge within a few cycles. **Only PP** (and some PE grades) can survive millions of flexes. If your design needs a living hinge, the part must be PP; no exceptions.

17.8.2 Snap-Fit Clips

Cantilever snaps are the most common: a beam deflects during assembly, then locks behind a lip. Design for the allowable strain of your material: 2% for ABS, 4% for PP, 1.5% for PC. Maximum deflection: $y = \epsilon L^2 / (1.5t)$ where ϵ = allowable strain, L = beam length, t = beam thickness. Include a lead-in angle (30–45°) for easy assembly and a steeper retaining angle (80–90°) for secure locking. Snap-fits eliminate screw hardware; a significant DFA (Design for Assembly) advantage.

17.9 Artifact Handoff: What This Chapter Supports

Artifact	Description	Forward Reference
DFM Review	Checklist verification for each molded part	CDR deliverable
Material Selection	Resin choice with shrinkage and property data	BOM, tolerance analysis
Tooling Estimate	Mold class, cost, and timeline	Budget, schedule

17.10 DFM Checklist

Review every item before sending your CAD model to a molder.

1. Wall thickness uniform ($\pm 10\%$)? Within recommended range for material?
2. Draft angles applied to ALL surfaces parallel to pull direction? Texture draft added?
3. Ribs at 50–70% of wall? Height $\leq 3 \times$ wall? Base radii present?
4. Bosses at $2 \times$ hole ID? Connected to walls with ribs? Boss wall $\leq 0.6 \times$ nominal?
5. All inside corners radiused (min $0.5 \times$ wall)? Outside radius = inside + wall?
6. Undercuts identified? Side actions justified or redesigned out?
7. Gate location at thickest section? Not on cosmetic surface?
8. Parting line location acceptable on finished part?
9. Ejector pin locations on non-cosmetic surfaces? Adequate draft for release?
10. Shrinkage accounted for in critical dimensions? Tolerances achievable (± 0.005 in typical)?
11. Material selected? Properties verified against operating conditions?
12. Moldflow or fill analysis run? Weld lines and air traps identified?

Fix Problems in CAD, Not in Steel

Fixing a design problem in CAD takes minutes. Finding the same problem after the mold is cut costs \$5,000–\$20,000 in mold modifications and 2–4 weeks of delay. Run this checklist at your CDR before committing to tooling.

Requirements Mapping: From DFM Rules to Shall-Statements

Injection molding constraints generate system requirements:

- “Part must survive 1 m drop onto concrete” → MAT-REQ: Material shall have Izod impact ≥ 200 J/m (ABS or PC/ABS).
- “Outdoor enclosure, 5+ year UV exposure” → MAT-REQ: Material shall be UV-stabilized or painted.
- “Volume 50K units/year” → TOOL-REQ: Mold shall be SPI Class 104 (P20 steel, ≤ 100 K cycle life).
- “Assembly time < 30 s” → DFA-REQ: Enclosure shall use snap-fit closure (no fasteners). Cantilever strain $\leq 2\%$ for ABS.

17.11 Practice Questions

17.11.1 Conceptual Questions

1. Why does cooling time scale with the *square* of wall thickness rather than linearly? How does this affect part cost?
2. A part has a 4 mm thick boss connected to a 2 mm wall. Predict the defect you will see on the surface opposite the boss. What is the design fix?
3. Explain why a part shrinks onto the core side of the mold during cooling. How does this affect draft angle requirements for the core versus cavity sides?
4. Your part has a snap hook that creates an undercut on one side. The molder quotes \$18,000 for the side action. Propose two redesign alternatives that might eliminate the side action.
5. Compare ABS and polypropylene for a consumer electronics enclosure that must accept paint and withstand a 1-meter drop test. Which do you recommend and why?
6. Why must the gate be placed at the thickest section? What happens if you gate at a thin section instead?
7. A textured surface (SPI D-2, depth ≈ 0.002 in) on a 3-inch-deep wall currently has 1° of draft. Is this sufficient? If not, what is the correct draft angle?

17.11.2 Application / Scenario Questions

8. You are designing a two-piece snap-fit enclosure for an IoT sensor that will be installed outdoors. The enclosure must be IP65, withstand -20 to $+60^\circ\text{C}$, resist UV for 5+ years, and cost under \$3/part at 10,000 units. Select the material, specify wall thickness, describe the snap-fit design, and estimate whether injection molding is economically justified.

9. Your teammate's CAD model for a motor mount bracket has: 5 mm walls, zero draft on all vertical surfaces, sharp inside corners, and a boss with $OD = 1.5 \times \text{hole ID}$. List every DFM violation, predict the defects that will result from each, and specify the corrected dimensions.
10. **Cost analysis:** A part weighs 45 g in PC/ABS (\$0.003/g). Cycle time is 30 seconds on a machine at \$75/hr. The mold costs \$35,000. Calculate the total part cost at 5,000, 50,000, and 500,000 units. At what volume does injection molding become cheaper than SLA 3D printing at \$8.50/part?

17.11.3 Answers

1. Cooling is a transient heat-conduction problem governed by the Fourier equation. The characteristic time for heat to diffuse through a slab of thickness t scales as t^2/α where α is thermal diffusivity. Doubling wall thickness from 2 mm to 4 mm quadruples cooling time; potentially adding 20–40 seconds per cycle. At a machine rate of \$60/hr, each additional second costs approximately \$0.017/part. Over 100,000 parts, a 20-second cycle-time penalty costs \$17,000.
2. A sink mark (dimple/depression) on the surface opposite the boss. The 4 mm boss is twice the 2 mm wall, creating a thick section that cools more slowly. The interior shrinks and pulls the surface inward. Fix: reduce boss wall to $0.6 \times 2 = 1.2$ mm ($OD = \text{hole ID} + 2 \times 1.2$ mm). Connect the boss to the nearest wall with a thin rib.
3. As plastic cools it contracts volumetrically. The cavity half forms the outside surface, and the part shrinks *away* from it, naturally releasing. The core half forms the inside surface, and the part shrinks *onto* it, gripping tighter as it cools. This means the core side typically needs more draft (1.5° – 2°) than the cavity side (0.5° – 1°) to overcome the shrink-grip.
4. (a) Reorient the snap hook to flex along the mold-open direction; if the hook can deflect during ejection, no side action is needed (the hook “snaps” past the mold surface). (b) Replace the snap hook with a separate clip that installs post-molding via a push-fit or ultrasonic weld, eliminating the undercut entirely.
5. ABS. Both have adequate impact strength, but ABS accepts paint directly (smooth surface, good adhesion) while PP requires flame treatment or specialty primers due to its low surface energy. ABS also has a higher HDT ($\sim 95^\circ\text{C}$ versus $\sim 60^\circ\text{C}$ for unfilled PP), providing more thermal margin. ABS shrinkage (0.5–0.7%) is lower and more predictable than PP (1.0–2.5%), enabling tighter dimensional control for snap-fits and mating features.
6. Gating at the thickest section ensures that the section that shrinks the most remains connected to the runner for packing pressure the longest. If you gate at a thin section, the thin area freezes first and seals off (“gates off”), cutting the thick section off from packing pressure. The result is severe sink marks and voids in the thick section because the material cannot be replenished as it shrinks.
7. No. The texture-depth rule requires adding 1° per 0.001 in of depth. At 0.002 in depth, add 2° beyond the standard 1° for a total of 3° per side. Without adequate draft, the textured surface acts like sandpaper during ejection, leaving drag marks that ruin the finish and gouge the mold steel.
8. Material: ASA (acrylic-styrene-acrylonitrile), UV-stable variant of ABS, good impact from -20 to $+80^\circ\text{C}$, IP65-rated with proper gasket design, paintable if needed. Wall thickness:

2.0 mm nominal (meets structural requirements, enables IP65 with perimeter gasket groove). Snap-fit: cantilever snaps along the mold-open axis (no side action), 2% max strain for ASA, designed with 1.5° draft. Economics: at 10,000 units with a Class 104 mold (\$25K), tooling amortization is \$2.50/part. Material (20 g × \$0.003): \$0.06. Cycle (20 s at \$60/hr): \$0.33. Total: ~\$2.89/part, under the \$3 target. Injection molding is justified.

9. Violations and fixes: (a) 5 mm walls are too thick for any common plastic; excessive cooling time, sink marks, voids. Fix: core out to 2 mm uniform walls, add ribs for stiffness. (b) Zero draft on all vertical surfaces; parts will stick, ejector damage, mold wear. Fix: add minimum 1° per side to all surfaces parallel to pull. (c) Sharp inside corners; stress concentrations (3× multiplier), impeded flow. Fix: add $R = 0.5 \times 2 = 1$ mm fillet to all inside corners; outside radius = $1 + 2 = 3$ mm. (d) Boss OD at 1.5× hole ID is too thin-walled, creating a thick ring that sinks. Fix: increase OD to 2.0× hole ID. Predicted defects: sink marks everywhere (thick walls + oversized boss), warpage (non-uniform cooling from variable thickness), ejection damage (no draft), cracking at corners under load (no radii).
10. Material: $45 \times 0.003 = \$0.135/\text{part}$. Cycle: $30/3600 \times 75 = \$0.625/\text{part}$. These are constant.
At 5,000 parts: tooling = $35000/5000 = \$7.00/\text{part}$. Total: \$7.76/part.
At 50,000 parts: tooling = $35000/50000 = \$0.70/\text{part}$. Total: \$1.46/part.
At 500,000 parts: tooling = $35000/500000 = \$0.07/\text{part}$. Total: \$0.83/part.
Break-even vs. SLA at \$8.50/part: solve $0.135 + 0.625 + 35000/n = 8.50$. $35000/n = 7.74$. $n = 4,522$ parts. Above ~4,500 parts, injection molding is cheaper.

17.11.4 Application / Scenario Questions

7. You are designing an outdoor sensor enclosure in ABS. The enclosure must be IP65 (dust-tight, water-jet resistant). Specify: wall thickness, draft angles, sealing method, and surface finish. Justify each choice.
8. A client asks you to injection mold a part with a living hinge. They want to use ABS because it matches the rest of their product line. What do you tell them?
9. Your injection-molded part has sink marks opposite the boss locations. The bosses are 3.0 mm wall in a part with 2.0 mm nominal wall. What is wrong, and how do you fix it?

17.11.5 Answers to Application Questions

7. Wall: 2.0 mm nominal (meets structural, enables IP65 with perimeter gasket groove). Draft: 2° standard on all external walls; add 1° per 0.001 in of texture depth if textured. Sealing: silicone O-ring in a groove on the lid mating surface (not adhesive or gasket tape). Surface: C-1 matte (hides minor defects, professional appearance, reduces mold cost vs. polish). Internal surfaces: D-1 textured (hides ejector pin marks).
8. ABS cannot survive living hinge cycles; it will crack within 1–5 flexes. Only PP (and some PE) can form living hinges that last millions of cycles. Options: (a) redesign the part in PP; (b) use a separate PP living hinge piece bonded/overmolded to the ABS body; (c) replace the living hinge with a mechanical hinge (pin + knuckle) that can be molded in ABS.

9. Boss wall = 3.0 mm vs. 2.0 mm nominal = 150% of wall thickness. This thick section cools slowly, creating a void that pulls the surface inward (sink mark). Fix: reduce boss wall to $0.6 \times 2.0 = 1.2$ mm (60% of nominal). If the boss needs more material for screw engagement, connect it to the nearest wall with a rib (rib at 60% wall = 1.2 mm).

Chapter 18

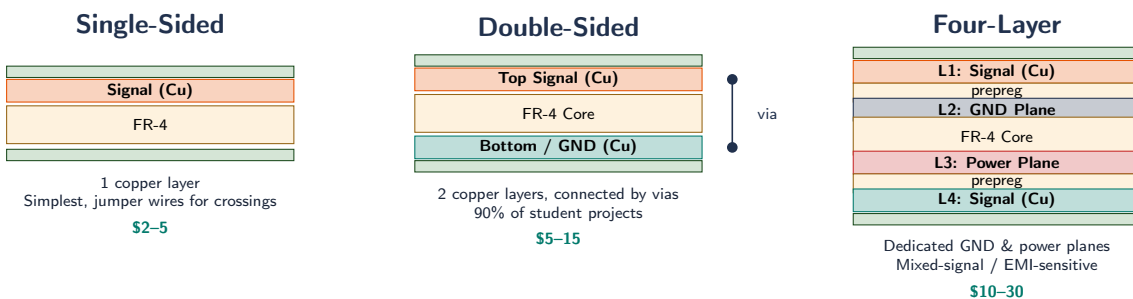
Printed Circuit Board Design Fundamentals

When to Read This Chapter

Sprint 3: Read the full chapter if your project includes a custom PCB. The design must be frozen at CDR to allow 4–6 weeks for fabrication and bring-up. **Sprint 4:** Use the bring-up protocol and visual inspection checklist when your boards arrive. **Dev board projects:** Skim the grounding, decoupling, and trace width sections; these principles apply to breadboard wiring too. **Related:** Circuit theory in Chapter 13 (p. 138); power architecture in Chapter 12 (p. 125); ESP32 pin assignments in Chapter 19 (p. 199); debugging in Chapter 20 (p. 210).

18.1 PCB Structure and Materials

A printed circuit board is a layered sandwich of insulating substrate (FR-4 fiberglass-epoxy) with patterned copper conductors. Copper traces, pads, and planes are created by chemically etching unwanted copper from a continuous sheet. A **solder mask** (the green coating) protects copper from oxidation and shorts. A **silkscreen** layer prints component outlines and reference designators.



Trace Width Quick Reference (1 oz Cu, external, derated)

10 mil = 0.3 A	20 mil = 0.7 A	50 mil = 1.5 A	100 mil = 3.0 A
----------------	----------------	----------------	-----------------

Conservative/derated values. Use IPC-2221 calculator for exact.

Figure 18.1: PCB layer stack-ups from single-sided to four-layer with key specifications and trace width current reference.

18.1.1 Layer Stack-Ups

Single-sided: one copper layer. Simplest, cheapest (\$2–5). Jumper wires for crossings. Very simple circuits only. **Double-sided:** copper on both sides connected by vias (plated through-holes). Standard for 90% of student projects (\$5–15). **Four-layer:** adds dedicated ground and power planes between signal layers. Use for mixed-signal or EMI-sensitive designs (\$10–30).

18.1.2 Key Specifications

FR-4 substrate, 1.6 mm standard thickness, 1 oz (35 μ m) copper, 6 mil minimum trace/space, 0.3 mm minimum drill. Surface finishes: HASL (cheapest, good for THT), ENIG (flat pads, good for SMD), OSP (organic, short shelf life).

Trace Width for Current

A 10-mil trace on 1-oz external copper safely carries only 0.3 A. These are conservative/derated values with safety margin for quick reference: 10 mil = 0.3 A, 20 mil = 0.7 A, 50 mil = 1.5 A, 100 mil = 3.0 A, 200 mil = 6.0 A. Use the IPC-2221 calculator for exact values. Signal traces can use minimums; power traces must be sized for current. See the detailed table in Section 18.10.1 for IPC-2221B values at 10 °C rise.

18.1.3 Component Packages

Component Packages — Through-Hole vs Surface Mount	
Through-Hole (THT)	Surface Mount (SMD/SMT)
Leads pass through holes in the PCB and are soldered on the opposite side.	Components soldered directly to pads on the board surface. No holes required.
DIP Dual in-line package. ICs, sockets. 0.1" pin spacing. Easy to hand-solder. TO-220 Power transistors, voltage regulators. Tab for heatsink mounting. Axial/Radial Resistors, capacitors, diodes. Axial = flat; Radial = standing. Headers Pin headers, screw terminals. 0.1" (2.54mm) pitch standard.	0805/0603 Standard resistors/caps. 0805 hand-solderable; 0603 needs skill. SOIC/TSSOP IC packages. Gull-wing leads. Hand-solderable with flux + iron. QFP/QFN Fine-pitch ICs. QFN has ground pad underneath — needs reflow/hot air. BGA Ball grid array. Cannot hand-solder. Requires reflow oven + X-ray.
Advantages: easy to solder, inspect, rework, prototype Disadvantages: large, expensive, slower assembly, fewer options RECOMMENDED for MEGN 301 prototypes (unless component only available in SMD)	Advantages: small, cheap, fast pick-and-place, better HF performance Disadvantages: harder to hand-solder, inspect, rework; needs magnification Use for: production, space-constrained, SMD-only parts (JLCPCB assembly can place SMD for \$8 setup + per-part)
Mixed designs are common: THT for connectors, headers, and power components; SMD for passives and ICs where THT is unavailable.	

Figure 18.2: Through-hole vs surface-mount packages with common types and soldering guidance.

Through-hole (THT): leads pass through holes. DIP, TO-220, axial/radial, pin headers. Easy to hand-solder, inspect, and rework. Recommended for MEGN 301 prototypes. **Surface-mount (SMD/SMT):** soldered to pads on the board surface. 0805/0603 passives, SOIC/TSSOP ICs, QFP/QFN fine-pitch. Smaller, cheaper, but harder to hand-solder. Use SMD only when THT is unavailable or space is critical. Mixed designs are common.

18.2 Design Workflow

Six stages: (1) **Schematic capture**; draw circuit with component symbols and net connections. (2) **Component selection**; map symbols to physical footprints, verify availability, create BOM. (3) **Board layout**; place components, route traces, pour copper fills. (4) **Design review**; peer review, check clearances, verify footprints, review Gerbers. (5) **Fabrication + assembly**; generate Gerbers, order boards, solder. (6) **Bring-up + test**; inspect, smoke test, functional test, debug.

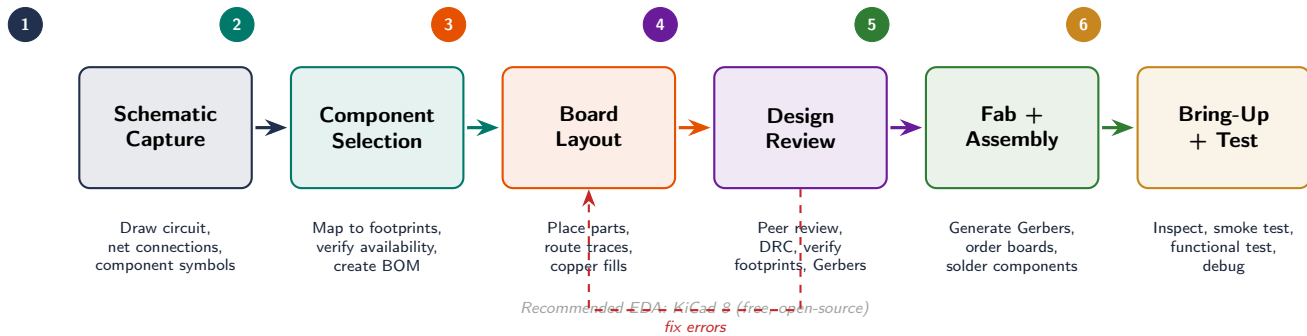


Figure 18.3: Six-stage PCB design workflow from schematic capture through board bring-up.

Recommended EDA tool: **KiCad 8** (free, open-source, fully capable). Alternatives: Altium (industry standard, student license), EasyEDA (browser-based, JLCPCB integrated).

18.3 Schematic Capture Best Practices

18.3.1 Schematic Organization

A professional schematic is a *communication document*, not just a wiring diagram. Follow these rules from IPC-2612:

1. **One functional block per sheet:** power supply on Sheet 1, MCU + peripherals on Sheet 2, motor drivers on Sheet 3, connectors on Sheet 4. Title each sheet.
2. **Signal flow left to right:** inputs on the left, outputs on the right. Power flows top (VCC) to bottom (GND).
3. **Net labels over long wires:** replace wires that cross the entire sheet with named net labels (e.g., MOTOR_PWM, SDA, VBAT). This eliminates spaghetti routing.
4. **Power flags:** explicitly show where power enters the schematic. Add decoupling caps adjacent to every IC symbol, not on a separate “capacitor page.”
5. **Consistent reference designators:** R for resistors, C for capacitors, U for ICs, J for connectors, D for diodes, Q for transistors, L for inductors.

18.3.2 Component Selection and Footprint Verification

The most expensive PCB mistake is a wrong footprint. For every component:

1. Download the **manufacturer’s datasheet** (not a distributor summary page).
2. Find the **recommended land pattern** (footprint) in the datasheet’s “Package Information” section.
3. Verify the KiCad library footprint dimensions match the datasheet **exactly**: pad size, pad pitch, courtyard.
4. If no library footprint exists, create one from the datasheet using KiCad’s footprint editor. Measure twice.
5. **Cross-check:** print the footprint at 1:1 scale and physically place the component on the printout.

The \$50 Footprint Mistake

A team ordered 5 PCBs (\$25 total) with a QFN-32 footprint that had the wrong pad pitch (0.5 mm vs. 0.65 mm). All 5 boards were scrap. The fix required a new board spin: another \$25 + 2 weeks of lost schedule. Total cost of one unchecked footprint: \$50 + 2 weeks. The 1:1 printout would have caught this in 30 seconds.

18.3.3 Design Rule Configuration

Configure KiCad's DRC rules before starting layout:

Rule	Minimum	Recommended	Notes
Trace width (signal)	6 mil	10 mil	Wider = easier to solder rework
Trace width (power)	Per IPC calc	50–200 mil	Size for actual current
Trace spacing	6 mil	8 mil	Tight spacing increases fab cost
Via drill	0.3 mm	0.4 mm	Smaller vias cost more
Via annular ring	0.15 mm	0.25 mm	Larger = more reliable
Clearance to edge	0.3 mm	0.5 mm	Prevents copper exposure at board edge

Table 18.1: Recommended DRC rules for MEGN 301 PCBs (JLCPCB/PCBWay compatible).

18.4 Critical Layout Rules

PCB Layout Rules — Trace Sizing, Clearance, and Grounding

Trace Width vs Current (1 oz Cu, 10 C rise)		
0.5 A	10 mil (0.25 mm)	Signal default
1.0 A	20 mil (0.50 mm)	LEDs, sensors
2.0 A	50 mil (1.27 mm)	Motor drivers
3.0 A	80 mil (2.03 mm)	Power rails
5.0 A	150 mil (3.81 mm)	High-current bus
10 A	400 mil or polygon pour	Use plane, not trace

Use a trace width calculator for exact values.

Critical Layout Rules

Decoupling caps: 100 nF at EVERY IC power pin.
Place as close as physically possible.

Ground plane: Pour copper on bottom layer.
Unbroken under sensitive signals.

Signal clearance: 6 mil min trace-to-trace (low V).
20 mil min for mains voltage.

Component spacing: Min 0.5 mm between SMD pads.

Thermal relief: Spokes, not solid fill to planes.

Edge clearance: 0.5 mm copper, 2 mm components.

Place components first (80%), then route. Good placement makes routing easy; bad placement makes it impossible.

Run DRC before exporting. Fix EVERY violation.

Figure 18.4: PCB layout rules: decoupling, ground planes, component placement, and routing priorities.

18.4.1 Decoupling Capacitors

The critical layout rules are summarized in Figure 18.4. Place a 100 nF ceramic capacitor within 5 mm of every IC power pin, connected with the shortest possible traces. Add one 10 μ F bulk capacitor per power rail. Via directly to ground plane for the ground side. This is not optional; without decoupling, MCUs reset and ADCs produce garbage data.

18.4.2 Ground Plane

On a 2-layer board, fill the bottom layer with a ground copper pour. This provides a low-impedance return path for all signals and dramatically reduces noise. Connect the pour to GND with abundant vias. Avoid splitting the ground plane; a split forces return current to detour.

18.4.3 Component Placement and Routing

Place components before routing. Group by function. Keep high-current paths short. Route power first, then sensitive signals, then non-critical digital. Avoid 90-degree trace corners (use 45-degree). Keep analog and digital sections physically separated. Add thermal relief pads on ground-plane connections for hand soldering.

The #1 Rule

PRINT YOUR BOARD AT 1:1 SCALE AND PLACE REAL COMPONENTS ON THE PRINTOUT before ordering. This single step catches 80% of footprint errors; the most expensive PCB mistake (wrong footprint = scrap board).

18.5 Fabrication, Assembly, and Bring-Up

18.5.1 Fabrication Files

Gerber files (RS-274X format): one per layer. **Drill files** (Excellon): hole coordinates and diameters. **BOM + pick-and-place**: for assembly services. Always verify in a standalone Gerber viewer before ordering.

18.5.2 Fab Houses

JLCPCB: 5 boards for \$2–7 + shipping (5–7 day fab). PCBWay: similar. OSH Park: \$5/sq-in, US-based. Order early; total timeline is 4–6 weeks from design start to working board.

18.5.3 Board Bring-Up Protocol

Step 1 Is Mandatory: No Exceptions

Do not apply power to any board until Step 1 is complete and signed off. Skipping visual inspection is the fastest way to destroy a \$40 ESP32 module. A single solder bridge on a 0.5 mm-pitch IC creates a dead short that damages the component the instant power is applied.

1. **Visual inspection under $\geq 10\times$ magnification** (loupe, USB microscope, or stereo microscope). This is a **mandatory gate**; a teammate must co-sign the inspection before power is applied. Inspect *every* solder joint. Specific defects to look for:
 - **Solder bridges**: look between adjacent pads on fine-pitch ICs (QFP, SOIC). A bridge shorts two pins.
 - **Cold joints**: dull, grainy, or blobby solder that did not wet the pad. These create intermittent connections.
 - **Tombstoned passives**: 0603/0805 components standing on one end (one pad soldered, one lifted).
 - **Missing components**: compare the board to the BOM and the schematic. Every refdes on the silkscreen should have a component.
 - **Wrong orientation**: check polarity marks on diodes, tantalum caps, ICs (pin 1 dot), and electrolytic caps.
2. **Continuity check**: multimeter between VCC and GND on every power rail. Any reading $< 10\ \Omega$ indicates a short. **Do not proceed until all shorts are resolved.**
3. **Current-limited power**: set bench supply to expected voltage, current limit to 50 mA. Apply power. If the supply immediately current-limits, there is a short. Remove power and re-inspect.
4. **Voltage verification**: measure every power rail with a multimeter. Each rail should read within $\pm 5\%$ of its nominal voltage.
5. **Smoke test**: increase current limit to the expected operating current. Monitor for 60 seconds. Watch for: components getting hot (touch-test cautiously), smell of burning flux or plastic, any visible smoke.
6. **Functional test**: load firmware, verify heartbeat LED, test each subsystem individually before connecting external hardware.

18.5.4 Design Review Checklist

Before ordering PCBs, verify:

1. DRC (Design Rule Check) passes with zero errors and zero unreviewed warnings.
2. All footprints verified against component datasheets (not just library defaults).
3. Board outline correct with mounting holes matching the enclosure.
4. All connectors accessible and oriented correctly for cable routing.
5. Silkscreen readable (refdes near components, polarity marks on diodes/caps/ICs).
6. 1:1 printout with real components placed; **this catches 80% of errors**.
7. Gerber files opened in a standalone viewer and visually inspected layer by layer.
8. BOM cross-checked against schematic (every schematic component in BOM and vice versa).
9. Peer review by at least one teammate who did not do the layout.

18.5.5 Development Board Integration

When using a development board (ESP32 DevKit, Arduino Nano) instead of a custom PCB:

- Mount the dev board on a custom carrier PCB or perfboard with proper headers.
- Use screw terminals or JST connectors (not breadboard jumper wires) for all permanent connections.
- Add status LEDs and test points to the carrier board.

- Document the wiring diagram with connector pin assignments.
- Secure all wires with strain relief (zip ties, hot glue at connector exit points).

A well-wired development board that meets all requirements is superior to a custom PCB that delays the project or has errors.

18.6 EMI, Noise, and Signal Integrity

18.6.1 The Return Current Problem

Every signal current has a return current that flows through the ground plane (or ground trace) back to the source. The return current follows the path of *least impedance*; at high frequencies, this is the path directly beneath the signal trace. If the ground plane is split or missing under a signal trace, the return current detours, creating a large loop antenna that radiates EMI and picks up noise.

Rule: never split the ground plane. If you must route a trace on the ground layer, stitch the ground pour back together with vias on both sides of the crossing.

18.6.2 Analog-Digital Separation

Mixed-signal boards (analog sensors + digital MCU + motor drivers) are the norm in MEGN 301 projects. Three strategies:

1. **Physical separation:** place analog components (sensors, ADC, analog filters) on one side of the board, digital components (MCU, motor drivers) on the other.
2. **Separate power rails:** use separate voltage regulators for analog and digital sections. At minimum, use an RC filter ($10\ \Omega + 10\ \mu\text{F}$) between the digital supply and the analog supply.
3. **Single-point ground connection:** if using split ground pours (advanced), connect them at exactly one point near the ADC. For most student projects, a single continuous ground pour with careful component placement is sufficient and safer.

18.6.3 Motor Noise Mitigation on PCB

Motor PWM switching generates noise on every power and ground trace. Mitigation:

- Route motor power traces on a separate layer or board region from signal traces.
- Place ceramic capacitors ($100\ \text{nF}$) directly across motor driver power pins.
- Add a bulk electrolytic ($100\text{--}470\ \mu\text{F}$) at the motor power input.
- Use a TVS diode on the motor power bus for transient suppression.
- Never share ground vias between motor driver and MCU sections.

18.7 Thermal Management on PCBs

Components that dissipate significant power require thermal design:

- **Thermal pads:** many voltage regulators and motor drivers have an exposed pad on the bottom of the package. This pad **must** be soldered to a copper area on the PCB and connected to internal ground planes with thermal vias (array of $0.3\ \text{mm}$ vias on $1.0\ \text{mm}$ pitch under the pad).
- **Copper pour as heatsink:** for SOT-223 or D-PAK regulators, extend the ground pour under and around the package. Each square centimeter of 1-oz copper on the outer layer provides approximately $60\ ^\circ\text{C}/\text{W}$ of thermal resistance to ambient.

- **Thermal relief:** for hand-soldering, use thermal relief pads (spoke connections) on ground-connected pads. Without them, the ground plane wicks heat away faster than the soldering iron can supply it, resulting in cold joints.

Quick Thermal Check

For any linear regulator: $P = (V_{in} - V_{out}) \times I_{load}$. If $P > 0.5\text{ W}$, you need thermal analysis. If $P > 1\text{ W}$, you probably need a switching regulator instead.

Requirements Mapping: From PCB Specs to Shall-Statements

PCB design decisions directly generate system requirements:

- “Board fits in 60×40 mm enclosure” → SYS-REQ: The PCB shall have maximum dimensions of 58×38 mm with mounting holes at [coordinates].
- “Motor driver draws 3 A” → PCB-REQ: Motor power traces shall be ≥ 150 mil wide on 1-oz copper.
- “ADC reads 0–3.3 V sensors” → PCB-REQ: Analog input traces shall be routed ≥ 10 mm from motor power traces with a continuous ground plane beneath.

Every technical specification in Part II should trace back to a shall-statement in your SDRD.

18.8 Common Mistakes

Common PCB Layout Mistakes — What to Check Before Ordering			
Mistake	What Goes Wrong	How to Prevent	When Caught
Wrong footprint	Component doesn't fit pads; pins misaligned; board is scrap	Print 1:1 on paper; place real component on printout; verify every footprint	Board arrival
Trace too narrow	Trace burns/fuses under load current; intermittent connection	Use IPC-2221 trace width calculator for every power trace; verify in DRC	Under load testing
Missing decoupling caps	MCU resets, ADC noise, erratic digital behavior	Place 100nF cap within 5mm of every IC power pin; add bulk cap per rail	Functional testing
Poor grounding	Ground loops, noise coupling, analog drift, EMI failures	Use ground pour/plane; star-point for mixed signal; keep return paths short	Integration testing
No mounting holes	Board cannot be secured in enclosure; vibration failures	Add 4x M3 mounting holes at corners; match enclosure standoff pattern	Assembly
Connector on wrong side	Cable routing impossible; mechanical interference with enclosure	Check 3D model in EDA tool; verify against enclosure CAD before ordering	Assembly
Silkscreen under parts	Reference designators invisible after assembly; debugging impossible	Move refdes outside component outline; check silkscreen layer visibility	Assembly

Figure 18.5: Seven common PCB layout mistakes with consequences and prevention strategies.

Wrong footprint (scrap board), trace too narrow (burns under load), missing decoupling caps (erratic behavior), poor grounding (noise), no mounting holes (cannot secure board), connector on wrong side (cable routing impossible), silkscreen under parts (cannot read refdes). Prevention: print 1:1, run DRC, peer review, use 3D viewer.

18.8.1 Development Boards vs Custom PCBs

Use dev boards when: circuit is straightforward, form factor is not critical, timeline is tight, design is still evolving, or a suitable module already exists. Use custom PCBs when: specific form factor required, noise-sensitive analog, high reliability needed, multiple units, or clean cable routing. A well-wired dev board that meets requirements is superior to a custom PCB that delays the project.

18.9 Artifact Handoff: What This Chapter Supports

Artifact	Description	Key Deliverable For
Schematic	Complete circuit with all component values	CDR, FDR
PCB Layout	Gerber files ready for fabrication	CDR (procurement)
BOM (electronics)	Components with package, value, vendor, Plan B	CDR, FDR

18.10 PCB Design Rules for Student Projects

18.10.1 Trace Width and Current Capacity

The PCB trace must be wide enough to carry the required current without excessive temperature rise. The following values are per IPC-2221B at 10 °C temperature rise for external layers on 1 oz copper (35 μm thick). These are slightly higher than the conservative/derated quick-reference values in the warning box above, which include additional safety margin:

- 0.25 mm (10 mil): 0.5 A max (signal traces)
- 0.5 mm (20 mil): 1.0 A (low-power signals and buses)
- 1.0 mm (40 mil): 2.0 A (moderate power)
- 2.5 mm (100 mil): 4.0 A (motor drivers, heater connections)
- 5.0 mm (200 mil): 7.0 A (high-current power)

These values assume 10 °C temperature rise above ambient. For buried (internal) layers: derate by 50%. For 2 oz copper: multiply current capacity by $\sqrt{2} \approx 1.4$.

Use an Online IPC-2221 Calculator

The lookup table above gives conservative approximations. For exact trace width calculations, use an online IPC-2221 calculator (search “IPC-2221 trace width calculator”; Saturn PCB Toolkit is a popular free option). Enter your current, copper weight, allowable temperature rise, and ambient temperature to get a precise result. This is more accurate than a table and accounts for your specific conditions. For MEGN 301 projects, always round up to the next standard trace width and add 25% margin on power traces.

18.10.2 Clearance and Creepage

Clearance: the shortest distance through air between two conductors. For circuits under 50 V: minimum 0.25 mm (10 mil). For 50–300 V: minimum 1.5 mm.

Creepage: the shortest distance along a surface between two conductors. Creepage must be larger than clearance because surface contamination (dust, flux residue, moisture) reduces the breakdown voltage. For circuits under 50 V: minimum 0.5 mm. For mains-connected circuits: consult IPC-2221B Section 6.3 or the relevant safety standard.

18.10.3 Via Design

Vias connect traces between layers. For signal vias: 0.3 mm drill, 0.6 mm pad. For power vias: 0.4 mm drill, 0.8 mm pad, or use multiple vias in parallel for high current. Thermal vias under power components (voltage regulators, MOSFETs) connect the component pad to internal or bottom copper planes for heat dissipation. Use a grid of 4–9 vias with 0.3 mm drill directly under the thermal pad.

18.11 Component Placement Strategy

Good placement is 80% of good PCB design. Follow this sequence:

- (1) **Place connectors first:** power input, signal I/O, programming header. These have fixed locations (edge of board, near panel cutouts). All other placement works around them.
- (2) **Place the MCU:** central location, near the programming header. Orient so that bus signals (SPI, I2C, UART) route cleanly to their destinations.
- (3) **Place power components:** voltage regulator near the power input connector. Input capacitor between the regulator and connector. Output capacitor between the regulator and the MCU. Keep the power path short and wide.
- (4) **Place analog components:** sensors, amplifiers, ADCs as far as possible from switching noise sources (voltage regulators, motor drivers, digital buses). Ideally on the opposite side of the board or in a separate analog ground island.
- (5) **Place remaining digital components:** group by function (communication, display, storage). Keep I2C devices on a clean bus with short traces.

18.12 Design Review Before Fabrication

Before submitting the PCB for fabrication, perform a systematic review:

Schematic review: verify every net is connected as intended. Check power and ground connections for every IC. Verify component values match the design calculations.

Layout review: run the DRC (Design Rule Check) and fix all violations. Visually inspect: are all traces routed? Are power traces wide enough? Are decoupling capacitors close to their ICs? Is there adequate clearance around mounting holes?

3D review: if the EDA tool supports 3D rendering (KiCad, Altium), verify that components do not interfere with the enclosure, connectors are accessible, and tall components do not block test points.

Fabrication files: generate Gerber files and the drill file. Open them in a Gerber viewer (KiCad's built-in viewer, or the fab house's online viewer) and verify that the actual output matches the design. Check that the board outline, mounting holes, and silkscreen are correct.

Mandatory: 10× Visual Inspection Before Power-On

Before applying power to any newly assembled PCB, perform a visual inspection under at least 10× magnification (jeweler's loupe, USB microscope, or stereo microscope). Inspect every solder joint for:

Solder bridges: unintended connections between adjacent pads, especially on fine-pitch ICs (0.5 mm pitch QFP, 0.65 mm SOIC). A single bridge between V_{CC} and an I/O pin will destroy the IC at power-on.

Cold joints: dull, grainy, or blobby joints that did not wet properly. These create intermittent connections that work on the bench but fail under vibration.

Tombstoned components: 0402 and 0603 passives that lifted on one end during reflow, creating an open circuit. The component stands vertically like a tombstone.

Missing components: compare the board to the BOM and assembly drawing. A missing decoupling capacitor can cause oscillation; a missing pull-up resistor can hang the I2C bus.

Polarity: verify orientation of every polarized component (electrolytic capacitors, diodes, ICs, connectors). A reversed electrolytic capacitor will explode within seconds of power-on.

This inspection is a **gate**: no power is applied until a second team member has signed off on the visual inspection. This 5-minute check prevents the most common cause of PCB failure in student projects: immediate destruction of components due to assembly errors that were visible but unchecked.

Quick Reference: PCB Specifications for JLCPCB/PCBWay Student Orders

Parameter	Typical Student Project Values
Layer count	2-layer (default; 4-layer for mixed-signal or dense designs)
Copper weight	1 oz/ft ² external (standard); 2 oz for high-current (>3 A) traces
Min trace/space	6 mil / 6 mil (JLCPCB standard); use 10/10 mil unless space-constrained
Min via drill/pad	0.3 mm drill / 0.6 mm pad (standard); 0.2/0.45 mm available
Trace width: 1 A	≥20 mil (1-oz external copper, 10 °C rise)
Trace width: 3 A	≥50 mil (1-oz) or ≥30 mil (2-oz)
Decoupling cap placement	≤5 mm from IC power pin; via to ground plane directly under cap pad
Board thickness	1.6 mm (standard); 0.8 mm for flexible mounting
Surface finish	HASL (cheapest), ENIG (flat pads, fine-pitch, \$2–5 extra)
Typical cost	\$2–7 for 5 boards (2-layer, <100×100 mm)
Lead time	5–7 days (standard), 1–3 days (express, \$15–30 extra)

18.13 Practice Questions

1. Your project has a motor driver that draws 3 A peak. What is the minimum trace width for the motor power traces on a 2-layer board with 1-oz external copper? What width would you actually use and why?
2. A teammate's PCB layout has the 100 nF decoupling capacitors placed 25 mm from the MCU power pins "because there was more room over there." Explain why this is a problem and what the correct placement is.
3. You receive your first PCB from JLCPCB. Describe the exact sequence of steps you follow before connecting the board to your system.
4. Your team is debating whether to design a custom PCB or use an Arduino Nano + sensor breakout boards. The project is a battery-powered environmental monitor that must fit inside a 60×40×25 mm waterproof enclosure. Argue for or against a custom PCB with specific reasoning.

5. A 2-layer board has all signal traces on the top layer and a ground pour on the bottom. A teammate routes a high-current motor trace on the bottom layer, cutting through the ground pour. What problems will this cause and how should it be fixed?
6. Your schematic uses an LM1117-3.3 voltage regulator in an SOT-223 package to power an ESP32. The datasheet shows a thermal pad on the bottom. Your PCB layout connects the thermal pad to a small isolated copper island with no vias. Calculate the power dissipation at 5 V input, 3.3 V output, 300 mA load, and explain why the current thermal design is inadequate.
7. You are designing a mixed-signal board with a 12-bit ADC reading a strain gauge (millivolt-level signal) and an H-bridge motor driver on the same PCB. Describe three specific layout strategies to prevent motor noise from corrupting the ADC readings.
8. Your team's KiCad DRC reports 47 "courtyard overlap" warnings. Your teammate dismisses them: "It compiles fine, just ignore warnings." Explain why courtyard overlaps are serious and what physical problem they represent.

18.13.1 Answers

1. From the IPC-2221 reference: 3 A on 1-oz external copper requires approximately 100 mil (2.5 mm) trace width. In practice, use 150–200 mil (3.8–5 mm) to provide thermal margin, reduce voltage drop, and account for peak transients. Always add margin on power traces; the cost is board space, not money.
2. At 25 mm, the trace inductance between the cap and the IC power pin negates the decoupling effect. High-frequency noise (MHz range) sees the inductance as a high impedance and is not filtered. The cap must be within 5 mm with the shortest possible traces and a via directly to the ground plane. The loop area (VCC pin → cap → GND via → GND plane → IC GND pin) must be minimized.
3. Follow the six-step bring-up protocol: (1) Visual inspection under magnification for bridges, cold joints, and wrong-orientation parts. (2) Multimeter continuity check for VCC-to-GND shorts. (3) Current-limited bench supply at 50 mA, apply expected voltage. (4) Measure all power rails with multimeter. (5) Increase current limit to operating level, monitor 60 seconds for heat/smell. (6) Load firmware, test basic I/O.
4. Argue FOR custom PCB: the 60×40×25 mm enclosure is too small for an Arduino Nano (45×18 mm) plus breakout boards plus wiring. A custom board can integrate the MCU, sensors, and battery management into the available footprint with board-edge connectors for waterproof cable glands. The waterproof requirement also favors a single board with conformal coating over loose wiring that can vibrate and corrode. However: this requires the design to be frozen by CDR and 4–6 weeks of lead time.
5. The motor trace cuts the ground pour, forcing signal return currents to detour around the gap. This creates a large current loop that radiates EMI and couples noise into adjacent signal traces. The motor's switching noise (from PWM) will appear as voltage spikes on the ground plane near the gap, potentially corrupting ADC readings and causing MCU resets. Fix: route the motor power traces on the top layer using wide traces, keeping the bottom ground pour continuous. If the motor trace must cross to the bottom, use a narrow bridge with ground pour restored on both sides and stitching vias.

6. $P = (5.0 - 3.3) \times 0.3 = 0.51 \text{ W}$. SOT-223 θ_{JA} without thermal vias $\approx 150^\circ\text{C/W}$, so $\Delta T = 76.5^\circ\text{C}$. At 25°C ambient: $T_J = 101.5^\circ\text{C}$, approaching the 125°C limit. At 40°C ambient (inside an enclosure): $T_J = 116.5^\circ\text{C}$, dangerous. The isolated copper island with no vias provides poor thermal coupling to inner layers. Fix: add an array of thermal vias (5–9 vias, 0.3 mm drill, 1.0 mm pitch) under the thermal pad, connecting to the ground plane on Layer 2. This reduces θ_{JA} to $\sim 60^\circ\text{C/W}$.
7. (1) Physical separation: place ADC and strain gauge conditioning on the left half of the board, motor driver on the right, with a clear “quiet zone” boundary between them. (2) Separate analog supply: use an RC filter ($10\ \Omega + 10\ \mu\text{F}$) between digital VCC and the ADC’s AVCC pin to prevent motor switching noise from reaching the analog reference. (3) Guard ring/trace: surround the strain gauge signal traces with a grounded guard trace connected to the analog ground, shielding them from electric field coupling. Additionally: route motor power traces perpendicular (not parallel) to analog signal traces when crossings are unavoidable.
8. Courtyard overlaps mean two components occupy the same physical space on the board. They will either mechanically interfere (components crash into each other during assembly), or their solder pads will short together. DRC warnings are not cosmetic; they represent physical manufacturing problems. Fix: move one component until courtyards no longer overlap. If space is genuinely tight, verify in the 3D viewer that the components physically clear each other.

Chapter 19

Microcontrollers & the ESP32

When to Read This Chapter

Sprint 3: Read the ESP32 deep dive, gotchas, and safe GPIO sections before assigning pins at CDR. Print the GPIO pin map table and tape it to your workbench. **Sprint 4–5:** Return to the communication protocol and firmware best practices sections during integration. **Related:** Sensors in Chapter 14 (p. 148); driver circuits in Chapter 12 (p. 125); PCB layout in Chapter 18 (p. 186); debugging in Chapter 20 (p. 210).

19.1 MCU Fundamentals

19.1.1 What Is a Microcontroller?

A microcontroller (MCU) integrates a CPU, memory (flash + SRAM), and peripherals (GPIO, ADC, timers, communication) on a single chip. Unlike a microprocessor (Raspberry Pi), an MCU requires no external RAM or storage; it is self-contained and runs bare-metal or RTOS code.

Key blocks: CPU core (executes instructions, 16–600 MHz), flash memory (stores program, 32 KB–16 MB), SRAM (runtime variables, 2 KB–520 KB), peripherals (GPIO, ADC, DAC, PWM, timers, watchdog, RTC), communication (UART, SPI, I2C, CAN, Wi-Fi, BLE), and power management (regulators, sleep modes, brown-out detect).

19.1.2 MCU Platform Comparison

Selection criteria: GPIO count (add 20% margin), communication protocols needed, processing power, power budget (deep sleep current for battery), ecosystem (libraries, community), and cost/availability.

19.2 ESP32 Deep Dive

ESP32 Variant Landscape

Espressif has released newer variants of the ESP32 family, including the ESP32-S3 (USB-OTG and vector instructions for AI/ML acceleration), the ESP32-C3 (RISC-V core at lower cost), and the ESP32-C6 (Wi-Fi 6, Thread, and Zigbee support). The ESP32-WROOM-32 remains the recommended platform for MEGN 301 due to its extensive library support, mature documentation, broad community resources, and low cost. Students interested in exploring newer variants for personal or capstone projects can use the same Arduino framework and programming patterns taught in this course.

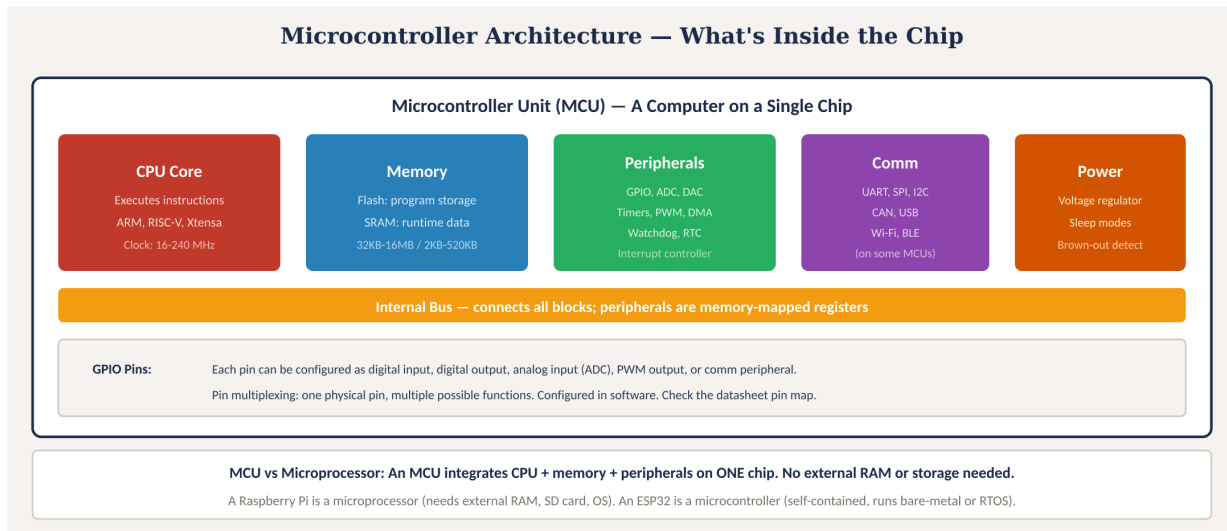


Figure 19.1: MCU internal architecture: CPU, memory, peripherals, communication, and power on one chip.

Arduino Uno	Nano 33 IoT	ESP32	STM32 Blue Pill	RPi Pico	Teensy 4.0
ATmega328P 16 MHz, 8-bit 32 KB flash 2 KB RAM 14 GPIO, 6 ADC UART, SPI, I2C No wireless \$3-5	SAMD21 48 MHz, 32-bit 256 KB flash 32 KB RAM 14 GPIO, 8 ADC UART, SPI, I2C Wi-Fi + BLE \$15-20	Xtensa LX6 240 MHz, dual-core 4 MB flash 520 KB RAM 34 GPIO, 18 ADC UART, SPI, I2C, CAN Wi-Fi + BT + BLE \$4-8	Cortex-M3 72 MHz, 32-bit 64 KB flash 20 KB RAM 37 GPIO, 10 ADC UART, SPI, I2C, CAN No wireless \$2-5	RP2040 133 MHz, dual-core 2 MB flash 264 KB RAM 26 GPIO, 3 ADC UART, SPI, I2C Pico W: Wi-Fi \$4-6	Cortex-M7 600 MHz, 32-bit 2 MB flash 1 MB RAM 40 GPIO, 14 ADC UART, SPI, I2C, CAN No wireless \$20-25

Selection criteria: GPIO count (+20% margin), protocols needed, processing power, power budget, ecosystem, cost/availability.

Figure 19.2: Six MCU platforms compared. ESP32 recommended for MEGN 301.

The ESP32-WROOM-32 is the recommended MCU for MEGN 301 projects: dual-core Xtensa LX6 at 240 MHz, 520 KB SRAM, 4 MB flash, built-in Wi-Fi (802.11 b/g/n) and Bluetooth (Classic + BLE 4.2), 34 GPIO, 18 ADC channels (12-bit), 2 DAC channels (8-bit), 16 PWM channels, 3 UART, 2 SPI, 2 I2C, 1 CAN, 10 capacitive touch pins, deep sleep at $\sim 10 \mu\text{A}$. Cost: \$4–8 for a DevKit board.

ESP32 Deep Dive — The MEGN 301 Standard MCU

ESP32-WROOM-32 Specifications

Core:	Dual-core Xtensa LX6 @ 240 MHz
Memory:	520 KB SRAM, 4 MB flash (up to 16 MB)
Wi-Fi:	802.11 b/g/n, 2.4 GHz, station + AP mode
Bluetooth:	BLE 4.2 + Classic (dual-mode)
GPIO:	34 GPIO (not all exposed on DevKit)
ADC:	18 channels, 12-bit (but noisy above 3.1V)
DAC:	2 channels, 8-bit (GPIO 25, 26)
PWM:	16 channels (LEDc peripheral), any GPIO
Comm:	3x UART, 2x SPI, 2x I2C, 1x I2S, 1x CAN
Power:	3.3V logic, 80 mA (Wi-Fi off), 240 mA (Wi-Fi)
Touch:	10 capacitive touch-sense pins built-in
Sleep:	Deep sleep $\sim 10 \mu\text{A}$; wake on timer, touch, or GPIO
Cost:	\$4-8 for DevKit board (ESP32 chip $\sim$ \$2)

ESP32 Gotchas (Know These!)

GPIO 6-11: connected to flash — DO NOT USE

GPIO 34-39: input only — no output, no pull-up

ADC2 disabled when Wi-Fi active (use ADC1 only)

ADC nonlinear above $\sim 3.1\text{V}$ — calibrate or use external ADC

3.3V logic — use level shifter for 5V sensors/actuators

Boot pins (GPIO 0, 2, 12, 15) — avoid for critical signals

Development Environment

Arduino IDE: easiest, huge library ecosystem, recommended for 301

PlatformIO: professional, better dependency management, VS Code

ESP-IDF: Espressif native SDK, FreeRTOS, maximum control

MicroPython: Python on ESP32, rapid prototyping, slower execution

ESP32 variants: ESP32-S3 (USB native, AI accelerator), ESP32-C3 (RISC-V, lower cost), ESP32-C6 (Wi-Fi 6, Thread/Zigbee).

Figure 19.3: ESP32 specifications, gotchas, and development environment.

ESP32 Gotchas: Memorize These

- **GPIO 6–11:** connected to internal flash. DO NOT USE.
- **GPIO 34–39:** input only; no output, no internal pull-up.
- **ADC2 disabled when Wi-Fi active:** use ADC1 (GPIO 32–39) only.
- **ADC nonlinear above $\sim 3.1\text{V}$:** use `esp_adc_cal` library for calibration; design voltage dividers to map sensor output into the 0–2.5 V linear range rather than using the full 0–3.3 V span. For precision, use an external ADC (ADS1115, 16-bit, I2C, \$3).
- **3.3 V logic:** use level shifters for 5 V sensors/actuators.
- **Boot/strapping pins (0, 2, 5, 12, 15):** avoid for critical signals; unexpected levels prevent boot.

19.2.1 Safe GPIO Selection

Safe for any use: GPIO 4, 13, 14, 16–19, 21–23, 25–27, 32, 33. **Input only:** GPIO 34, 35, 36 (VP), 39 (VN). **Boot-sensitive:** GPIO 0, 2, 12, 15 (usable after boot with care). **Off-limits:** GPIO 6–11 (flash), 1 (TX0), 3 (RX0).

Quick Reference: ESP32 GPIO Pin Map

GPIO	ADC	Category	Notes
4, 13, 14, 16, 17	—	Safe	General purpose, any function
18, 19, 21, 22, 23	—	Safe	SPI default pins (18=SCK, 19=MISO, 23=MOSI)
25, 26, 27	ADC2	Safe	Also DAC outputs (25=DAC1, 26=DAC2)
32, 33	ADC1	Safe	Preferred analog inputs (Wi-Fi safe)
34, 35, 36, 39	ADC1	Input only	No output, no pull-up; use external pull
0, 2, 12, 15	varies	Boot-sensitive	Float or pull as needed; avoid driving at boot
6–11	—	Off-limits	Internal flash SPI; will crash if used
1 (TX0), 3 (RX0)	—	Serial	USB-serial; avoid for other functions

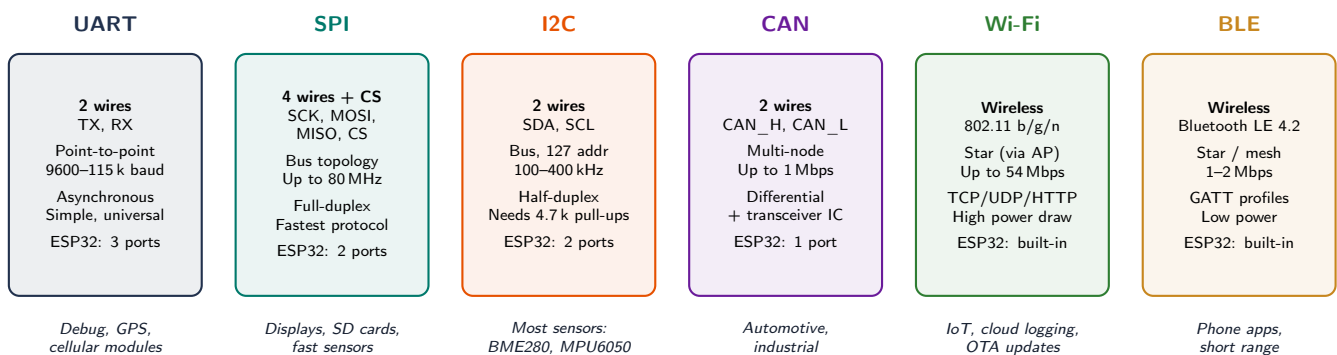
19.2.2 Wi-Fi and Bluetooth

Wi-Fi: Station mode (connect to router), AP mode (create network), or both. ESP-NOW for peer-to-peer without router. **BLE:** GATT server/client for phone apps; lower power than Classic BT. **Classic BT:** Serial Port Profile for wireless UART.

19.2.3 Power Modes

Active (~ 240 mA with Wi-Fi), modem sleep (~ 20 mA, radio off), light sleep (~ 0.8 mA, CPU paused), deep sleep (~ 10 μ A, RTC only). Battery sizing example: 2000 mAh LiPo at active+Wi-Fi = 10 hours; at deep sleep with 1-min wakeups (~ 0.5 mA average) = 166 days.

19.3 Communication Protocols



Decision rule: I2C for sensors (simple). SPI for displays/SD (fast). UART for debug/GPS. Wi-Fi/BLE for IoT. CAN for automotive.

Figure 19.4: Six communication protocols with wire count, speed, topology, and ESP32 support.

UART: 2 wires, point-to-point, 9600–115200 baud. Debug, GPS, cellular. **SPI:** 4 wires + CS, bus, up to 80 MHz. Displays, SD cards, fast sensors. **I2C:** 2 wires, bus, 100–400 kHz, 127 addresses. Most sensors (BME280, MPU6050). Needs 4.7 k Ω pull-ups. **CAN:** 2 wires + transceiver, multi-node, 1 Mbps. Automotive/industrial. **Wi-Fi/BLE:** wireless IoT.

Protocol Decision Rule

I2C for sensors (simple, 2 wires). SPI for displays and SD cards (fast). UART for debug and GPS. Wi-Fi/BLE for IoT connectivity. CAN for automotive.

19.4 Firmware and Integration

19.4.1 Firmware Best Practices

Use **state machines** (IDLE, READING, TRANSMITTING, SLEEPING). Use `millis()`-based timing instead of `delay()`. Use **interrupts** for hardware events (ISRs must be short; set a flag, process in loop). Enable the **watchdog timer** for crash recovery. Add a **heartbeat LED** as a health indicator.

19.4.2 System Integration

Power: 3.3 V regulated or 5 V via USB DevKit. Add $10\ \mu\text{F} + 100\ \text{nF}$ decoupling at MCU power pins. Wi-Fi transmit draws 240 mA transient; inadequate decoupling causes brown-out resets. **Level shifting:** BSS138-based modules for I2C/SPI to 5 V devices. **I2C:** $4.7\ \text{k}\Omega$ pull-ups to 3.3 V. **Analog:** keep wires short, average readings for noise reduction.

19.5 Artifact Handoff: What This Chapter Supports

Artifact	Description	Key Deliverable For
Pin Map	GPIO assignments with function and component	CDR, FDR
Firmware Architecture	State machine, ISRs, communication protocols	CDR, FDR
Power Mode Plan	Active/sleep duty cycle for battery sizing	CDR (power budget)

19.5.1 Development Environment Setup

Arduino IDE (recommended for MEGN 301): Install Arduino IDE 2.x, add ESP32 board support via Board Manager¹, select “ESP32 Dev Module,” and verify with the Blink example. **PlatformIO:** more professional, supports version-pinned dependencies and multi-file projects. **ESP-IDF:** Espressif’s native framework; most powerful but steepest learning curve.

19.5.2 Common Integration Patterns

- **Sensor polling loop:** read all sensors in `loop()`, average 3–5 readings, apply calibration, publish to cloud/display. Use `millis()`-based timing, never `delay()`.
- **Actuator control:** state machine with IDLE, HEATING, COOLING, FAULT states. Transitions triggered by sensor thresholds with hysteresis (e.g., turn on at $3\ ^\circ\text{C}$, turn off at $5\ ^\circ\text{C}$ to prevent relay chatter).
- **Data logging:** write sensor data to SPIFFS/LittleFS with timestamps. Upload batch to cloud on Wi-Fi schedule. Buffer locally if Wi-Fi is unavailable.

¹Board Manager URL: https://espressif.github.io/arduino-esp32/package_esp32_index.json

- **OTA updates:** enable ArduinoOTA for wireless firmware updates during development. Disable in production firmware for security.

19.5.3 Battery Sizing Worked Example

GreenShield off-grid: ESP32 wakes every 60 s for 2 s of active measurement (80 mA average during wake, 10 μ A during sleep). Average current:

$$I_{avg} = \frac{2 \times 80 + 58 \times 0.01}{60} = 2.68 \text{ mA} \quad (19.1)$$

With a 3000 mAh LiPo: runtime = $3000/2.68 = 1119$ hours = 46 days. Adding Wi-Fi transmit (240 mA for 1 s every 5 minutes): adds 0.80 mA average, total 3.48 mA, runtime = 36 days. Add 30% margin for battery aging: design for 25 days between charges.

19.6 Sensor Interfacing Patterns

19.6.1 Analog Sensors

Most analog sensors output a voltage proportional to the measured quantity. The ESP32's 12-bit ADC maps 0–3.3 V to 0–4095 counts. Conversion: $V = \text{count} \times 3.3/4095$. Then apply the sensor's transfer function (from datasheet) to convert voltage to engineering units.

Common problems and fixes:

- **Noise:** average 10–50 readings per sample. Use a moving average or median filter in firmware.
- **Nonlinearity:** the ESP32 ADC is nonlinear above ~ 3.1 V. Use the ESP-IDF calibration API (`esp_adc_cal`) or limit input range to 0–3.0 V with a voltage divider. For best results: characterize your specific ESP32 module by measuring 10 known voltages (0.0, 0.3, 0.6, ..., 3.0 V from a bench supply) and fitting a polynomial correction in firmware. The 0–2.5 V range is the most linear; design your voltage divider to map the full sensor range into this window rather than using the full 0–3.3 V span.
- **Offset/gain error:** calibrate against a known reference (multimeter) at two points and apply linear correction: $V_{true} = a \times V_{raw} + b$.
- **5 V sensors:** use a voltage divider or external ADC (ADS1115, 16-bit, I2C, \$3).

19.6.2 Digital Sensors (I2C)

Most modern sensors (temperature, humidity, IMU, pressure) use I2C. Basic integration:

1. Connect SDA to GPIO 21, SCL to GPIO 22 (ESP32 defaults).
2. Add 4.7 k Ω pull-up resistors from SDA and SCL to 3.3 V. **One pair per bus**, not per device.
3. Install the sensor's Arduino library via Library Manager.
4. Initialize with `Wire.begin()` and use the library's `.read()` methods.
5. If the sensor is not detected: check wiring, verify address with an I2C scanner sketch, check pull-ups with a multimeter (SDA/SCL should read 3.3 V when idle).

19.6.3 Motor and Actuator Control

DC motors: use `ledcSetup()` and `ledcWrite()` for hardware PWM at 20–25 kHz (above audible range). Never drive a motor directly from a GPIO pin. Always use a driver IC (L298N, BTS7960, DRV8871) or MOSFET with flyback diode.

Servo motors: use the ESP32Servo library. Attach to any PWM-capable GPIO. Standard range: 500–2400 μ s pulse width at 50 Hz. Power servos from a separate 5–6 V supply, not from the ESP32’s 3.3 V rail.

Relay control: drive a relay module’s signal input from a GPIO through a 1 k Ω resistor. Use optocoupled relay modules to isolate the MCU from relay coil transients.

19.7 Debugging and Troubleshooting

19.7.1 Serial Monitor Debugging

`Serial.println()` is the primary debugging tool. Best practices:

- Print sensor values with labels: `Serial.print("Temp: "); Serial.println(temp);`
- Add state machine transitions: `Serial.println("STATE: HEATING");`
- Print timestamps: `Serial.print(millis()); Serial.print(" ms: ");`
- Remove or conditionally compile debug prints before final deployment (`#define DEBUG`).

19.7.2 Common ESP32 Failure Modes

Symptom	Likely Cause	Fix
Random resets	Brown-out (insufficient decoupling)	Add 100 μ F + 100 nF at regulator output
Won’t boot	GPIO 0/2/12/15 held at wrong level	Disconnect boot-sensitive pins during upload
ADC reads garbage	ADC2 + Wi-Fi conflict, or input >3.3 V	Use ADC1 channels; add voltage divider
I2C not detected	Missing pull-ups, wrong address, SDA/SCL swapped	Check with I2C scanner; verify pull-ups
Wi-Fi disconnects	Antenna blocked by ground plane or metal	Keep antenna area clear of copper pour
Servo jitter	Shared power supply with MCU	Use separate servo power supply

Table 19.1: Common ESP32 failure modes, causes, and fixes.

Requirements Mapping: From MCU Specs to Shall-Statements

MCU capabilities generate system requirements:

- “Read 4 analog sensors” → SYS-REQ: The system shall acquire analog data from 4 channels at ≥ 10 Hz using ADC1 (GPIO 32–35).
- “Wireless data logging” → SYS-REQ: The system shall transmit sensor data to a cloud server via Wi-Fi at ≥ 1 upload per 5 minutes.
- “Battery operation for 30 days” → SYS-REQ: The system shall operate from a 3000 mAh LiPo for ≥ 30 days with a duty cycle of 2 s active per 60 s.

The battery sizing calculation in this chapter directly verifies the last requirement. Every MCU design choice must trace to a shall-statement.

19.8 Microcontroller Fundamentals

A microcontroller (MCU) is a complete computer on a single chip: processor (CPU), memory (RAM and flash), input/output peripherals, and communication interfaces. Unlike a desktop CPU (which requires external memory, storage, and I/O), the MCU is self-contained and designed for embedded applications where cost, power, and size are constrained.

19.8.1 Key Specifications

Clock speed: determines the instruction execution rate. The ESP32 runs at 240 MHz (dual core), executing ~240 million simple instructions per second. This is more than adequate for control loops running at 1 kHz, sensor acquisition, and display updates.

Memory: RAM stores runtime variables (sensor readings, control states, buffers). Flash stores the program (firmware) and constants (calibration tables). The ESP32 has 520 kB RAM and 4 MB flash, sufficient for most MEGN 301 projects. If you run out of RAM: optimize data structures, reduce buffer sizes, or use external SPI RAM.

GPIO (General Purpose I/O): pins that can be configured as digital input, digital output, analog input (ADC), PWM output, or communication interface. The ESP32 has 34 GPIO pins, though some have restrictions (input-only, strapping pins, occupied by flash interface).

ADC: the ESP32's internal ADC is 12-bit with significant nonlinearity and noise (± 50 LSB at 12-bit). For precision measurement: use an external ADC (ADS1115: 16-bit, I2C, \$3) or the NI DAQ. The internal ADC is adequate for: battery voltage monitoring, potentiometer reading, and coarse sensor measurement where $\pm 5\%$ accuracy suffices.

Communication interfaces: UART (serial: 3 ports), SPI (high-speed: 2 ports), I2C (moderate speed: 2 ports). Wi-Fi (802.11b/g/n) and Bluetooth (Classic + BLE) are built-in, enabling wireless data logging, remote monitoring, and over-the-air firmware updates.

19.9 ESP32 Pin Planning

Before writing code, create a pin assignment table mapping every signal to a specific GPIO pin. Constraints to observe:

GPIO 0, 2, 5, 12, 15: strapping pins that affect boot mode. Avoid using for general I/O unless you understand the boot-time implications.

GPIO 34–39: input only (no output capability). Use for analog inputs or digital inputs only.

GPIO 6–11: connected to the internal flash. Do not use.

ADC2 (GPIO 0, 2, 4, 12–15, 25–27): cannot be used for ADC while Wi-Fi is active. Use ADC1 (GPIO 32–39) for analog inputs in Wi-Fi-enabled projects.

19.10 Firmware Architecture for Control Systems

19.10.1 Superloop Architecture

The simplest firmware structure: a single infinite loop that reads sensors, computes control, writes actuators, and updates the display, repeating at a fixed rate:

```
void loop() {  
    read_sensors();  
    compute_pid();  
}
```

```

write_actuators();
update_display();
delay_until_next_period();
}

```

The superloop is adequate for single-rate systems with modest complexity. It fails when: (1) the loop takes longer than the control period (timing violation), (2) different tasks need different rates (sensor at 100 Hz, display at 10 Hz, logging at 1 Hz), or (3) blocking operations (Wi-Fi, SD card write) stall the control loop.

19.10.2 Task-Based Architecture (FreeRTOS)

The ESP32 includes FreeRTOS, a real-time operating system that runs multiple tasks concurrently with priority-based scheduling. Each task has its own stack and runs as if it were the only program. The RTOS scheduler switches between tasks automatically:

High priority: control loop (reads sensor, computes PID, writes actuator). Runs every 10 ms without interruption.

Medium priority: data logging (writes to SD card). Runs every 1 s; may be interrupted by the control task.

Low priority: display update and Wi-Fi communication. Runs when higher-priority tasks are idle.

This ensures the control loop always meets its timing deadline, even when lower-priority tasks are slow.

19.11 Common ESP32 Pitfalls in MEGN 301

(1) **Watchdog timer resets:** the ESP32's watchdog timer resets the system if the main loop (or a FreeRTOS task) does not yield within ~ 3 s. Long operations (SD card initialization, Wi-Fi connection) trigger the watchdog. Solution: call `yield()` or `vTaskDelay(1)` inside long operations.

(2) **Power supply noise:** the ESP32's Wi-Fi transmitter draws 120–240 mA current pulses that cause voltage sag on weak supplies. Symptoms: random resets, ADC reading spikes, Wi-Fi disconnections. Solution: use a dedicated 5 V supply with ≥ 1 A capacity and place a $470\ \mu\text{F}$ capacitor at the ESP32's 3.3 V input.

(3) **GPIO conflicts:** accidentally configuring a strapping pin as output, or using an ADC2 pin while Wi-Fi is active. Symptoms: boot failure, garbage ADC readings. Solution: always check the pin assignment table and the ESP32 datasheet before committing to a pin map.

(4) **Floating inputs:** unconnected GPIO pins read random noise. If your code checks a button on an unconnected pin, it sees phantom presses. Solution: enable internal pull-up or pull-down resistors on all input pins, or connect external pull-ups/pull-downs.

Requirements Mapping: Microcontroller Selection

From technical specs to “shall” statements:

The ESP32 ADC has 12-bit resolution on a 0–3.3 V range ($\text{LSB} = 0.8\ \text{mV}$) with ± 50 LSB noise ($\pm 40\ \text{mV}$). For a temperature sensor producing $10\ \text{mV}/^\circ\text{C}$: resolution is $0.8/10 = 0.08\ ^\circ\text{C}$ but noise floor is $40/10 = 4\ ^\circ\text{C}$. This becomes: “SYS-ADC-001: The temperature measurement subsystem shall achieve $\pm 0.5\ ^\circ\text{C}$ accuracy. *Note: The ESP32 internal ADC cannot meet this requirement. An external ADS1115 (16-bit, $\pm 0.01\%$ accuracy) shall be used.*”

The ESP32's Wi-Fi transmitter draws 240 mA peak, causing ~200 mV sag on a weak 3.3 V supply. This becomes: "SYS-PWR-003: The 3.3 V rail shall maintain $\pm 5\%$ regulation under Wi-Fi transmit load. *Verification: Test: measure 3.3 V rail with oscilloscope during Wi-Fi scan.*"

19.12 Practice Questions

1. Your project requires: 4 analog sensors, 2 PWM motor outputs, I2C temperature sensor, SPI display, Wi-Fi for data logging, and battery operation. Select an MCU platform and justify your choice. Create the pin map.
2. An ESP32 project works on USB power but the MCU resets randomly when running on a 3.7 V LiPo battery through a 3.3 V regulator. What is the likely cause? Propose a fix.
3. You need to read a 0–5 V analog pressure sensor with an ESP32. What problem does this create and how do you solve it?
4. Explain why ADC2 channels should not be used in an ESP32 project that uses Wi-Fi. What happens if you do?
5. A teammate connects a motor driver's PWM input to ESP32 GPIO 8. The ESP32 fails to boot. Why? What GPIO should they use instead?
6. Your I2C temperature sensor (BME280) works perfectly on the bench, but returns NaN after you add a second I2C device (OLED display) to the same bus. Both devices use different I2C addresses. What are the three most likely causes?
7. Design a firmware state machine for the GreenShield frost protection controller. Define all states (at least 4), all transitions, and the conditions that trigger each transition. Include a FAULT state.
8. Your battery-powered sensor node needs to last 60 days on a 2000 mAh LiPo. The active current is 80 mA and the deep sleep current is 10 μ A. What is the maximum duty cycle (active seconds per minute) you can afford?

19.12.1 Answers

1. ESP32; meets all requirements: 4 ADC1 channels (GPIO 32, 33, 34, 35), 2 PWM (GPIO 25, 26 via LEDC), I2C on GPIO 21 (SDA) / 22 (SCL), SPI display on GPIO 18 (SCK) / 23 (MOSI) / 5 (CS), Wi-Fi built-in, deep sleep for battery life. No other \$4–8 MCU provides Wi-Fi + sufficient ADC + PWM + SPI + I2C simultaneously.
2. Insufficient decoupling. Wi-Fi transmit draws ~240 mA transient. If the regulator cannot supply this or the decoupling capacitance is too small, the 3.3 V rail droops below the brown-out threshold (~2.7 V) and the ESP32 resets. Fix: add 100 μ F electrolytic + 100 nF ceramic at the regulator output. Ensure the regulator can supply ≥ 500 mA.
3. ESP32 GPIO is 3.3 V maximum. Applying 5 V directly will damage the pin. Solution: use a resistive voltage divider (e.g., 10 k Ω + 6.8 k Ω) to scale 0–5 V to 0–3.0 V (staying below the 3.1 V ADC linearity limit), or use an external ADC (ADS1115) with its own voltage reference.

4. ADC2 shares hardware with the Wi-Fi radio. When Wi-Fi is active, the ADC2 peripheral is unavailable. Calling `analogRead()` on an ADC2 pin returns garbage (random values or zero). Always use ADC1 channels (GPIO 32–39) for analog readings in Wi-Fi projects.
5. GPIO 6–11 are connected to the internal SPI flash that stores the firmware. Any external connection on these pins interferes with flash communication, preventing the bootloader from loading the program. The ESP32 fails to boot or crashes immediately. Use any safe GPIO (4, 13, 14, 16–19, 21–23, 25–27, 32, 33).
6. (1) Duplicate I2C address: despite having different *default* addresses, some modules have configurable addresses that may conflict. Run an I2C scanner to verify both addresses are unique. (2) Pull-up resistor overload: each device may have its own pull-ups on the breakout board. Two sets of 4.7 k Ω in parallel = 2.35 k Ω , which may be too strong for 3.3 V I2C. Remove the on-board pull-ups from one device. (3) Bus capacitance: the OLED's long ribbon cable adds capacitance, slowing I2C rise times. Try reducing the I2C clock speed from 400 kHz to 100 kHz.
7. States: IDLE (monitoring, heater off), HEATING (heater on, monitoring), COOLDOWN (heater off, waiting for hysteresis), FAULT (sensor failure or communication loss). Transitions: IDLE $\rightarrow$ HEATING when $\text{temp} \leq \text{setpoint}$. HEATING $\rightarrow$ COOLDOWN when $\text{temp} \geq \text{setpoint} + \text{hysteresis}$ (e.g., +2 $^{\circ}\text{C}$). COOLDOWN $\rightarrow$ IDLE when temp stabilizes. Any state $\rightarrow$ FAULT when sensor reads out-of-range ($< -40^{\circ}\text{C}$ or $> 85^{\circ}\text{C}$) or I2C communication fails 3 consecutive times. FAULT $\rightarrow$ IDLE after manual reset via button or remote command.
8. Target runtime: 60 days = 1440 hours. Required average current: $I_{avg} = 2000/1440 = 1.39 \text{ mA}$. Let t_{on} = active seconds per 60-second cycle. $I_{avg} = (t_{on} \times 80 + (60 - t_{on}) \times 0.01)/60 = 1.39$. Solving: $t_{on} \times 80 + 0.60 - 0.01t_{on} = 83.4$. $79.99t_{on} = 82.8$. $t_{on} = 1.035 \text{ s}$. Maximum active time: $\sim 1.0 \text{ s}$ per minute. This is tight; the team should optimize wake time or use a larger battery.

Chapter 20

Systematic Debugging and Troubleshooting

When to Read This Chapter

Week 1: Read the debugging pyramid and five-step protocol now, before you touch hardware.

Sprint 4: Reread before subsystem demonstrations. **Sprint 5–6:** Keep the five-step flowchart (Figure 20.3) printed at your workbench during integration.

A working system is never assembled all at once. It is built in layers, verified at each layer, and only combined into the full system after each layer passes. Debugging is not what you do when something goes wrong; it is the disciplined verification process you follow at every stage. Students who skip this process and assemble everything at once (see Figure 6.5 in Chapter 6) (“big-bang integration,” which Chapter 6 warns against) consistently spend 5–10× more time debugging than students who follow the bottom-up approach.

20.1 Why Bottom-Up Debugging Matters

Consider a GreenShield system with five components: temperature sensor, signal conditioning amplifier, ESP32 MCU, SSR heater driver, and a 500 W cartridge heater. If you wire everything and power on, the fault could be in any of the five components, any of the four interfaces, the power supply, the wiring, or the firmware. That is at least 15 possible failure locations.

Bottom-up debugging reduces the search space at every step. Verify the sensor alone: eliminated. Verify the amplifier with a known input: eliminated. By the time you connect the full chain, the fault is almost certainly at the most recently added interface, not buried inside a previously verified component.

The Debugging Pyramid

Level 1: Component. Test each part in isolation. Does the sensor produce the right voltage? Does the MOSFET switch? Does the firmware blink an LED?

Level 2: Interface. Connect two adjacent components. Does the amplifier correctly amplify the sensor? Does the ADC read the correct voltage?

Level 3: Subsystem. Connect a functional chain (sensor → amp → ADC → MCU display). Does the displayed value match the physical input?

Level 4: System. Connect all subsystems. Does the PID controller read the sensor, compute correctly, and drive the heater to setpoint?

Never advance with failures at the current level. Failures compound exponentially.

20.2 Debugging Hardware

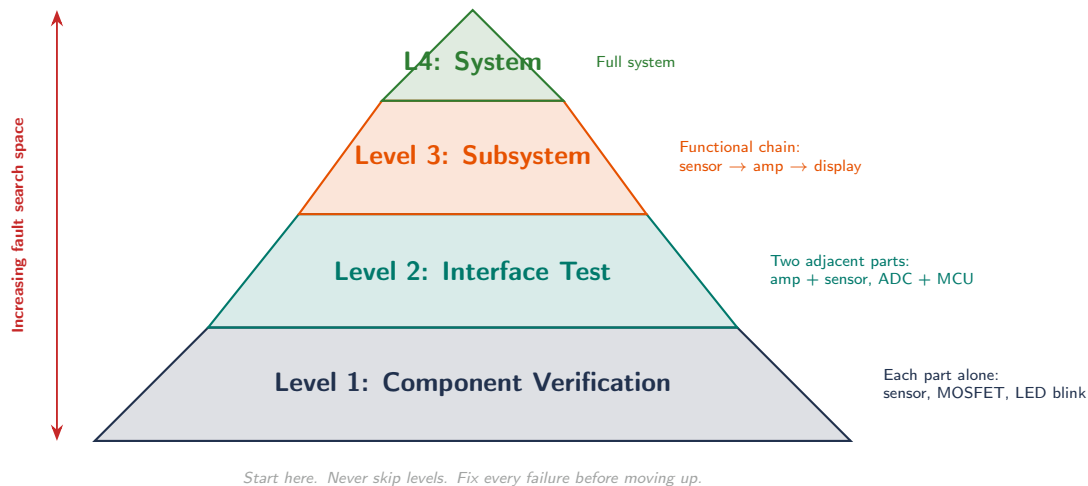


Figure 20.1: The debugging pyramid. Start at the base: verify each component in isolation. Only after all components pass, test interfaces between pairs. Then subsystem chains, then the full system. Skipping levels means the fault could be anywhere in the pyramid below, exponentially increasing the search space.

20.2.1 Visual Inspection

Before applying power, inspect: solder bridges (especially fine-pitch ICs), cold joints (dull, grainy), missing components, wrong orientation (electrolytic caps, diodes, ICs), loose wires, pinched insulation. A 10× magnifier is essential. The PCB chapter’s mandatory inspection applies here for all hardware, not just PCBs.

20.2.2 Continuity and Short-Circuit Check

Power off. Multimeter continuity mode: verify VCC and GND are not shorted, all ground connections are continuous, and signal connections reach their destinations at both ends.

20.2.3 Power-Up Sequence

Current-limited bench supply at 50 mA initially. A healthy circuit draws predictable current; a fault draws zero (open) or excessive (short). Measure every power rail before connecting ICs or modules.

20.2.4 Signal Tracing: The Signal Walk

Inject a known signal at the input and trace it stage by stage with an oscilloscope. At each node, compare measured vs. expected (from your circuit calculations). The first deviation localizes the fault.

The Signal Walk Protocol

Start at the sensor output. Measure. Move one stage downstream: after the divider, after the amplifier, at the ADC input pin. At each point ask: is the amplitude correct? Is the

offset correct? Is the frequency content correct? The first point where measured $\neq$ expected narrows the fault to that stage. Five minutes of walking saves hours of guessing.

20.2.5 Common Hardware Faults

Open connection: signal present upstream, absent or floating downstream. Oscilloscope shows 60 Hz pickup on the floating node. **Short circuit:** two nodes at the same voltage that should differ; supply current higher than expected. **Wrong value:** circuit functions but with incorrect gain, cutoff, or bias; compare measured node voltages to calculated values. **Reversed component:** electrolytic may swell; reversed diode conducts wrong direction; reversed IC draws excessive current. **Dead IC:** outputs unresponsive; replace with known-good to confirm.

Change One Thing at a Time

If you swap a component AND change a wire AND edit firmware simultaneously, and it starts working, you do not know which change fixed it. One of the others may have introduced a latent fault. Make one change, test, record, then decide the next change.

20.3 Debugging Firmware

20.3.1 Incremental Development

Mirror the hardware pyramid: (1) Blink (verify toolchain and MCU function), (2) Serial output (print variables to confirm code execution), (3) Single peripheral (one ADC channel, one GPIO, one I2C device), (4) Peripheral combinations (add one at a time, regression test), (5) Control logic (add PID only after all I/O works).

20.3.2 Common Firmware Bugs

Wrong pin assignment: GPIO in code $\neq$ physical wiring. Print the pin number; verify against wiring diagram. **Missing initialization:** ADC reads zero or random values. Verify every peripheral init call. **Blocking calls:** `delay()` stops all processing. Use `millis()`-based timing. **Integer overflow:** 16-bit counter wraps at 65,535. Use `uint32_t` for timestamps. **I2C bus hang:** failed device holds SDA low. Implement bus recovery (toggle SCL 9 $\times$) and timeouts.

Serial Monitor = Oscilloscope for Code

Print everything during development: ADC raw values, engineering units, PID error and output, state transitions, loop execution time. When behavior is unexpected, the serial log tells you exactly which variable went wrong and when. Remove verbose printing only after the system is verified.

20.4 Debugging Integrated Systems

20.4.1 The Integration Test Sequence

1. **Power rail verification:** all supply voltages correct under load before connecting active components.

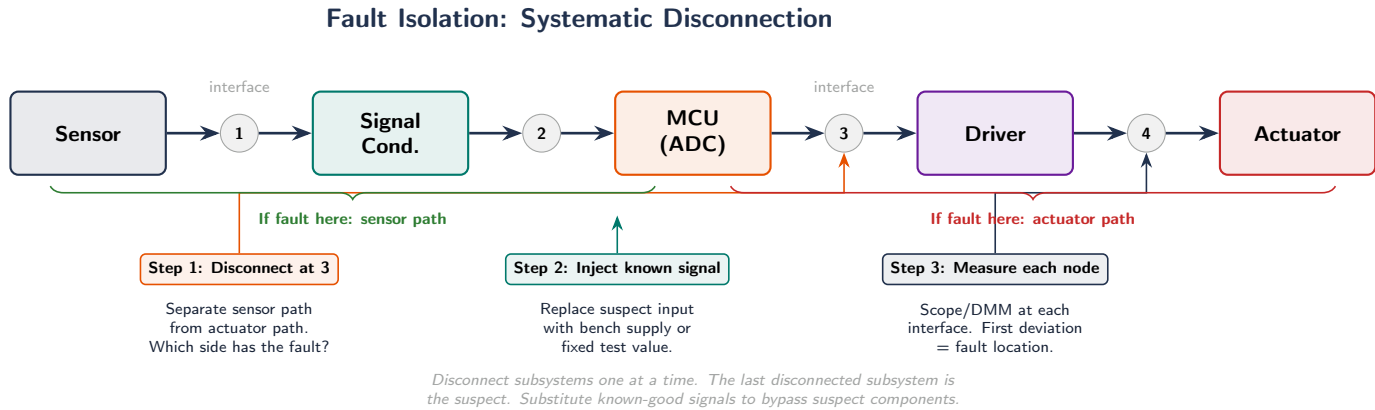


Figure 20.2: Fault isolation strategy: systematic disconnection to identify the failing subsystem or interface.

2. **Sensor-to-display path:** sensor + conditioning + ADC + MCU only. Known physical input → correct displayed value.
3. **Command-to-actuator path:** disconnect sensor feedback. Fixed command → correct actuator response.
4. **Open-loop dry run:** both paths connected, manual setpoints. Verify sensor reflects actuator effect.
5. **Closed-loop commissioning:** enable PID with conservative gains (K_p at 25% of expected, $K_i = K_d = 0$). Increase gradually.

20.4.2 Common Integration Failures

Ground loops: motor return current flows through sensor ground, injecting noise. Fix: star grounding (Chapter 12). **Brown-out:** actuator inrush current sags the supply, resetting the MCU. Fix: separate rails, bulk capacitors ($>470\mu\text{F}$). **EMI:** PWM switching noise couples into sensor signals. Fix: snubbers, decoupling, physical separation. **Timing conflicts:** loop takes longer than control period. Fix: measure loop time; prioritize time-critical tasks. **Feedback polarity error:** positive feedback instead of negative; system immediately saturates. Fix: verify increasing output drives process variable toward setpoint.

20.4.3 The Five-Step Debugging Protocol

Step 1: Reproduce. Make the failure happen reliably. A failure you cannot reproduce cannot be systematically fixed.

Step 2: Isolate. Using the fault isolation strategy (Figure 20.2), disconnect subsystems one at a time. The last one disconnected is the suspect. Or substitute a known-good signal to bypass suspect components.

Step 3: Measure. Oscilloscope, multimeter, serial monitor at the suspected location. Compare measured vs. expected.

Step 4: Hypothesize and test. Form a specific, testable hypothesis. Design one measurement that confirms or refutes it.

Step 5: Fix and regression test. Fix it. Verify the original failure is resolved. Re-run all previously passing tests to confirm nothing else broke.

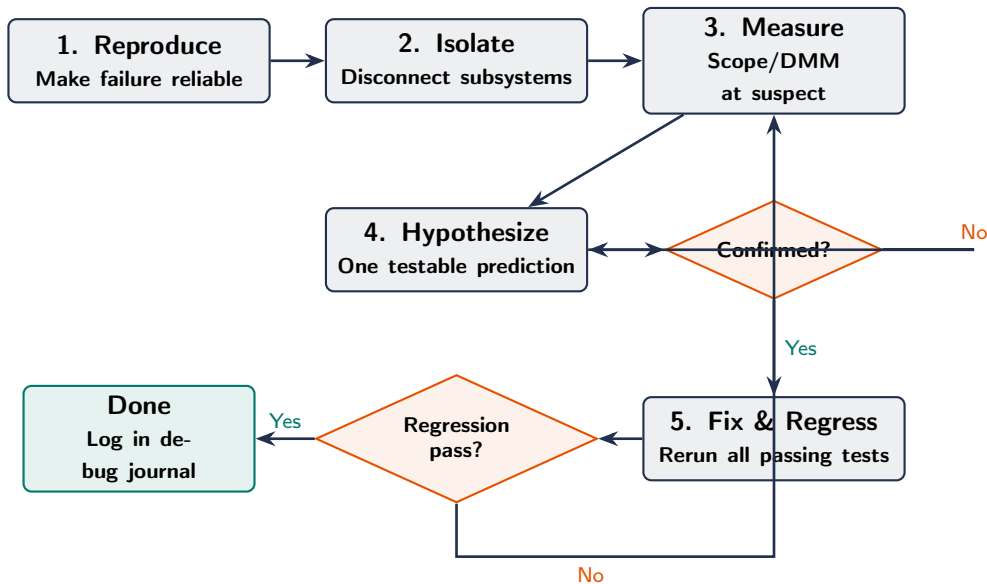


Figure 20.3: Five-step debugging protocol flowchart. Failed hypothesis returns to step 4; failed regression returns to step 3 because the fix introduced a new fault.

20.4.4 Debugging Tools

Multimeter: first reach; DC/AC voltage, current, resistance, continuity. Cannot capture transients. **Oscilloscope:** waveforms, timing, transient events, PWM verification, protocol debugging. Learn triggering. **Logic analyzer:** multi-channel digital capture with protocol decode (SPI, I2C, UART). **Bench supply current limit:** diagnostic tool, not just safety; anomalous current draw reveals faults. **Serial terminal/logger:** timestamped firmware variable capture for intermittent fault analysis.

20.5 Building a Debugging Mindset

Five principles that distinguish expert debuggers (see also the debugging pyramid in Figure 20.1):

1. Expect problems. No circuit works on the first try. No firmware is bug-free. Plan debugging time in your schedule.

2. Trust measurements, not assumptions. “It should be 2.5 V” is an assumption. “The multimeter reads 0.3 V” is a measurement. When they disagree, the measurement wins.

3. Binary search. Measure at the midpoint of the signal chain. If correct, fault is downstream; if not, upstream. Halves the search with each measurement.

4. Rubber duck debugging. Explain the problem aloud. Articulating forces you to organize your understanding and often reveals the wrong assumption.

5. Walk away after 30 minutes. Fatigue causes fixation. A fresh perspective spots the obvious error that tunnel vision concealed.

Document Every Debug Session

Keep a debug log: date, symptom, tests performed, measurements, hypothesis, fix, result. When the same failure reappears (it will), you have the answer in 30 seconds. The debug log also demonstrates professional practice for your FDR documentation.

Requirements Mapping: Debugging and Verification

“Each subsystem shall be individually verified with documented test results before integration.”
 “The integration sequence shall follow a bottom-up protocol with entry/exit criteria at each level.” “All power rails shall be verified under load before active components are connected.”
 “The debug log shall record all faults, diagnostics, root causes, and corrective actions.”

20.6 Practice Questions

1. You connect your complete GreenShield system and the heater runs at full power regardless of temperature. Describe your debugging steps starting from the pyramid.
2. Thermocouple readings are accurate with the motor off but show $\pm 5^\circ\text{C}$ noise (25 kHz spikes) when the motor runs. Root cause and three mitigations?
3. A teammate says “I changed the wiring, the firmware, and the op-amp, and now it works.” Why is this a problem?

20.6.1 Answers

1. Disconnect heater driver from MCU. Check MCU PWM output on oscilloscope: is duty cycle varying? If stuck at 100%, fault is upstream (firmware or sensor path). Disconnect sensor; substitute known voltage from bench supply: does PWM change? If not, firmware fault (PID saturated, wrong pin, positive feedback). If PWM does vary with bench supply but not with sensor, fault is in the sensor or its conditioning. Isolate one interface at a time.
2. 25 kHz matches PWM switching frequency. Motor switching current couples into thermocouple path via ground loop or shared conductor. (1) Star grounding: separate ground wires for TC amplifier and motor driver. (2) Low-pass filter ($f_c \approx 100\text{ Hz}$) at amplifier input. (3) Twisted-pair TC leads routed away from motor wiring.
3. Three simultaneous changes: cannot identify which one fixed the problem. One of the other changes may have introduced a latent fault. Correct approach: revert all three, apply one at a time, test after each.

Part III

Major Project Milestones

This part provides a comprehensive guide to the three formal design reviews that gate your project’s progression. Each chapter covers one review with deliverables, evaluation criteria, and common failure patterns.

The review sequence maps to the V-Model (see Figure 1 in the front matter):

- **PDR** (Sprints 1–3): left side of the V; problem definition, requirements, architecture, concept selection.
- **CDR** (Sprints 4–5): bottom of the V; detailed design, analysis, manufacturing planning.
- **FDR** (Sprint 7 + Final): right side of the V; verification, validation, documentation.

Each review has a distinct character:

- **PDR** asks “**Should this design work?**”; feasibility, viability, desirability.
- **CDR** asks “**Will this design work?**”; completeness, manufacturability, testability.
- **FDR** asks “**How well did it work?**”; evidence, honesty, documentation.

A strong PDR enables a strong CDR enables a strong FDR. Rushing or skipping an early review creates compounding problems at every subsequent gate.

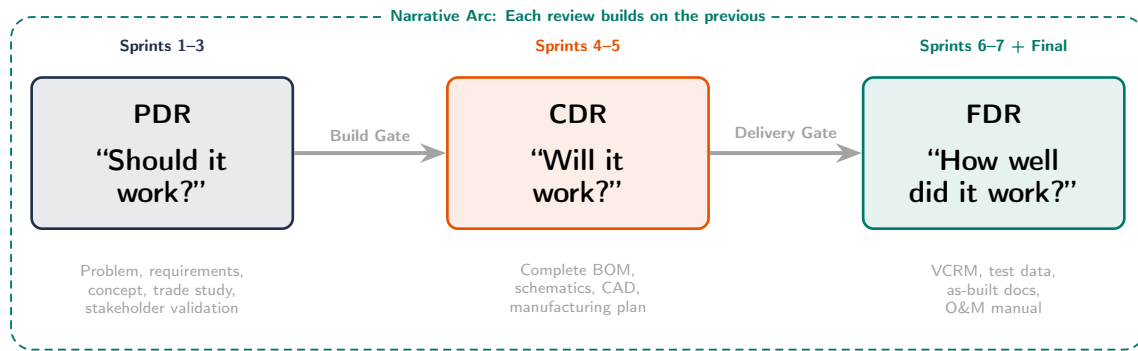


Figure 20.4: The three design reviews form a gated progression. PDR evaluates whether the concept should work (feasibility). CDR evaluates whether the design will work (completeness). FDR evaluates how well it worked (evidence). Each gate must be passed before proceeding; a weak PDR cascades into a weak CDR and a weak FDR.

Table 20.1: The Complete Artifact Progression

Artifact	At PDR	At CDR	At FDR
Problem Statement	Finalized	Unchanged	Unchanged
Stakeholder Map	Complete	Updated if needed	Final
ConOps	Approved	Unchanged	Unchanged
SDRD	Baselined (v1.0)	Updated (v2.0)	Final (v3.0)
Functional Decomp.	Complete	Unchanged	Unchanged
Trade Studies	Completed	Referenced	Archived
BOM	Preliminary	Finalized with orders	As-built
Schematics/Drawings	Preliminary	Finalized	As-built redlines
VCRM (Figure 5.5)	All Planned	Partial results	Complete
Test Procedures	Drafted	Reviewed/updated	Executed
Integration Plan	Preliminary	Detailed	Completed
O&M Manual	Not started	Outline	Complete

Chapter 21

Preliminary Design Review (PDR)

When to Read This Chapter

Sprint 2: Read this chapter in full one week before your PDR presentation. Use the deliverables checklist (Figure 21.1) to audit your package. **Before PDR:** Run the litmus test: “Could a different team arrive at the same concept selection?”

The Preliminary Design Review is the first formal design gate in the project lifecycle, marking the transition from problem definition and concept exploration into detailed design.

The PDR (first gate in Figure 20.4) answers one question: “**Should this design work?**” It evaluates whether the team has correctly identified the problem, written verifiable requirements, explored the solution space, and selected a concept that is technically feasible, economically viable, and schedule-achievable.

21.1 Purpose and Scope

PDR is the first formal gate. It confirms that the team has done enough front-end work to justify moving into detailed design and procurement. A team that passes PDR has *earned* the right to spend project funds. A team that fails PDR is redirected before wasting money on a flawed concept.

21.2 Deliverables

21.2.1 Problem Definition and Background Research

Complete five-element problem statement, stakeholder map with engagement strategies, ConOps with at least two operational scenarios (normal and degraded), competitive analysis (3–5 existing solutions with identified gap), and context diagram showing system boundary and external interfaces.

21.2.2 System Requirements and Traceability

SDRD (v1.0) with all system-level requirements. Each requirement has: unique ID, shall-statement, measurable acceptance criteria, rationale, verification method (T/A/I/D), and parent traceability to a stakeholder need. Forward and backward traceability demonstrated. No orphan requirements. No undecomposed parents.

PDR Deliverables Checklist

Problem & Stakeholders	Concept Selection
☐ Five-element problem statement	☐ Morphological chart
☐ Stakeholder map (power/interest)	☐ ≥ 3 concepts sketched
☐ ConOps (normal + degraded)	☐ Pugh screening matrix
☐ Competitive analysis (3–5 products)	☐ Weighted decision matrix
☐ Context diagram	☐ Sensitivity analysis
Requirements	Risk & Test Plan
☐ SDRD v1.0 baselined	☐ Risk register (≥ 5 risks)
☐ Traceability (fwd + bwd)	☐ VCRM with methods assigned
☐ All reqs pass quality checklist	☐ ≥ 3 test procedures drafted
Litmus Test: Could a different team arrive at the same concept selection?	

Figure 21.1: PDR deliverables checklist. Every item must be complete before the review. The litmus test at the bottom is the ultimate self-check: if a different engineering team reading only your PDR package would independently reach the same concept selection, your PDR is defensible.

21.2.3 Concept Development and Selection

Evidence of divergent thinking (see the full checklist in Figure 21.1): morphological chart with ≥ 4 functions and ≥ 3 options each. Three or more distinct concepts developed to sketch/block-diagram level. Pugh screening matrix showing systematic elimination. Weighted decision matrix for final selection with sensitivity analysis. The selected concept must be clearly justified by the data, not by team preference.

21.2.4 Risk and Feasibility

Risk register with ≥ 5 identified risks, each rated by probability and impact. Mitigation strategy for every high/medium risk. Feasibility assessment against four lenses: technical, economic, schedule, and operational. Any feasibility failure requires resolution before PDR.

21.2.5 Initial Test Plan

VCRM with all requirements listed and verification methods assigned. At least three test procedures drafted in full (all 10 sections). Equipment and facility needs identified. Test schedule aligned with integration timeline.

21.2.6 Concept Development and Selection: Detail

- **Reverse engineering analysis:** Documented disassembly and analysis of the COTS product, including component inventory, interface inventory, and constraints on the add-on system. Photographs of each disassembly step. Block diagram showing subsystem connections.

- **Concept generation evidence:** Morphological chart built from functional decomposition with ≥ 3 options per function. Evidence that divergent techniques (brainstorming, SCAMPER, bio-inspired) were used. Sketches of at least 3 distinct concepts.
- **Concept screening:** Pugh matrix showing qualitative evaluation against a datum.
- **Trade study:** Weighted decision matrix with criteria from requirements, documented weight rationale, data-backed ratings, sensitivity analysis, and clear recommendation.
- **Preliminary architecture:** Block diagram of selected concept showing subsystems, interfaces, and allocated functions.

21.2.7 Budget and Schedule

A preliminary budget showing: component costs (from actual vendor quotes, not estimates), fabrication costs (3D printing, PCB, machining), procurement shipping, and a 15–20% contingency. A project schedule (Gantt chart or sprint plan) with milestones, dependencies, and critical path identified. Demonstrate that the selected concept can be designed, built, and tested within the remaining project timeline.

21.2.8 Team Organization

Role assignments with clear ownership: who is responsible for each subsystem, who leads integration, who owns V&V, who manages procurement. A communication plan: meeting cadence, documentation repository, change control process.

21.3 Evaluation Criteria

Criterion	What Reviewers Look For	Ref
Problem clarity	Solution-agnostic? All five elements present?	Ch. 1
Stakeholder completeness	All nine categories checked? Power/interest grid populated?	Ch. 1
Requirements quality	One SHALL per req? Quantitative? Testable? Implementation-free?	Ch. 2
Traceability	Every req traces to a need? Every need has a req?	Ch. 2
Concept breadth	≥ 20 concepts before screening? Morph chart complete?	Ch. 3
Selection rigor	Trade study criteria from reqs? Sensitivity analysis done?	Ch. 3
Risk identification	High-severity failures flagged? Red-zone risks have mitigation?	Ch. 3
Test plan completeness	Every req has TAID method and test description?	Ch. 4

Table 21.1: PDR evaluation criteria.

Common PDR Failures

(1) Problem statement is actually a solution statement. (2) Requirements are vague or untestable (“shall be reliable”). (3) Only one concept explored; no evidence of design space exploration. (4) Trade study weights clearly biased to get the “right” answer. (5) No risk register or feasibility assessment. (6) BOM is a wish list with no prices, lead times, or vendors.

The PDR Litmus Test

Could a different team, given only your PDR documents, understand the problem, agree with your concept selection, and begin detailed design without asking you questions? If yes, you pass. If they would need to ask “what did you mean by...”; you are not ready.

21.4 The PDR Presentation: Structure and Delivery

A PDR presentation typically runs 20–30 minutes plus 10–15 minutes for questions. Structure the presentation to tell a coherent story:

Opening (2 min): state the problem, the user, and the impact. Make the review board care about the problem before presenting the solution. Use a compelling statistic, user quote, or scenario.

Context (3 min): competitive landscape, existing solutions, and why they are inadequate. This establishes the gap your product fills.

Value proposition (3 min): the four-dimension assessment (Chapter 2). Demonstrate that the project is feasible, desirable, viable, and sustainable. This is the “should we build this?” argument.

Requirements (5 min): present the SDRD. Focus on the top 10 requirements that define the product’s core performance. Show traceability from stakeholder needs to shall-statements. The full SDRD is in the appendix; do not read every requirement aloud.

Architecture and design concept (8 min): the heart of the presentation. Show the system block diagram, the mechanical concept (CAD renderings), the electrical concept (block-level schematic), and the software concept (state machine). For each, explain the key design decisions and why this approach was chosen.

Risk and mitigation (3 min): the top 5 risks with mitigation plans. If you have a risk-reduction prototype (e.g., a proof-of-concept for the most uncertain technical element), demonstrate or show results.

Plan forward (3 min): Gantt chart, budget, and team responsibilities. Show that the remaining work is planned, resourced, and scheduled realistically.

21.4.1 Handling Questions

Questions from the review board are not attacks; they are the board’s way of stress-testing your design thinking. For each question: (1) restate it to confirm understanding, (2) answer directly with data if available, (3) if you do not know, say so honestly and commit to a follow-up date, (4) do not argue; if the board identifies a real concern, acknowledge it and describe how you will address it.

21.5 PDR Deliverables Checklist

The following documents must be submitted with the PDR presentation:

- System Design Requirements Document (SDRD) – complete, numbered, with verification methods
- System architecture document – block diagram, interface definitions, functional decomposition
- Concept drawings – CAD renderings or detailed sketches of the mechanical design
- Electrical block diagram – all subsystems, power flow, signal flow, communication buses

- Software architecture – state machine diagram, data flow, user interface mockup
- Risk register – top 10 risks with probability, impact, and mitigation
- Preliminary test plan – VCRM draft with test methods for all critical requirements
- Project plan – Gantt chart, BOM estimate, budget summary, team assignments
- Value proposition summary – feasibility, desirability, viability, sustainability assessment

21.6 Scoring Rubric

PDR is typically graded on:

Technical depth (40%): quality of requirements, soundness of architecture, appropriateness of design decisions. Are the requirements testable? Is the architecture implementable? Do the design choices reflect engineering analysis, not guesswork?

Completeness (25%): are all required deliverables present and substantive? Missing sections receive zero credit. Placeholder sections (“TBD”) receive partial credit only if accompanied by a plan and date for completion.

Communication (20%): clarity of the presentation, quality of visual aids, ability to answer questions. Can the review board understand the design without reading the appendix?

Project management (15%): realism of the schedule, adequacy of the budget, clarity of team roles. Does the team have a credible plan for completing the project?

21.7 Practice Questions

1. Your team’s trade study shows scores of 3.81 vs. 3.80 between two concepts. The reviewer challenges your selection. How do you respond?
2. A team presents a PDR with 15 requirements, all verified by “Test.” The reviewer asks why none are verified by Analysis, Inspection, or Demonstration. What is wrong?
3. A team passes PDR but then struggles through CDR because their requirements were actually solutions in disguise (“shall use an Arduino Mega”). How should the PDR reviewer have caught this, and what specific requirement quality test would have flagged it?
4. Your competitive analysis found no existing product that solves your problem. Does this make your project more or less risky? Explain.

21.7.1 Answers

1. A 0.01 margin is within noise. Perform sensitivity analysis: vary each weight by ± 0.05 and check if the winner changes. If it does, the selection is not robust. Consider prototyping both finalists or identifying the single criterion that differentiates them and gathering better data on that criterion.
2. Assigning “Test” to everything suggests the team did not think carefully about methods. Physical presence/labeling should be Inspection. Thermal or structural margin should be Analysis. User-observable behavior should be Demonstration. Over-reliance on Test also inflates the test schedule.
3. The PDR reviewer should apply the “implementation-free” test from the requirements quality checklist: does this requirement specify WHAT (a capability) or HOW (a specific component)?

“Shall use Arduino Mega” specifies HOW. The correct requirement: “The controller shall provide ≥ 12 analog inputs, ≥ 4 serial interfaces, ≥ 256 KB flash, and support Wi-Fi connectivity.” This is testable, implementation-free, and allows the trade study to objectively select the best platform.

4. More risky: either the problem is not real (no one else thinks it needs solving), the problem is too hard (others tried and failed), or your search was inadequate. Investigate which explanation applies. If the problem is real and unsolved, the project has high value but also high technical risk.

Chapter 22

Critical Design Review (CDR)

When to Read This Chapter

Sprint 3: Read this chapter one week before CDR. Use the deliverables checklist (Figure 22.1) to verify completeness. **Before CDR:** Run the litmus test: “Could a technician build this from these documents alone?”

The Critical Design Review is the second formal design gate, following PDR. Where PDR confirmed that the team selected a feasible concept, CDR confirms that the concept has been developed into a complete, buildable design.

The CDR answers: “**Will this design work?**” It evaluates whether the detailed design is complete, manufacturable, and verifiable. CDR is the gate between design and fabrication; passing CDR authorizes the team to build.

22.1 Purpose and Scope

CDR confirms (see Figure 22.1 for the full checklist) that every component is specified, every interface is defined, every manufacturing step is planned, and every requirement has a path to verification. The design is frozen after CDR; changes require formal change control.

22.2 Deliverables

22.2.1 Finalized Technical Design

Complete schematics with all component values, ratings, and part numbers. PCB layout (if applicable) reviewed and ready for fabrication. CAD models of all custom mechanical parts with engineering drawings including dimensions, tolerances, material callouts, and surface finishes. For each fabrication method: specific parameters (e.g., “FDM, PLA, 0.2 mm layer, 40% infill” or “CNC 6061-T6 per Order of Operations”).

22.2.2 Design Analysis and Verification

Analysis results for all A-method requirements: FEA stress/thermal, worst-case circuit analysis, power budget with actual component data (not estimates), link budget for wireless communications, and motor/transmission torque calculations with margin. Every analysis must show: assumption, calculation, result, margin, and verdict.

CDR Deliverables Checklist

Technical Design <ul style="list-style-type: none"> ☐ Complete schematics w/ part numbers ☐ PCB layout reviewed (if applicable) ☐ CAD models + eng. drawings ☐ Wiring diagrams w/ wire gauges ☐ Firmware architecture + pin map 	Procurement & Mfg <ul style="list-style-type: none"> ☐ BOM finalized w/ vendor PNs ☐ Long-lead items ordered ☐ Manufacturing plan + schedule ☐ Plan B components identified ☐ Assembly sequence documented
Analysis <ul style="list-style-type: none"> ☐ Power budget (actual data, 25% margin) ☐ Motor/transmission torque calcs ☐ Worst-case circuit analysis 	V&V Updates <ul style="list-style-type: none"> ☐ VCRM updated for design changes ☐ Test procedures reviewed ☐ Test equipment confirmed

Litmus Test: Could a technician build the prototype from these documents alone?

Figure 22.1: CDR deliverables checklist. The litmus test: if a competent technician who has never seen this project could build the prototype from your CDR package alone, the package is complete.

22.2.3 Manufacturing and Procurement

Finalized BOM with: actual part numbers, verified vendor availability, confirmed pricing, and realistic lead times. All long-lead items ordered. Manufacturing plan: what is being fabricated (PCBs ordered, 3D prints queued, machining scheduled), what is being purchased, and assembly sequence.

The Spring Break Procurement Trap

If CDR falls before spring break, all procurement orders must be placed at CDR, not “after break.” Parts ordered the week before break arrive during break with no one to receive them. International orders (JLCPCB, AliExpress) need 3–4 weeks. McMaster ships next-day.

22.2.4 Updated Test Plan

VCRM updated with any new or modified requirements since PDR. All test procedures reviewed and updated based on the finalized design. Test equipment identified and availability confirmed. Test schedule with specific dates and responsible engineers. Subsystem test results from early prototyping (if available).

22.2.5 Software and Firmware

If your system includes firmware: source code under version control (Git), state machine diagram, pin mapping table, communication protocol definitions, and build/flash instructions. Code must compile and flash from the documented instructions without “tribal knowledge.” Include a hardware abstraction layer (HAL) that isolates sensor-specific code from application logic.

22.2.6 Thermal and Structural Analysis

For any component operating near its thermal limit: thermal analysis showing junction temperature under worst-case conditions with adequate margin. For structural components: FEA or hand calculations showing stress/deflection under design loads with safety factor ≥ 2.0 . Document: assumptions, model, results, margin, and verdict for each analysis.

22.2.7 Integration and Test Readiness

Integration plan: sequence of subsystem combinations with test gates at each stage. Interface verification plan: which interfaces will be tested at each stage and what constitutes pass/fail. Any subsystem demonstrations already completed (with data). Hardware/software integration strategy: how firmware will be loaded, tested, and version-controlled.

22.2.8 Risk Register Update

All PDR risks updated with current status. New risks identified during detailed design added. Top-5 risks have active mitigation plans with assigned owners and deadlines.

22.2.9 Procurement Status

For every BOM item: ordered/received/backordered status. Any substitutions from the PDR BOM documented with engineering justification. Lead time tracking for critical-path items. Budget actual vs. planned with variance explanation.

22.3 Evaluation Criteria

Criterion	What Reviewers Look For	Ref
Design completeness	Can this be built from CDR package alone? All parts specified?	Ch. 2–3
Drawings quality	Fully dimensioned? Toleranced? Material and finish specified?	Ch. 3
DFM compliance	Polymer parts pass IM checklist? Draft, wall, ribs correct?	Ch. 17
Analysis evidence	Calculations/simulations show reqs met with margin?	Ch. 2, 4
FMEA maturity	Updated for detailed design? High-RPN resolved? Safety hazards mitigated?	Ch. 3, 6
BOM completeness	Every component listed, priced, Plan B for critical parts? Within budget?	Ch. 3
Requirements currency	Changes since PDR documented? RTM current?	Ch. 2
Test readiness	Critical procedures written? Equipment identified?	Ch. 4, 5

Table 22.1: CDR evaluation criteria.

How CDR Differs From PDR

CDR requires a completed, fabrication-ready design—not just a feasible concept. At PDR, the team demonstrated that the selected concept *should* work (feasibility-level analysis, preliminary architecture, concept sketches). At CDR, every component is specified by part number, every interface is defined with pinouts and wire gauges, every custom part has dimensioned

and toleranced drawings, and every analysis uses actual component data rather than estimates. PDR earns the right to spend money on design; CDR earns the right to build.

Common CDR Failures

Incomplete drawings: CAD exists but engineering drawings with tolerances and manufacturing notes do not. **BOM gaps:** fasteners, wire, consumables missing; budget surprises. **No Plan B:** sole-source component goes out of stock with no alternative. **DFM ignored:** polymer parts have zero draft and non-uniform walls. **Stale FMEA:** PDR version never updated. **Late purchase requests:** ordered at CDR deadline instead of immediately after approval.

The CDR Litmus Test

Could a contract manufacturer build your system from only the CDR documents (BOM, schematics, drawings, assembly instructions) without calling you? If yes, your documentation is complete. If they would need to ask “what connector goes here?” or “what tolerance is this?”; you are not ready.

22.4 The CDR-to-Fabrication Transition

After CDR approval: (1) place orders for all BOM items, (2) begin fabrication of mechanical parts (3D printing, machining, laser cutting), (3) begin PCB fabrication and component procurement if using custom PCBs, (4) begin firmware development in parallel with hardware fabrication, (5) prepare the test station (calibrate instruments, build test fixtures). The goal is to have all subsystems ready for integration within 2–3 weeks of CDR.

22.5 CDR Presentation Structure

CDR builds on PDR but shifts emphasis from “what we plan to build” to “exactly what we will build.” The presentation should be 25–35 minutes:

Requirements update (3 min): any changes since PDR, with rationale. Highlight requirements that were added, deleted, or modified based on PDR feedback or design evolution.

Detailed design walkthrough (15 min): this is the core of CDR. Walk through the design at component level:

Mechanical: show fully dimensioned assembly and part drawings. Demonstrate that parts fit together (interference check screenshots). Identify the manufacturing process for each part and verify it is achievable. Call out critical tolerances and explain how they will be verified during fabrication.

Electrical: show complete schematics (not block diagrams). Identify every component by part number. Show the PCB layout (if applicable) or the wiring diagram (if hand-wired). Explain the power budget: total current at each voltage rail vs. supply capacity.

Software: show the complete state machine with all transitions and actions. Show the control algorithm with gains and update rate. Show the user interface layout. Demonstrate any prototype firmware running on the actual MCU.

BOM and procurement (3 min): present the complete BOM with costs. Identify long-lead items (already ordered or in stock). Confirm total cost is within budget.

Assembly plan (3 min): the sequence for building the system, including any fixtures, jigs, or special tools required. Identify quality checkpoints where intermediate testing is performed.

Updated test plan (3 min): the VCRM is now complete. All test procedures are written. Test equipment is identified (owned, borrowed, or to be purchased).

Updated risks and schedule (3 min): risks re-evaluated in light of the detailed design. Schedule updated with fabrication, assembly, and testing milestones.

22.6 CDR Technical Depth Expectations

At CDR, the review board expects:

Thermal analysis: if the system generates heat (power electronics, motors, heaters), show a thermal analysis demonstrating that no component exceeds its maximum operating temperature. This can be a simple hand calculation (Chapter 18 in MEGN 300) or a thermal simulation.

Structural analysis: if the system bears loads (mounting brackets, frames, enclosures), show a stress analysis demonstrating that no component exceeds its yield strength with an adequate safety factor (≥ 2 for student projects). A hand calculation using beam bending or FEA screenshot is acceptable.

Power budget: tabulate every load on each power rail. Sum the currents. Verify the power supply can provide the total with $\geq 20\%$ margin. Include worst-case scenarios (all motors stalled, all heaters on, Wi-Fi transmitting simultaneously).

Timing analysis: if the system has real-time requirements (control loop, sensor acquisition, communication), show that the MCU can execute all tasks within the available time. List each task's execution time and verify the total fits within the loop period.

22.7 CDR Deliverables Checklist

- Updated SDRD with change log
- Detailed mechanical drawings – fully dimensioned, toleranced, with material and finish callouts
- Complete electrical schematics – component-level with part numbers and values
- PCB layout files or wiring diagram – with connector pinouts and wire gauge specifications
- Firmware architecture document – state machine, pseudocode for critical algorithms, pin map
- Complete BOM – with part numbers, suppliers, quantities, costs, and lead times
- Assembly procedure – step-by-step with quality checkpoints
- Complete VCRM – all requirements mapped to test methods with written test procedures
- Power budget – all loads tabulated by voltage rail
- Thermal and structural analyses (if applicable)
- Updated risk register and Gantt chart

22.8 Common CDR Deficiencies

(1) **Schematics are block diagrams:** at CDR, every wire must be specified. “Sensor connects to MCU” is not a schematic; “LM35 Vout (pin 2) connects to ESP32 GPIO36 via 100Ω series resistor with 100 nF capacitor to ground” is.

(2) **BOM is incomplete:** missing quantities, missing part numbers, or costs from an unreliable source. Every part must be sourceable from a real distributor (Digi-Key, Mouser, McMaster) with a current price.

(3) **No power budget:** the team cannot state the total current draw on each rail. This means they do not know if the power supply is adequate, which is a fundamental design question.

(4) **Test procedures are vague:** “test the sensor accuracy” is not a procedure. “Apply 50 °C reference, record 20 readings, compute mean and std dev, compare to requirement SYS-ACC-001” is.

(5) **Schedule has no slack:** every task is back-to-back with no buffer for delays. Professional project managers add 20–30% contingency to each task duration.

22.9 Practice Questions

1. Your CDR BOM lists “12V DC Motor” with a link to an Amazon product page. The reviewer rejects this. Why?
2. A team’s power budget shows 2.8 A total load on a 3.0 A power supply. The reviewer flags this. Why is 93% utilization a problem?
3. Your team’s schematics show a 3.3 V regulator powering the ESP32, but the thermal analysis for the regulator was never performed. The ESP32 draws 240 mA during Wi-Fi transmission. Is this a CDR pass or fail? What analysis should have been done?
4. Your PCB design uses a QFN-32 package that requires reflow soldering. Your team only has hand-soldering capability. What should you do?

22.9.1 Answers

1. Amazon listings change without notice (price, availability, even the actual product). A proper BOM entry needs: manufacturer name, manufacturer part number, key specifications (voltage, RPM, torque, current), and a stable vendor with a datasheet. Amazon is a marketplace, not a specification source.
2. 93% utilization leaves only 0.2 A margin. Motor inrush alone could draw 6–10× running current for 50–100 ms. The supply will current-limit or brown-out, causing MCU resets. Design rule: 20–25% current margin minimum. This team needs a larger supply or must reduce the load.
3. CDR fail; missing analysis evidence. Required thermal analysis: input voltage (e.g., 5 V from USB), output 3.3 V at 240 mA peak. LDO power dissipation $P = (V_{in} - V_{out}) \times I = (5.0 - 3.3) \times 0.24 = 0.41$ W. At thermal resistance $\theta_{JA} \approx 150$ °C/W for SOT-223, junction temp rise $= 0.41 \times 150 = 61.5$ °C. At 25 °C ambient: $T_J = 86.5$ °C; below the 125 °C limit but with only 38.5 °C margin. At 40 °C ambient: $T_J = 101.5$ °C; still acceptable but worth documenting. The analysis takes 5 minutes and prevents a field failure.
4. Options: (a) select a TSSOP or SOIC alternative of the same IC, (b) order a QFN breakout board/adaptor, (c) use the JLCPCB assembly service (SMT placement for \$8). Option (a) is best if available. Never plan to hand-solder QFN; it requires a hot air station at minimum and even then has high defect rates.

Chapter 23

Final Design Review (FDR)

When to Read This Chapter

Sprint 7: Read this chapter two weeks before FDR. The deliverables checklist (Figure 23.1) is your final audit tool. **Before FDR:** Run the litmus test: “Could someone operate, maintain, and retire this system using only your documentation?”

The Final Design Review is the third and final design gate, following CDR. Where CDR confirmed that the detailed design was complete and ready to build, FDR evaluates the built system against its original requirements with measured evidence.

The FDR answers: “**How well did it work?**” It evaluates the completed system against its original requirements, documents what was built versus what was designed, captures lessons learned, and delivers the operations and maintenance package.

23.1 Purpose and Scope

FDR is the capstone review (see Figure 23.1 for the full checklist). It demonstrates that the system meets (or acknowledges deviation from) every requirement, provides complete as-built documentation, and delivers everything the stakeholder needs to operate and maintain the system independently.

23.2 Deliverables

23.2.1 System Verification and Validation Results

Complete VCRM with every requirement showing PASS, FAIL (with Discrepancy Report), or WAIVED (with stakeholder-signed waiver and risk acceptance). For every test: actual measured data (not just “pass”), number of repetitions, measurement uncertainty, and test conditions. For any FAIL: root cause analysis, attempted corrective actions, and current status. Validation evidence: stakeholder demonstration that the system solves the original problem in realistic operating conditions.

23.2.2 Final Design Documentation

As-built BOM (what was actually used, not what was planned). As-built schematics reflecting any changes made during integration. As-built CAD models and drawings reflecting any modifications. Firmware source code with version tag and build instructions. All design files archived in a team repository with README.

FDR Deliverables Checklist

V&V Evidence	O&M Package
☐ VCRM 100% complete	☐ Quick-start guide (1 page)
☐ Test data for every requirement	☐ Operating manual
☐ Root cause for any FAIL/PARTIAL	☐ Maintenance manual + schedule
☐ Stakeholder validation evidence	☐ Troubleshooting guide
	☐ Spare parts list
As-Built Documentation	Process & Reflection
☐ As-built BOM (actual, not planned)	☐ Decision log + change log
☐ As-built schematics + CAD	☐ Risk register (final status)
☐ As-designed vs. as-built table	☐ Lessons learned (specific, not generic)
☐ Firmware with version tag	

Litmus Test: Could someone operate, maintain, and eventually retire this system using only your FDR package?

Figure 23.1: FDR deliverables checklist. The litmus test: if someone could operate, maintain, troubleshoot, and eventually retire the system using only your FDR documentation, your package is complete.

23.2.3 As-Designed vs. As-Built Comparison

A table comparing the CDR design intent with the actual implementation. Every deviation must be documented with: what changed, why it changed, what impact the change had on requirements, and whether the change was approved under change control. This comparison is the audit trail that demonstrates design discipline.

23.2.4 Design Process Documentation

Decision log: key decisions made during the project with rationale and alternatives considered. Change log: all changes after CDR baseline with approval records. Risk register: final status of all identified risks (mitigated, accepted, or realized). Lessons learned: what worked, what did not, and what the team would do differently.

23.2.5 Operations and Maintenance Materials

Quick-Start Guide (1 page, 10-minute setup), Operating Manual (complete reference), Maintenance Manual (procedures with photos), Troubleshooting Guide (symptom-based flowcharts), spare parts list with vendors and lead times, and training materials.

23.2.6 External Team Testing Results

Results from having another team execute at least one of your test procedures. Document: which procedure was tested, what ambiguities the external team identified, what procedure updates resulted, and whether the external team achieved the same results. This is one of the most valuable FDR deliverables; it proves your documentation is truly reproducible.

23.2.7 Demonstration Video

A 3–5 minute video showing the system operating in its intended environment (or a representative simulation). The video should demonstrate: power-on sequence, normal operation, at least one sensor reading verified against a reference, actuator response to a trigger condition, and communication/alerting function. This video serves as permanent evidence of system functionality.

23.2.8 As-Designed vs. As-Built Comparison: Template

Item	As-Designed (CDR)	As-Built (FDR)	Approved?
MCU	ESP32-WROOM-32	ESP32-WROOM-32	N/A (same)
Temp sensor	DS18B20	BME280	CR-003, approved
Enclosure material	ABS (injection mold)	PETG (3D print)	CR-007, approved
Relay	SRD-05VDC	SRD-12VDC	CR-009, not approved

Table 23.1: As-designed vs. as-built comparison template.

Every deviation must have a Change Request (CR) number, engineering justification, impact assessment on requirements, and approval status. The unapproved relay change (12 V coil on a 5 V rail) is a design error that must be resolved.

23.2.9 Lessons Learned: Beyond “Start Earlier”

Good lessons learned are specific, actionable, and traceable to evidence:

- **Bad:** “We should have communicated better.” (Vague, no action.)
- **Good:** “I2C pull-up ownership was undefined in the ICD, causing 8 hours of debugging. Action: every ICD must explicitly state which side provides pull-ups, termination, and bias voltages.” (Specific problem, specific fix.)
- **Bad:** “We should have started procurement earlier.” (Generic.)
- **Good:** “We ordered the cellular module from AliExpress 3 weeks before CDR. It arrived 2 weeks late due to customs delay. Action: use McMaster-Carr (next-day) or DigiKey (3-day) for critical-path components; international vendors only for non-critical items with 4-week buffer.” (Specific, actionable.)

23.2.10 Reflection and Future Work

Honest assessment: did the system solve the original problem? What requirements were not met and why? What would a v2.0 address? What skills did the team develop? This section demonstrates engineering maturity; the ability to critically evaluate your own work.

23.2.11 Budget Final Accounting

Final cost accounting: planned vs. actual for every BOM category. Total project cost including labor hours (if tracked). Cost drivers identified: what was more expensive than expected and why. This is valuable data for future project planning.

23.2.12 Schedule Performance

Planned vs. actual milestone dates. Critical path analysis: was the critical path what you expected? Where did schedule slip occur and why? Sprint velocity analysis: did productivity improve over the project? This retrospective drives continuous improvement.

The FDR Litmus Test

Could a new team, given only your FDR package, operate, maintain, troubleshoot, and eventually improve your system without ever talking to you? If yes, your knowledge transfer is complete. If they would need to call you for “that one trick to get it to boot”; you have not finished.

23.3 Evaluation Criteria

Criterion	What Reviewers Look For	Ref
VCRM completeness	Every req has final status with evidence? No unaddressed items?	Ch. 4
Test rigor	Measured values reported (not just PASS/FAIL)? Repeated trials? Boundary tests?	Ch. 4
Failure honesty	Failures documented with root cause? Waivers justified? No hidden defects?	Ch. 4
As-built accuracy	Documents match what was actually built? BOM, drawings, wiring current?	Ch. 5
O&M readiness	Quick-start usable? Spare parts identified? Maintenance plan actionable?	Ch. 6
Design evolution	Journey from PDR → CDR → FDR documented with rationale?	All
Lessons learned	Specific and actionable? Demonstrate growth in engineering judgment?	Ch. 6
Budget accounting	As-designed vs. as-built cost? Variances explained?	Ch. 3

Table 23.2: FDR evaluation criteria.

How FDR Differs From CDR

FDR demonstrates a working system with verification evidence—not just a design on paper. At CDR, the team showed that the design *will* work (complete schematics, analysis with margin, fabrication-ready drawings). At FDR, the team presents measured test data proving the system *does* work, documents every deviation between the as-designed and as-built configurations, and delivers operations and maintenance materials sufficient for an independent operator. CDR is about design completeness; FDR is about verified performance and knowledge transfer.

Common FDR Failures

VCRM with gaps: requirements never tested, no explanation. **PASS without evidence:** “all requirements met” but no measured values or data. **Hidden failures:** known defects not reported. **Stale documentation:** as-built system has changes not reflected in drawings. **No O&M materials:** system cannot be operated by anyone except the build team. **Generic lessons learned:** “we learned to start earlier” without specific, actionable insights.

23.4 The Complete Review Progression

The three reviews form a narrative: PDR asks “Should it work?” (problem + concept), CDR asks “Will it work?” (design completeness), FDR asks “How well did it work?” (evidence + documentation). Each review builds on the previous. A strong PDR enables a strong CDR enables a strong FDR. Skipping or rushing an early review creates compounding problems downstream.

23.5 Lessons Learned Documentation

The Lessons Learned report is the most valuable deliverable for future teams. Document:

What went well: successful design decisions, effective tools and processes, good team practices. Be specific: “using Git for firmware version control prevented two integration disasters.”

What went wrong: failures, delays, surprises. Analyze root causes. “The PCB arrived with a footprint error because we did not verify the footprint library against the datasheet. Mitigation for future teams: always print the PCB layout at 1:1 scale and physically verify component fit before ordering.”

What we would do differently: with hindsight, what would you change? “We should have started PID tuning 2 weeks earlier. The first tuning attempt took 3 sessions; we budgeted one.”

Recommendations for the next team: specific, actionable advice. “The INA128 oscillates when driving more than 10 cm of cable. Add a 100 Ω series resistor at the output.” “The 3D-printed enclosure warps if printed without a brim. Use a 5 mm brim on all parts.”

23.6 Practice Questions

1. Your VCRM at FDR shows 18 PASS, 2 FAIL, and 1 PARTIAL PASS out of 21 requirements. How do you present this to the stakeholder?
2. A team’s FDR as-built BOM has 8 components that differ from the CDR BOM, but the change log is empty. What does this indicate?
3. Your stakeholder says “the system works great but I wish the display were bigger.” Is this a verification failure or a validation failure? Explain.

23.6.1 Answers

1. Present the complete VCRM transparently. For the 18 PASS: evidence summaries. For the 2 FAIL: root cause analysis, corrective actions attempted, and recommendation (fix, waive, or defer). For the PARTIAL PASS: which criteria passed, which failed, risk assessment, and recommendation. The stakeholder decides on open items.
2. No change control was exercised. The team made ad-hoc substitutions during build without documenting why, assessing impact, or seeking approval. This is a process failure that undermines confidence in the entire build. Any undocumented change could have invalidated a test result.
3. Validation failure. The system met all written requirements (verification passed). But the stakeholder’s actual need for readability was not captured as a requirement (e.g., “display shall be readable at 2m distance”). The requirements were incomplete; a Chapter 1 failure that propagated to FDR.

Industry SE Parallels

In industry, the three-review gate structure maps to the standard SE review cycle: **PDR** corresponds to the System Requirements Review (SRR) and Preliminary Design Review. **CDR** corresponds to the Critical Design Review and Test Readiness Review (TRR). **FDR** corresponds to the System Verification Review (SVR) and Operational Readiness Review (ORR). The artifact maturity progression is consistent with INCOSE SEH Sec. 4.9 (Technical Reviews) and NASA SP-2016-6105 Chapter 6. The “should work / will work / how well did it work” framing is widely used in defense acquisition (DoD 5000 series) and commercial product development.

Glossary of Acronyms and Key Terms

ABET	Accreditation Board for Engineering and Technology
ADC	Analog-to-Digital Converter
BLE	Bluetooth Low Energy
BLDC	Brushless Direct Current (motor type)
BOM	Bill of Materials; complete list of every component
BSPP	British Standard Pipe Parallel (thread standard)
BSPT	British Standard Pipe Taper (thread standard)
CAN	Controller Area Network (communication bus)
CDR	Critical Design Review: “Will this design work?”
ConOps	Concept of Operations; stakeholder-facing system description
COTS	Commercial Off-The-Shelf (product)
DFA	Design for Assembly
DFM	Design for Manufacturability
DR	Discrepancy Report
EMC/EMI	Electromagnetic Compatibility / Interference
ESC	Electronic Speed Controller (for BLDC motors)
FDR	Final Design Review: “How well did the design work?”
FEA	Finite Element Analysis
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Modes, Effects, and Criticality Analysis
FDM	Fused Deposition Modeling (3D printing)
GPIO	General Purpose Input/Output
I2C	Inter-Integrated Circuit (2-wire serial bus)
ICD	Interface Control Document
INCSE	International Council on Systems Engineering
IP (rating)	Ingress Protection (e.g., IP67 = dust-tight, 1 m submersion)
LDO	Low Drop-Out (voltage regulator)

MTBF	Mean Time Between Failures
MTTR	Mean Time To Repair
MCU	Microcontroller Unit
MoSCoW	Must / Should / Could / Won't (prioritization)
NEMA	National Electrical Manufacturers Association (motor frame sizes)
NPT	National Pipe Thread (tapered, U.S. standard)
NPS	Nominal Pipe Size
PDR	Preliminary Design Review: "Should this design work?"
PWM	Pulse Width Modulation
RPN	Risk Priority Number (Severity $\times$ Occurrence $\times$ Detection)
RTM	Requirements Traceability Matrix; links each requirement to its parent need and forward to its verification method. Tracks <i>what</i> must be verified.
SDRD	System Design Requirements Document
SPI (comms)	Serial Peripheral Interface (4-wire bus)
SPI (mold)	Society of the Plastics Industry (mold finish standards)
TAID	Test, Analysis, Inspection, Demonstration (verification methods)
THT	Through-Hole Technology (component mounting)
SMD/SMT	Surface-Mount Device / Technology
TVS	Transient Voltage Suppressor (protection diode)
UART	Universal Asynchronous Receiver/Transmitter
UNC	Unified National Coarse (thread standard)
UNF	Unified National Fine (thread standard)
V&V	Verification and Validation
VCRM	Verification Cross-Reference Matrix; extends the RTM by recording test status (Planned/In Progress/PASS/FAIL/PARTIAL) at each review milestone. Tracks <i>whether</i> each requirement has been verified.

Exam Quick-Reference Cheat Sheet

Requirements

- SHALL = mandatory; SHOULD = goal
- One shall per requirement
- Quantitative, testable, implementation-free
- TAID: Test, Analysis, Inspection, Demonstration

V&V

- Verification: built right (meets specs)
- Validation: right system (meets needs)
- VCRM tracks every req to its test status
- PASS/FAIL/PARTIAL. Always with data

Motors

- $\tau = K_t \times I$; $V_{back} = K_e \times \omega$
- Stall current = $V/R_{winding}$ (6–10× rated)
- Design at 50–75% of continuous torque
- Every motor needs a matched driver

Power

- LDO: low noise, $V_{out} < V_{in}$, η = ratio
- Buck: efficient step-down, EMI risk
- Boost: step-up from low voltage
- 20% current margin minimum

Fasteners

- $T = K \cdot F \cdot d$ (torque-preload)
- Thread engagement: $1.5d$ steel, $2d$ aluminum

- Always specify grade in BOM
- Nyloc or Loctite for vibration; not split washers

Bearings

- Default: deep groove ball (6000 series)
- $L_{10} = (C/P)^3 \times 10^6 / (60n)$
- 2RS = sealed. Never hammer.
- Press fit inner ring; sliding fit outer ring

Transmission

- $GR = N_2/N_1 = \omega_1/\omega_2 = \tau_2/\tau_1$
- $P = \tau \times \omega$; $\eta_{total} = \eta_1 \times \eta_2 \times \dots$
- Self-locking: lead screws, worm gears
- Student defaults: GT2 belt, T8 lead screw

Injection Molding DFM

- Uniform wall $\pm 10\%$; draft $\geq 1^\circ$
- Rib = 60% wall; boss OD = $2 \times$ hole ID
- Inside radius $\geq 0.5 \times$ wall
- Living hinge: PP only, 0.25–0.5 mm

PCB Layout

- 100 nF decoupling within 5 mm of every IC
- Ground pour on bottom layer
- Print 1:1 and place real components
- 10 mil trace = 0.3 A; 100 mil = 3.0 A

Why Projects Fail. The Top 20

1. Problem statement is actually a solution statement.
2. Missing stakeholders discovered after PDR.
3. Requirements are vague, untestable, or compound.
4. Only one concept explored; no design space coverage.
5. Trade study weights manipulated to get the “right” answer.
6. No competitive analysis; reinventing an existing product.
7. BOM has no part numbers, no vendors, no prices.
8. Procurement delayed until after spring break.
9. Motor sized at stall torque instead of continuous.
10. Power supply sized for average current, not inrush peak.
11. I2C pull-up ownership undefined in ICD.
12. Integration done “big-bang” (everything at once).
13. Testing only at room temperature.
14. Test procedures that only the author can execute.
15. VCRM updated only at FDR (should evolve from PDR).
16. No configuration management; “which firmware version?”
17. O&M manual not written until the night before FDR.
18. As-built documentation does not match the physical system.
19. Change log empty despite 8 BOM substitutions.
20. No lessons learned or honest reflection on what went wrong.

Each of these failures has been observed in actual MEGN 301 projects. They are listed in approximate chronological order; early failures (1–6) cause more damage because they compound through the project lifecycle.

The cost of late discovery: A problem discovered at PDR costs 1 hour to fix (rewrite a requirement). The same problem discovered at CDR costs 1 week (redesign a subsystem). At FDR, it costs the entire grade (the system does not work and there is no time to fix it). This is the “1-10-100 rule” of systems engineering: fix problems at the earliest possible stage.

#	Failure	Caught At	Fix Cost
1	Solution masquerading as problem	PDR	1 hour
5	Trade study weights manipulated	PDR	2 hours
8	Procurement delayed past spring break	CDR	2–3 weeks
11	I2C pull-ups undefined	Integration	8 hours
12	Big-bang integration	Integration	1–2 weeks
17	O&M manual not written	FDR	2 days (rushed)

Table 23.3: Representative fix costs for common project failures.

ABET Student Outcome Mapping

MEGN 301 addresses the following ABET Student Outcomes for mechanical engineering:

SO	Description	MEGN 301 Coverage
1	Identify, formulate, and solve complex engineering problems	Ch. 1–3 (problem definition through concept selection)
2	Apply engineering design to produce solutions meeting specified needs	Ch. 2–6 (requirements through O&M), Part III (design reviews)
3	Communicate effectively with a range of audiences	Part III (PDR/CDR/FDR presentations and documentation)
5	Function on a team, establish goals, plan tasks, meet deadlines	Sprint structure, design review gates
6	Develop and conduct appropriate experimentation, analyze data	Ch. 4 (V&V and test plans)
7	Acquire and apply new knowledge as needed	Ch. 7–12 (technical reference applied to design projects)

Table 23.4: ABET Student Outcome mapping for MEGN 301.

GreenShield: Complete Worked Example

This appendix traces the GreenShield Automated Greenhouse Frost Protection System through every phase of the design-build process, consolidating the chapter-by-chapter GS boxes into a single narrative.

Phase 1: Problem Definition (Chapter 1)

Problem: Small-farm greenhouse operators [WHO] cannot maintain above-freezing temperatures in unheated hoop houses during Colorado Front Range winters [WHERE], resulting in crop losses averaging \$12,000/season [WHY], within a \$500 budget using existing 120 VAC infrastructure [CONSTRAINTS].

Stakeholders: Primary user (operator), beneficiary (crops/farm revenue), sponsor (operator/farm), maintainer (operator or farm hand), regulator (NEC electrical code, county building code), integrator (existing greenhouse structure), trainer (operator's spouse or employee).

ConOps: System monitors air temperature continuously. When temperature drops below a user-adjustable setpoint (default 3 °C), it activates a supplemental heater and sends an SMS alert. In degraded mode (communication failure), it continues autonomous operation. The operator can adjust setpoint and check status remotely.

Phase 2: Requirements & Architecture (Chapter 2)

Key Requirements:

1. SYS-REQ-001: Monitor temp at ≥ 1 Hz, accuracy ± 0.5 °C, range -20 to $+60$ °C. [T]
2. SYS-REQ-002: Activate heater when temp $\leq$ user setpoint. Hysteresis: 2 °C. [T]
3. SYS-REQ-003: SMS alert within 60 s of heater activation. [T]
4. SYS-REQ-004: Autonomous mode on comm failure. [D]
5. SYS-REQ-005: Operate on 120 VAC, < 50 W standby. [T]
6. SYS-REQ-006: Enclosure IP44 minimum (splash-resistant). [I]
7. SYS-REQ-007: All field connections tool-free (connectors, not solder). [I]

Architecture: Five subsystems. Sensor (DS18B20), Controller (ESP32), Actuator (relay + ceramic heater), Communication (SIM7000G LTE-M), Power (12 V adapter + buck converters).

Phase 3: Concept Generation & Selection (Chapter 3)

Three concepts evaluated via weighted decision matrix. Concept 1 (Grid-Connected, Wi-Fi) selected: lowest cost (\$85), simplest integration, meets all requirements. Off-Grid Solar concept deferred to v2.0.

Risk register: top risk = MCU brown-out during relay switching (RPN 12). Mitigation: 470 μ F decoupling, separate relay power trace.

FMEA top item: relay stuck closed (RPN 108). Mitigation: thermal fuse on heater as independent safety cutoff.

Phase 4: V&V Planning (Chapter 4)

VCRM established at PDR with 7 requirements, all methods assigned. Three test procedures drafted: TP-001 (temperature accuracy), TP-002 (heater activation threshold), TP-003 (SMS alert timing).

Test environment: ME department environmental chamber for boundary testing (0 °C and 50 °C). Ice bath as backup for 0 °C test point.

Phase 5: Integration & Debug (Chapter 5)

Integration sequence: (1) Power subsystem alone (verify 5 V and 3.3 V rails). (2) Add sensor (verify I2C communication). (3) Add relay (verify GPIO control without brown-out). (4) Add cellular module (verify SMS send). (5) System-level 48-hour burn-in.

Integration failure encountered: I2C communication intermittent. Root cause: missing 4.7 k Ω pull-up resistors; neither the sensor breakout nor the ESP32 DevKit provided them. ICD updated to specify pull-up ownership on the controller side.

Phase 6: Operations & Maintenance (Chapter 6)

Wear items: relay contacts (rated 100,000 cycles; at 10 cycles/night $\times$ 150 nights/year = 67 years), DS18B20 sensor (no wear items but verify annually against reference thermometer), SIM7000G antenna connector (inspect annually for corrosion).

O&M manual: Quick-start (1 page), operating manual (4 pages), maintenance schedule (1 page), troubleshooting flowchart (1 page), spare parts list with McMaster/DigiKey part numbers.

Final Results (FDR)

VCRM: 6 PASS, 1 PARTIAL PASS (SYS-REQ-001 accuracy was ± 0.6 °C at 0 °C boundary; missed by 0.1 °C). Root cause: DS18B20 factory calibration drift at low temperatures. Corrective action: polynomial correction applied in firmware, retest showed ± 0.4 °C across full range. Updated status: PASS.

Total cost: \$92 (\$8 under budget). Build time: 6 weeks. Lessons learned: (1) always define pull-up ownership in ICDs; (2) test at temperature boundaries early, not at FDR; (3) the 470 μ F capacitor fix for brown-out should have been in the original design, not discovered during integration.

Quick Reference Tables

These tables consolidate the most commonly needed design values. For full context, see the referenced chapters.

Fastener Grades and Proof Loads

Grade/Class	Tensile (MPa)	Proof (MPa)	Marking	Typical Use
SAE Gr. 2 / ISO 4.6	400	225	None / “4.6”	General purpose
SAE Gr. 5 / ISO 8.8	830 / 800	585 / 640	3 lines / “8.8”	Structural
SAE Gr. 8 / ISO 10.9	1035 / 1040	895 / 940	6 lines / “10.9”	High strength
ISO 12.9	1220	1100	“12.9”	Socket head caps

Thread Engagement by Material

Steel: 1.5*d* min. Aluminum: 2.0*d*. Cast iron: 2.0*d*. Plastic: 2.5*d* (consider threaded insert).

PCB Trace Width vs. Current (1-oz External Copper, IPC-2221)

Trace (mil)	10	20	50	100	200
Safe Current (A)	0.3	0.7	1.5	3.0	6.0

Wire Gauge Reference (Chassis Wiring, Single Conductor)

AWG	Dia. (mm)	Max (A)	Common Use
30	0.25	0.5	Wire-wrap, breadboard jumpers
26	0.40	1.3	Ribbon cable, signal wiring
22	0.64	3.0	General hookup, sensor cables
18	1.02	7.0	Motor power, relay wiring
14	1.63	15	High-current bus, battery cables

Injection Molding DFM Rules

IP Rating Quick Reference

IP20: finger-safe, no water. IP54: dust-protected, splash-resistant. IP65: dust-tight, water jets. IP67: dust-tight, 1 m submersion 30 min. IP68: continuous submersion per spec.

Parameter	Rule
Wall thickness (ABS/PC)	1.5–3.5 mm, uniform $\pm 10\%$
Wall thickness (PP)	0.8–3.0 mm
Draft angle	1° – 2° per side; add 1° per 0.001 in texture depth
Ribs	50–70% of wall thickness, height $\leq 3 \times$ wall
Bosses	OD = 2.0–2.5 $\times$ hole ID; connect to walls with ribs
Inside radii	$\geq 0.5 \times$ wall; outside = inside + wall

Key Formulas

Bearing life: $L_{10} = (C/P)^p \times 10^6 / (60n)$ hours [$p = 3$ ball, $10/3$ roller]

Bolt torque: $T = K \cdot F \cdot d$ [$K=0.20$ dry, 0.15 lube, 0.12 anti-seize]

Motor torque: $\tau = K_t \times I$ Back-EMF: $V_e = K_e \times \omega$

Gear ratio: $GR = N_2/N_1 = \omega_1/\omega_2 = \tau_2/\tau_1$

Clamp force: $F_{clamp} = P_{cavity} \times A_{projected}$

Availability: $A = MTBF / (MTBF + MTTR)$

The 15 Most Expensive Mistakes in Student Projects

This appendix consolidates the warning boxes scattered throughout the document into a single quick-reference, organized by the project phase where each mistake typically strikes. Use it as a self-assessment checklist at each sprint boundary.

Sprints 1–2: Problem Definition and Requirements

1. **Solution-as-problem statement.** “We need to build an Arduino-based system” specifies technology before the problem is understood. Problem statements must be solution-agnostic. (Ch. 1, p. 2)
2. **Vague requirements.** “Shall be fast,” “shall be user-friendly,” “shall work well.” Every requirement needs a number, a unit, and a tolerance. (Ch. 3, p. 20)

Sprint 3: Detailed Design and CDR

3. **Undersized power supply.** Sizing the supply for steady-state current, not motor inrush (which can be $5\text{--}10\times$ running current). Size for inrush with bulk capacitors and staggered startup. (Ch. 11, p. 112; Ch. 12, p. 125)
4. **Shared ground paths.** Motor return current flows through the sensor ground trace, corrupting ADC readings. Use star-point grounding with separate conductors for signal, power, and safety domains. (Ch. 12, p. 125)
5. **No flyback diodes on inductive loads.** Switching off a motor, relay, or solenoid without a flyback diode produces voltage spikes that destroy the driver. Every inductive load needs one. (Ch. 12, p. 125)
6. **Using ESP32 GPIO 6–11.** These are connected to internal flash. Using them crashes the board. Also: GPIO 34–39 are input-only; ADC2 is disabled when Wi-Fi is active. (Ch. 19, p. 199)
7. **Decoupling capacitors 25 mm from the IC.** They must be within 5 mm of the power pin. Trace inductance at 25 mm negates the capacitor’s benefit. (Ch. 18, p. 186)
8. **Loading a voltage divider.** The divider output voltage drops when the load resistance is comparable to the divider resistance. Rule: $R_L \geq 10 \times (R_1 \parallel R_2)$, or use a buffer. (Ch. 13, p. 138)
9. **Ordering parts the week before spring break.** Parts arrive during break with no one to receive them. International orders need 3–4 weeks. Place orders at CDR, not after. (Ch. 22, p. 224)
10. **Split lock washers for vibration resistance.** NASA testing shows they provide no meaningful locking. Use nyloc nuts, Loctite 242, or Nordlock wedge-lock washers. (Ch. 15, p. 153)

Sprints 4–5: Build, Integration, and Debugging

11. **Big-bang integration.** Assembling everything at once and powering on. Use bottom-up: one component at a time, test after each addition. (Ch. 6, p. 55; Ch. 20, p. 210)
12. **Skipping visual inspection before power-on.** A solder bridge between VCC and an I/O pin destroys the IC at power-on. Five minutes under 10× magnification prevents hours of replacement. (Ch. 18, p. 186)
13. **No pull-up resistor ownership in the ICD.** Both teams assume the other provides the I2C pull-ups. Neither does. Define pull-up ownership explicitly in every Interface Control Document. (Ch. 6, p. 55)
14. **Changing multiple variables while debugging.** Swapping a component AND changing firmware AND rewiring simultaneously. If it starts working, you do not know which change fixed it. Change one thing, test, record. (Ch. 20, p. 210)

Sprints 5–7: Testing and Verification

15. **Testing only at room temperature.** A requirement of 0–50 °C demands testing at 0 °C and 50 °C, not just 22 °C on a benchtop. (Ch. 5, p. 46)

References

- [1] INCOSE, *Systems Engineering Handbook*, 4th ed., Wiley, 2015.
- [2] NASA, “Systems Engineering Handbook,” SP-2016-6105 Rev 2, 2016.
- [3] B. S. Blanchard and W. J. Fabrycky, *Systems Engineering and Analysis*, 5th ed., Pearson, 2011.
- [4] A. Kossiakoff et al., *Systems Engineering Principles and Practice*, 3rd ed., Wiley, 2020.
- [5] K. Ulrich and S. Eppinger, *Product Design and Development*, 7th ed., McGraw-Hill, 2020.
- [6] G. Pahl and W. Beitz, *Engineering Design*, 3rd ed., Springer, 2007.
- [7] G. Kotonya and I. Sommerville, *Requirements Engineering*, Wiley, 1998.
- [8] S. Robertson and J. Robertson, *Mastering the Requirements Process*, 3rd ed., Addison-Wesley, 2012.
- [9] IEEE, “IEEE 1012: Standard for System, Software, and Hardware Verification and Validation,” 2016.
- [10] R. A. Malloy, *Plastic Part Design for Injection Molding*, 2nd ed., Hanser, 2010.
- [11] D. V. Rosato and M. G. Rosato, *Injection Molding Handbook*, 3rd ed., Springer, 2000.
- [12] B. S. Blanchard, *Logistics Engineering and Management*, 6th ed., Pearson, 2004.
- [13] R. S. Pressman, *Software Engineering: A Practitioner’s Approach*, 9th ed., McGraw-Hill, 2019.
- [14] IEC 60300, “Dependability Management,” International Electrotechnical Commission.
- [15] Protolabs, “Design Guide: Injection Molding,” protolabs.com, accessed 2025.
- [16] IPC, “IPC-2221B: Generic Standard on Printed Board Design,” 2012.
- [17] KiCad Project, “KiCad Documentation,” kicad.org, accessed 2025.
- [18] H. W. Ott, *Electromagnetic Compatibility Engineering*, Wiley, 2009.
- [19] M. I. Montrose, *PCB Design for Real-World EMI Control*, Wiley, 2000.
- [20] IPC, “IPC-2612: Sectional Requirements for Electronic Diagramming Documentation,” 2010. (Note: superseded by IPC-2612B, 2021.)
- [21] R. Khandpur, *Printed Circuit Boards: Design, Fabrication, Assembly and Testing*, McGraw-Hill, 2005.
- [22] Espressif Systems, “ESP32 Technical Reference Manual,” V5.1, 2023.
- [23] SKF Group, “Rolling Bearings Catalogue,” PUB BU/P1 10000/2 EN, 2018.

-
- [24] R. G. Budynas and J. K. Nisbett, *Shigley's Mechanical Engineering Design*, 11th ed., McGraw-Hill, 2020.
- [25] ASME, "B1.20.1: Pipe Threads, General Purpose (Inch)," American Society of Mechanical Engineers, 2013.
- [26] ASME, "B31.3: Process Piping," American Society of Mechanical Engineers, 2022.
- [27] IEC, "IEC 60529: Degrees of Protection Provided by Enclosures (IP Code)," International Electrotechnical Commission, 2013. (Note: consolidated edition IEC 60529:2013+AMD1:2024 CSV issued 2024.)
- [28] IEC, "IEC 61984: Connectors—Safety Requirements and Tests," International Electrotechnical Commission, 2008. (Note: current edition is IEC 61984:2008+AMD1:2012 CSV; Ed. 3 under revision.)
- [29] SAE International, "J514: Hydraulic Tube Fittings," SAE International, 2012.
- [30] MIL-DTL-38999, "Connectors, Electrical, Circular, Miniature, High Density, Quick Disconnect," U.S. Department of Defense.
- [31] FCC, "Title 47 CFR Part 15: Radio Frequency Devices," Federal Communications Commission.
- [32] IEC, "IEC 61000: Electromagnetic Compatibility (EMC)," International Electrotechnical Commission.
- [33] R. Fitzpatrick, *The Mom Test: How to Talk to Customers and Learn If Your Business Is a Good Idea When Everyone Is Lying to You*, Robfitz, 2013.
- [34] Espressif Systems, "ESP32 Datasheet," V4.4, 2024. Available: <https://www.espressif.com/en/products/socs/esp32>.
- [35] SIMCom, "SIM7000 Series Hardware Design," V1.05, 2022. Available: <https://www.simcom.com/product/SIM7000G.html>.
- [36] JLCPCB, "PCB Manufacturing Capabilities," 2025. Available: <https://jlcpcb.com/capabilities/pcb-capabilities>.

Index

A

ABET outcomes, 236
ADC (analog-to-digital converter), 29

B

back-EMF, 115
ball bearing, 159
bathtub curve, 73
bearing, 34
belt drive, 73
bill of materials (BOM), 39
binary search (debugging), 214
BMS (Battery Management System), 41
boundary testing, 51
brown-out, 37
brushless DC motor (BLDC), 115
BSP thread, 98
buck converter, 28

C

circular economy, 86
cold joint, 193
concept generation, 35
Concept of Operations (ConOps), 2
concept selection, 42
configuration management, 64
context diagram, 4
continuity check, 79
crimp termination, 91

D

decision matrix, 25
decoupling capacitor, 37
deep sleep (ESP32), 26
Design for Assembly (DFA), 104
Design for Disassembly, 70
Design for Manufacturability (DFM), 236
discrepancy report, 78
draft angle, 173
DRC (design rule check), 19

E

ejector pin, 172
end of life, 82

end of life (EOL), 14
ESD (electrostatic discharge), 129
ESP32, 26

F

fault isolation, 213
FCC Part 15, 3
FDR (Final Design Review), 21
feasibility assessment, 12
flyback diode, 117
FMEA (Failure Mode and Effects Analysis), 23
FMECA, 68
footprint verification, 188
fretting corrosion, 96
functional decomposition, 23

G

gate location (mold), 175
gear train, 164
Gerber files, 187
ground loop, 64
ground plane, 190
ground pour, 192

H

H-bridge, 115
handoff package, 63

I

I²C, 29
IDC (insulation displacement), 93
impedance, 60
injection molding, 13
inrush current, 129
integration planning, 58
Interface Control Document (ICD), 23
interface types, 59
interrupt (ISR), 130
IP rating, 3
IPC-2221, 187

J

JLCPCB, 188

K

KiCad, 188
Kirchhoff's Current Law (KCL), 139
Kirchhoff's Voltage Law (KVL), 139

L

LDO (Low Dropout Regulator), 28
lead screw, 123
lessons learned, 230
level shifter, 201
lithium battery safety, 15
living hinge, 174
logic analyzer, 62, 214

M

morphological chart, 33
MoSCoW prioritization, 3
MTBF (Mean Time Between Failures), 21
MTTR (Mean Time To Repair), 68
multimeter, 62, 79

N

non-conformance report, 78
NPT thread, 97
nyloc nut, 155

O

Ohm's Law, 138
operational amplifier (op-amp), 140
operational scenario, 6
oscilloscope, 52, 62
overcurrent protection, 43
overmolding, 180

P

PCB (printed circuit board), 13
PCB layout, 189
PDR (Preliminary Design Review), 3
PID controller, 203
planetary gear, 119
power budget, 119
power/interest grid, 9
preload (bolt), 153
prototyping, 39
Pugh matrix, 30
pull-up resistor, 58
PWM, 63

R

regression test, 78

relay, 28
reverse engineering, 34
reverse polarity protection, 44
RoHS Directive, 21
roller bearing, 159
RPN (Risk Priority Number), 36
rubber duck debugging, 214

S

schematic capture, 188
scope creep, 4
SDRD (System Design Requirements Document), 21
sensitivity analysis, 36
servo motor, 116
shaft coupling, 43
silkscreen, 186
sink mark, 172
snap-fit, 70
solder bridge, 190
solenoid, 112
spur gear, 164
stakeholder analysis, 2
star-point grounding, 126
state machine, 44
stepper motor, 132

T

TAID (Test, Analysis, Inspection, Demonstration), 46
thermal runaway, 84
Thevenin equivalent, 140
thread engagement, 157
thread galling, 98
thread-locking compound, 44
threaded insert, 157
timing belt, 132
tombstoned component, 196
torque, 43
total cost of ownership (TCO), 15
trace width (PCB), 186
trade study, 25
triple bottom line, 12
TVS diode, 114

U

UART, 29
UL (Underwriters Laboratories), 3
use case, 6

V

V-model, 50
Validation, 46
value proposition, 12
VCRM (Verification Cross-Reference
Matrix), 49
Verification, 46
via (PCB), 195

voltage divider, 139
voltage regulator, 192

W

wall thickness (injection molding), 100
watchdog timer, 44
WEEE Directive, 82
weld line, 176
worm gear, 167