Prototype to Product

Embedded Interface Design

with **Bruce Montgomery**

Learning Objectives

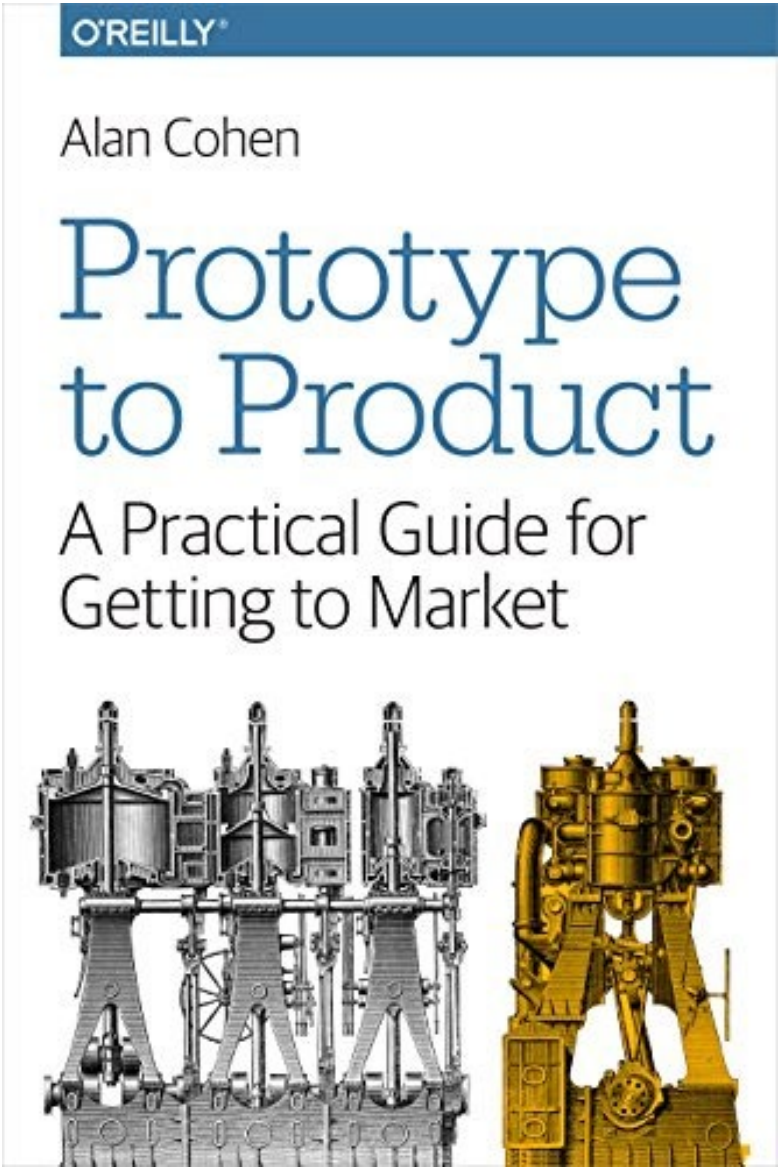
Students will be able to…

- Recognize principles and guidelines for creating embedded devices
- Consider the broad contributors to bringing a product to market

Prototype to Product - Cohen

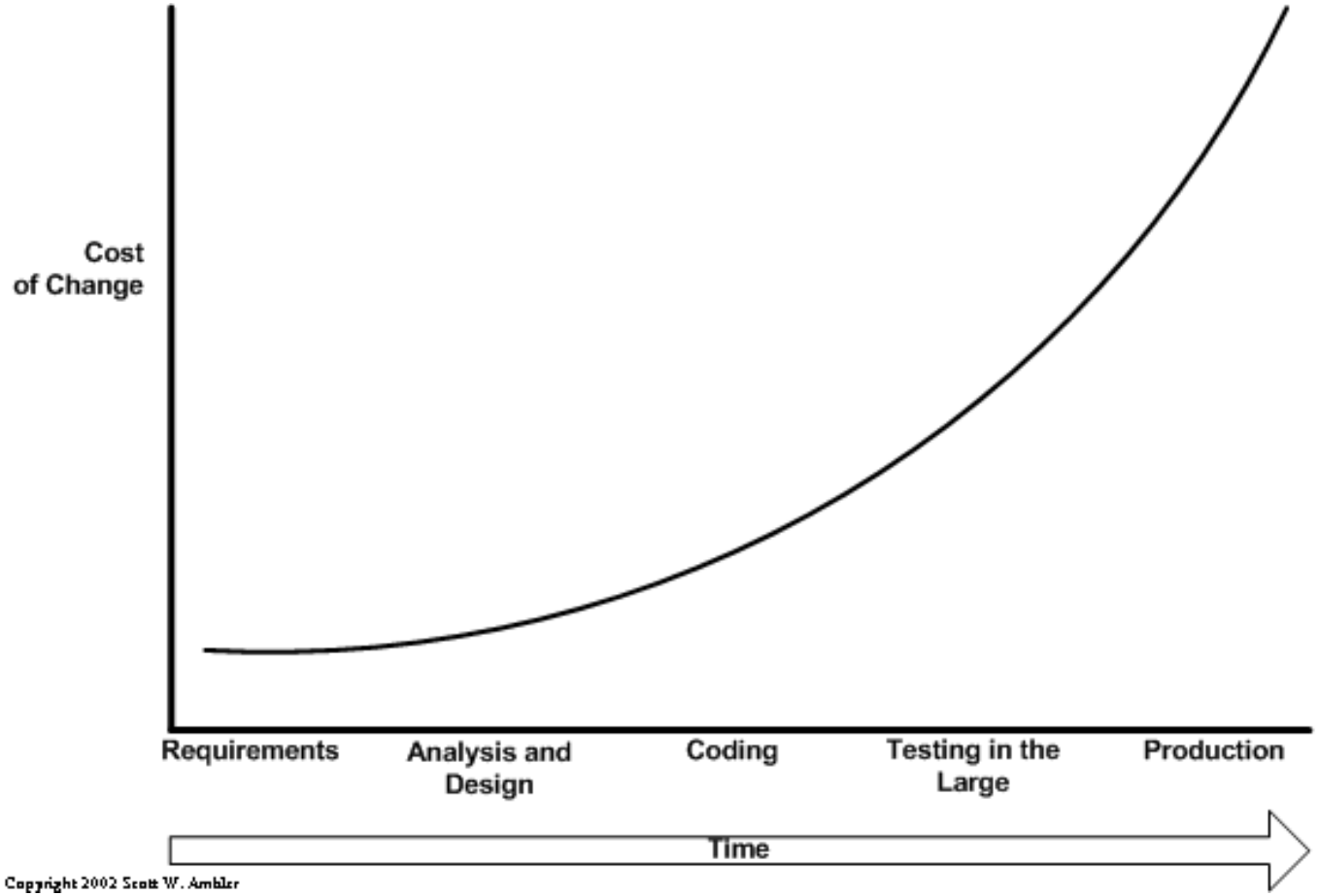
- Cohen, 2015, O'Reilly
- A practical guide for getting embedded devices to market
- Mistakes in product development
- Product functionality and cost
- Requirements
- Selecting system elements
- Complying with regulations and certification
- Development cycles

Reference [1],
similar details for PCB-based devices at [2]



Fundamental Principle of Product Development

- “Surprises only get more expensive if discovered later.”
- Putting it another way: “Product development is largely an exercise in uncovering surprises as soon as possible”
- Highlights need for early planning and discovery
- Reference [1], image [3]



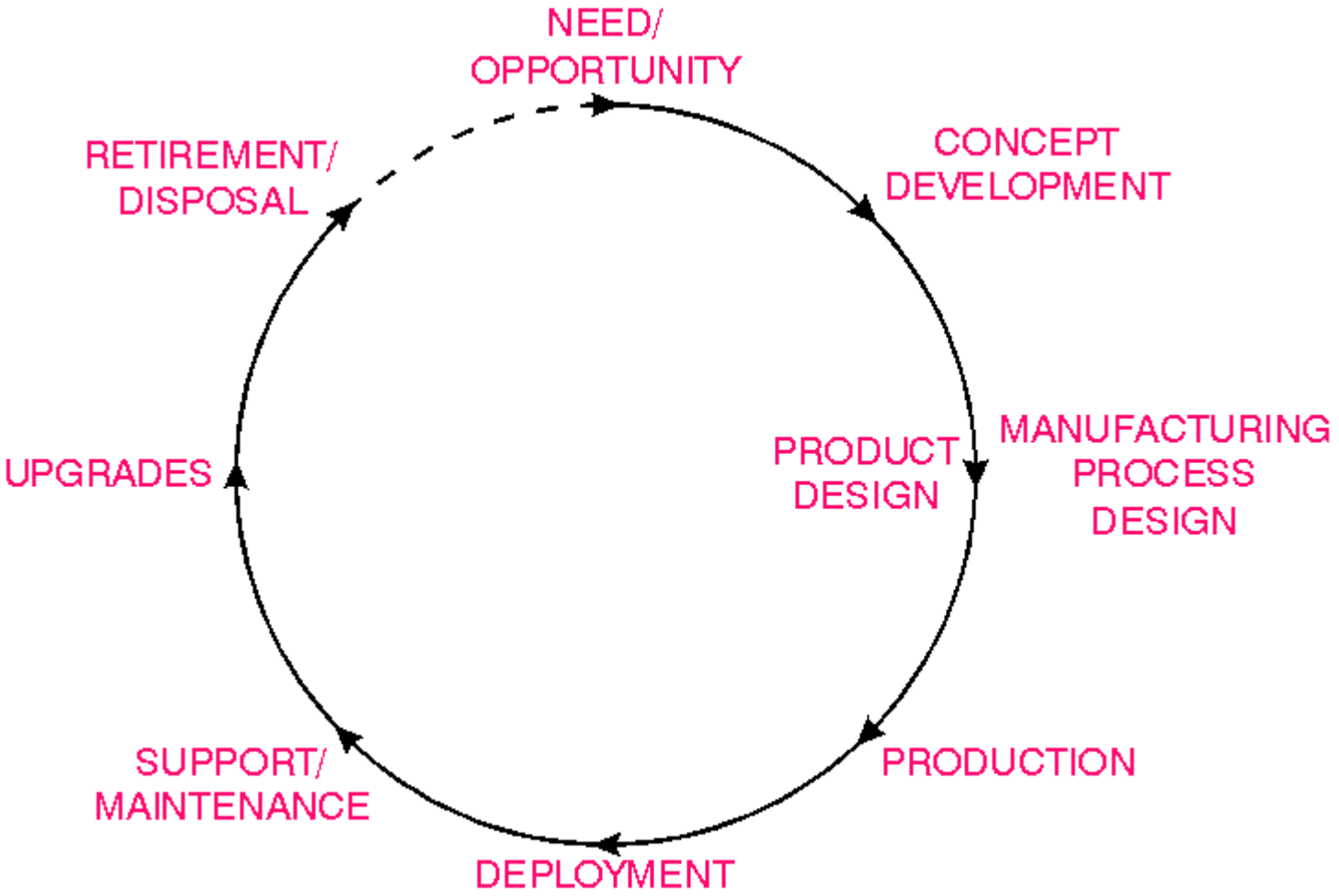
Deadly sins of product development

- Laziness: Putting off serious testing until end of development
- Assumption: Assuming you know what users want in a product
- Assumption: Assuming users know what they want in a product
- Fuzziness: Not having detailed requirements
- Fuzziness: Not having a detailed project plan
- Fuzziness: Not knowing who's responsible for what
- Cluelessness: Not addressing regulations and certification
- Perfectionism: New-feature-itis
- Perfectionism: Not knowing when to quit polishing
- Hubris: Not planning to fail
- Ego: Developing technology and not products

Reference [1]

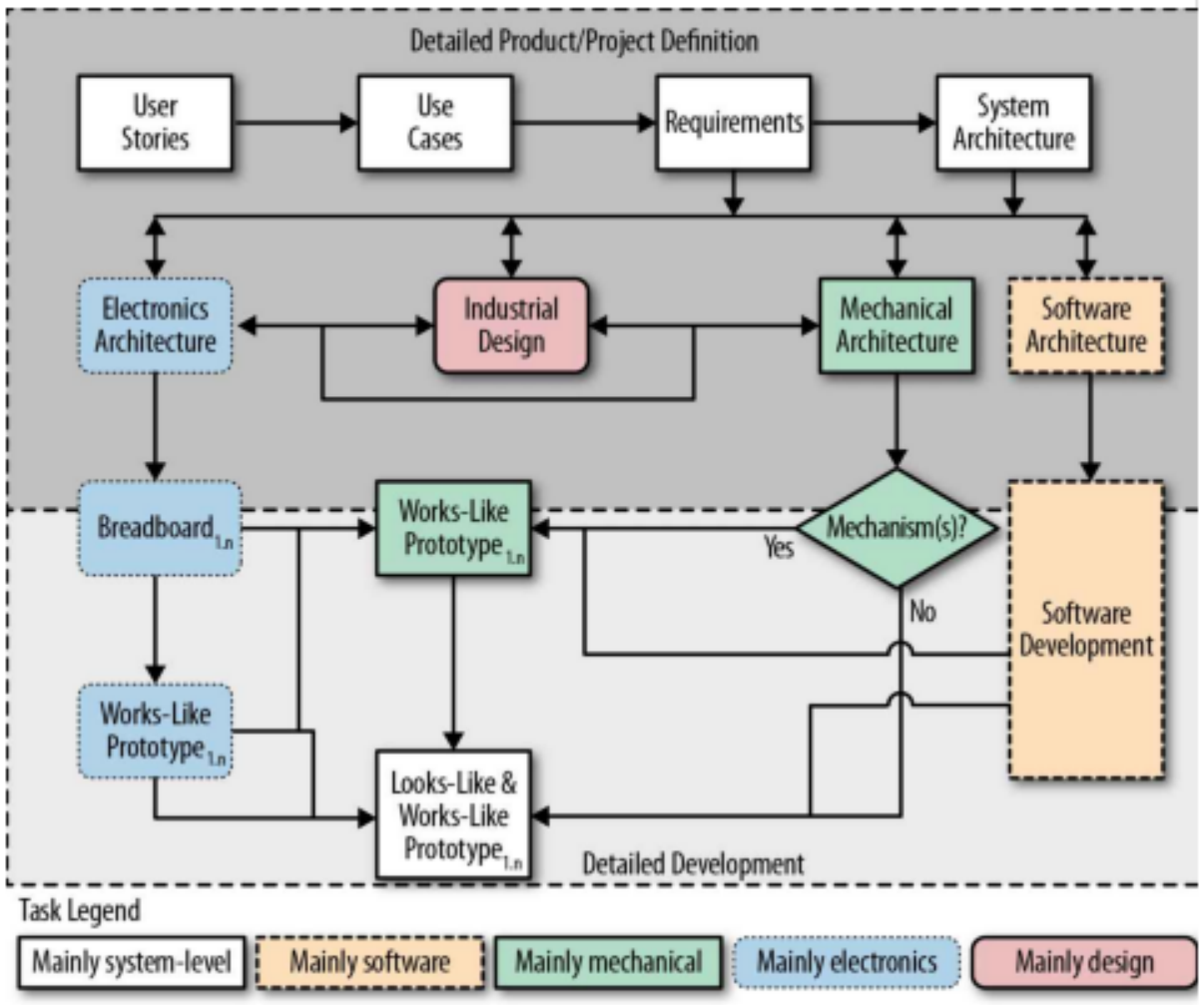
Typical Development Process

- Ideation
- Prototyping
- Planning
- Detailed Design
- Development
- Manufacturing
- Pilot
- Production
- Phase Gates (Decision Poi
- Reference [1], Image [4]



Typical Detailed Product/Project Definition Process

- User stories, use cases -> requirements and architecture
- Detailed design:
  - Hardware
  - Mechanical
  - Software/Firmware
  - Regulatory
  - Test
- Reference [1]



Embedded Platform and OS Considerations

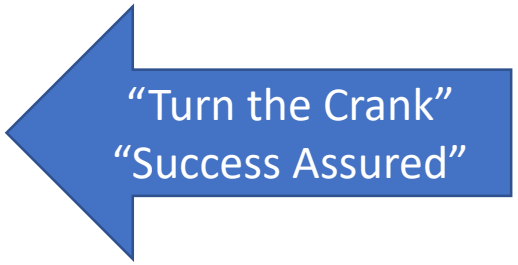
- Hardware
  - Low-end MCU (MSP-430)
  - Middle-size MCU (ARM)
  - Combined function microcontrollers -- SiLabs Gecko series (ARM + RF)
  - Big iron microcontrollers - Intel PC processors
  - Systems on Modules (SOMs)
  - Single-Board Computers (SBCs)
- Power: Batteries, Line Power
- OS Choices (see chart)
- Compliance
  - FCC, UL, CE, RoHS, REACH, etc.
- Reference [1]

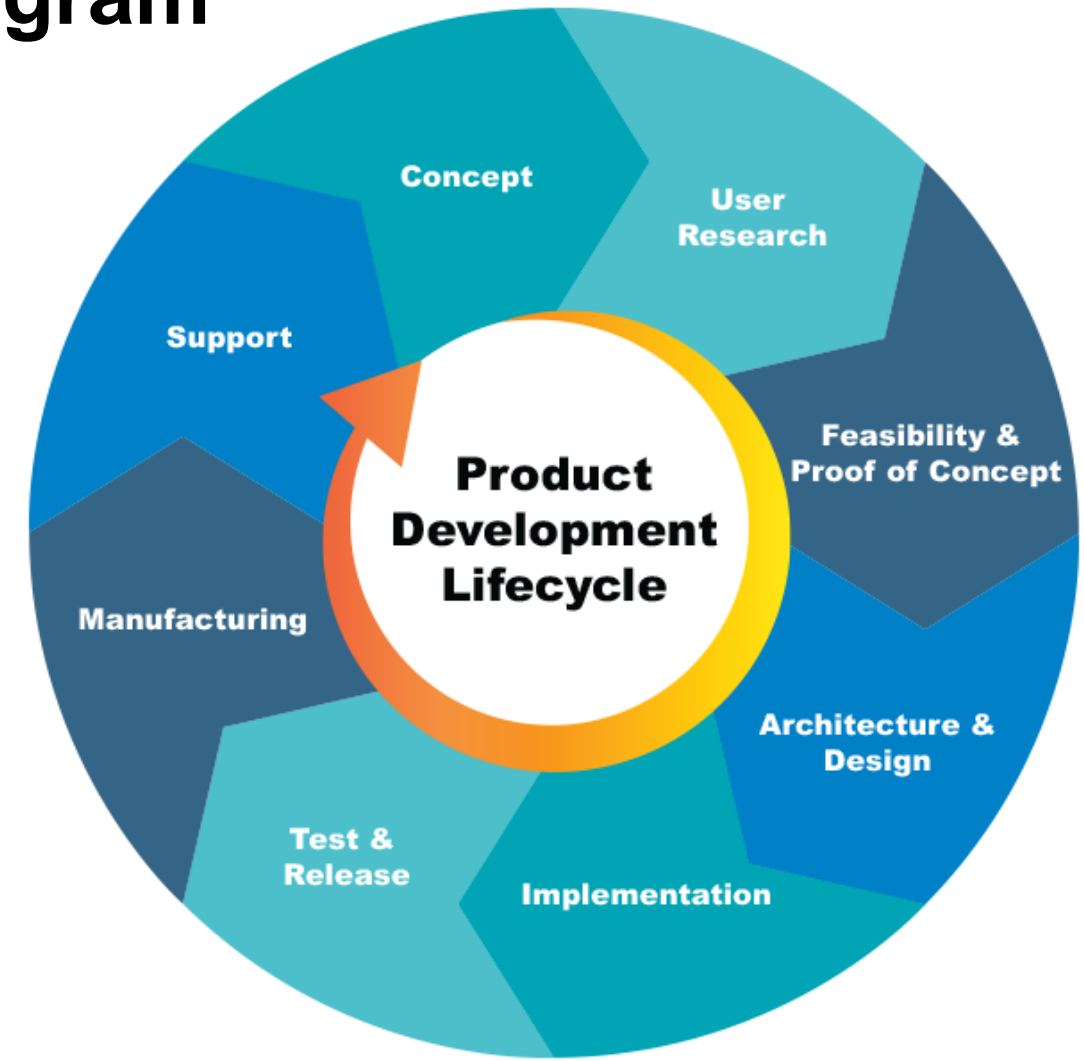
	No OS	RTOSes	Embedded Linux	Android	Desktop/Server Linux	Desktop/Serve Windows
Reliability (resistance to crashing)	Depends on code	Good to Excellent	Fair-Good	Fair	Fair	Fair
Interrupt response time	Depends on code	Good to Excellent	Fair	Fair	Poor	Poor
Flexible or sophisticated communications	Difficult to implement	Poor to Fair	Excellent	Good	Excellent	Excellent
Support other vendor peripherals	Difficult to implement	Poor	Fair	Fair	Fair	Excellent
Network security	Depends on code; usually good	Fair to Excellent	Fair to Good	Good	Fair-Good	Fair-Good
User security	Depends on code; usually excellent	Excellent	Fair to Excellent	Medium	Poor-Fair	Poor-Fair
Hardware cost (per unit manufactured)	Low	Low to High	Medium to High	Medium-High	High to Very High	High to Very High
Software cost (purchase)	Low	Low to High	Zero-Low	Zero	Zero to Low	Very High
Configuration effort	Low to High	Moderate-High	High	High	Low-High	Low-Moderate
Application development support	Poor to Good	Poor to Fair	Good	Good	Excellent	Excellent
GUI/touchscreen support	Poor	Poor to Good	Good	Excellent	Good to Excellent	Good to Excellen

Typical Development Flow Diagram

- Proof-of-concepts
  - Eliminate unknowns
- Prototypes
  - Increasing functionality and specificity
- Productions levels
  - Alpha
  - Beta
  - Pilot
  - Full Production

Reference [1], Image [5]

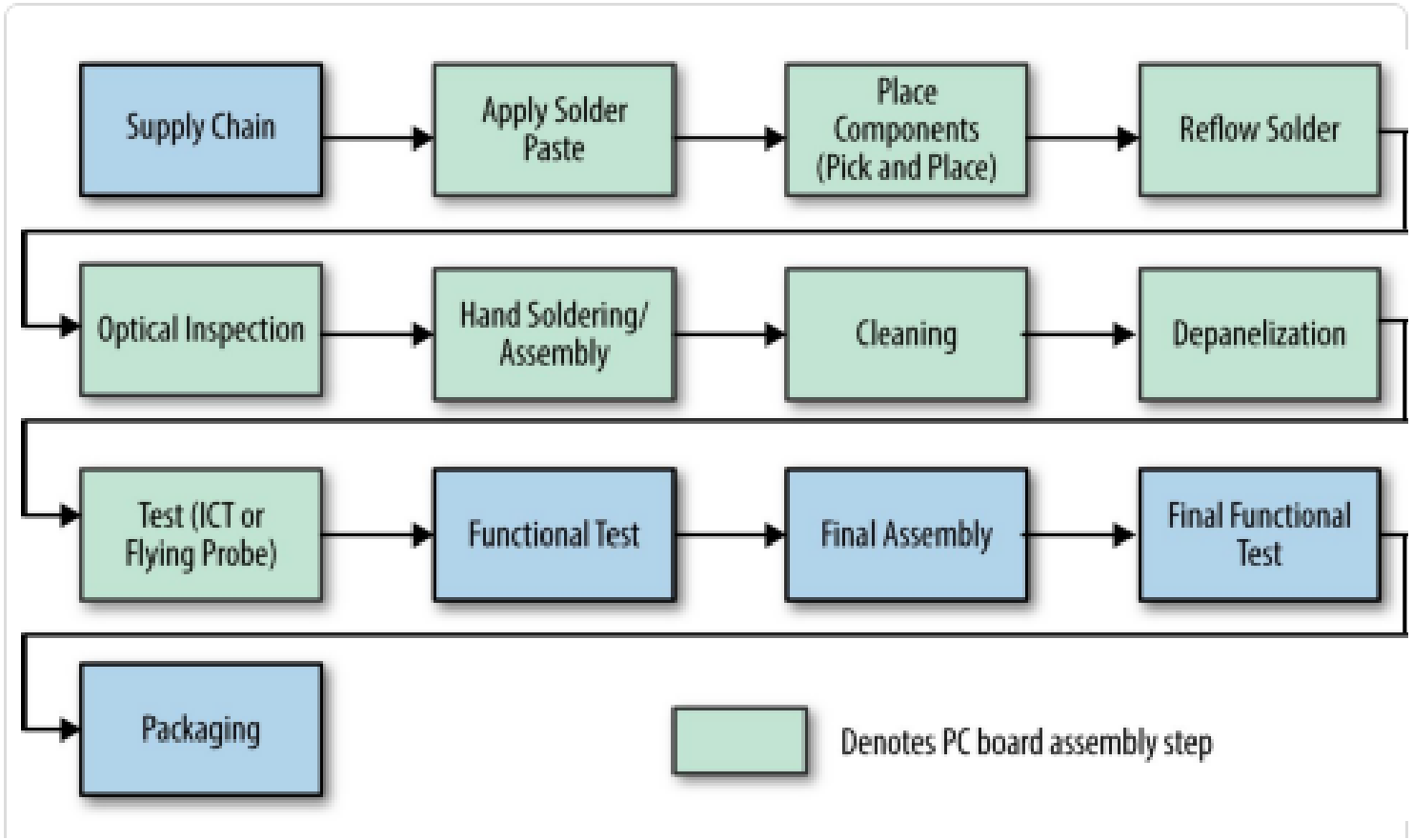




Typical Manufacturing Process

- Supply chain/ component assessment
- PCB assembly
- Test systems and firmware loads
- Packaging and labeling
- Automation/manual assembly
- Volume
- Cost of goods
  - Components
  - Labor/overhead
- Just the PCB… Mechanical, other sourcing, etc.

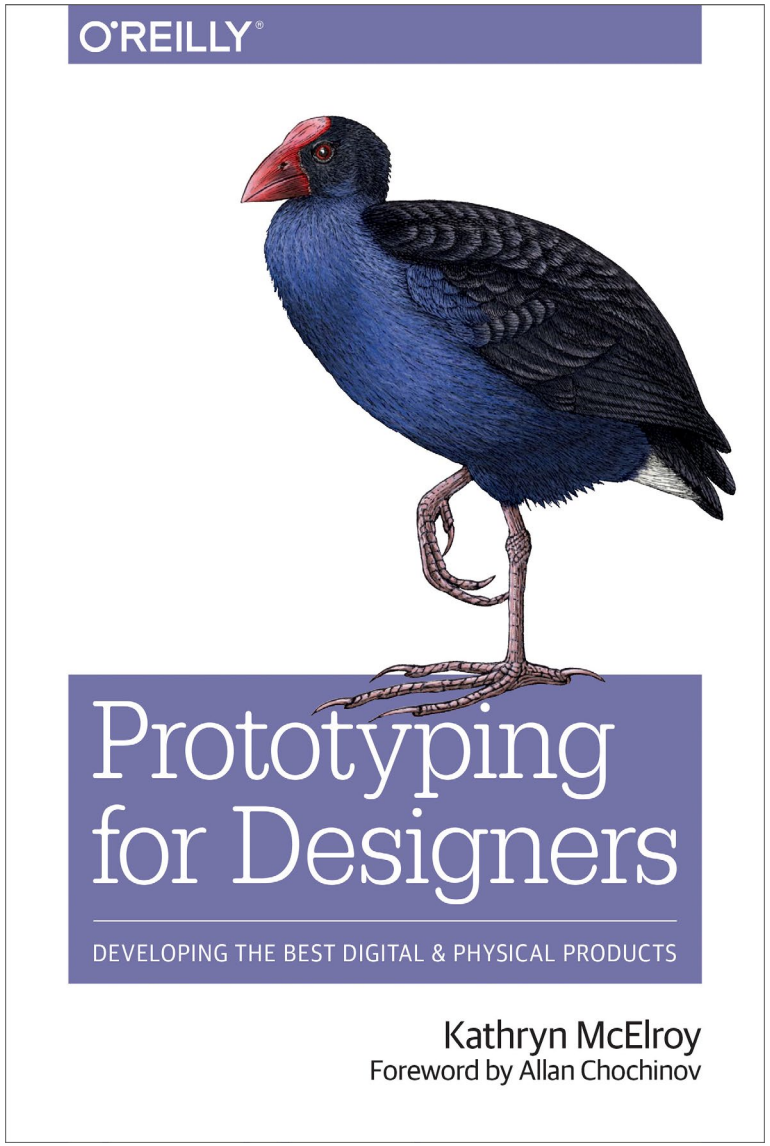
Reference [1]



Moving On…

- Next couple of lectures we'll look at the other class textbook:
- Prototyping for Designers – McElroy

- Review of prototyping best practices
- What is a prototype, why we prototype
- Fidelity
- Prototyping process
- Protos for Digital or Physical Products
- Testing with Users
- Prototyping Tools and Platforms



Next Steps

- REMEMBER – no class Monday – Alan Kay talk at Atlas – check in with me there for Quiz credit
- Project 5 starts today
- Project gear is mostly here
- New Quiz is up – another next weekend
- Class staff available to help
  - Shubham - Tues 12-2 PM, Fri 3-5 PM in ECEE 1B24
  - Sharanjeet - Tues 2-3 PM, Thur 2-3 PM in ECEE 1B24
  - Bruce - Tue 9:30-10:30 AM, Thur 1-2 PM in ECOT 242
- Final Exam is set
  - Tuesday Dec 17 7:30 PM - 10 PM ECCR 1B51
  - Final will be open notes and Canvas based, you'll need a PC

References

[1] Prototype to Product, Cohen, 2015, O'Reilly
[2] https://predictabledesigns.com/how-to-develop-and-prototype-a-new-product/
[3] http://www.agilemodeling.com/essays/costOfChange.htm
[4] https://users.ece.cmu.edu/~koopman/des_s99/life_cycle/index.html
[5] https://www.sunriselabs.com/Product-Development-Services/Product-Lifecycle-Support/