



Discoveries from the Agile Hardware Research Project

Kevin Thompson, PhD, PMP, ACP, CSP, CSM
kevin.thompson@cprime.com

107 S B St, Ste 200, San Mateo, CA 94401 | 650-931-1651 | www.cprime.com
The leader in training and consulting for Agile product development

Kevin Thompson, Ph.D.



Chief Scientist

- Trains, coaches teams and companies in Agile development
- Creator, Cprime Agile training content
- Author, *Solutions for Agile Governance in the Enterprise (Sage)*
- Education and certifications
 - **Certified Scrum Master and Scrum Professional**
 - **PMI Project Management Professional**
 - **PMI Agile Certified Practitioner**
 - **Scaled Agile Framework Program Consultant**
 - **Certified Yellow Belt Collaboration Architect**
 - **Doctorate in Physics, Princeton University**

Research Partners



John Carter

- Principal TCGen Inc.
- Cirrus Logic Board Member
- Former Chief Engineer (Bose)
- MS EE, MIT, BS Harvey Mudd



Scott Elliott

- Principal Techzecs
- R&D Director, HP
- Founder, Process Consulting HP
- BS,MS and PhD, UC Berkeley and Santa Barbara

The Research Project: 2013—2015

- Motivations and Key Questions
- What are effective techniques for
 - Agile development of hardware?
 - Concurrent Agile development of hardware and software?
- What are specific “recipes” for the above, using Cprime’s principles for Agile governance
 - Scrum, Kanban, something else...
- Capture what “best practices” exist and what gaps exist
- How organizations have effectively implemented such development

Companies and Methodology

- Nearly Twenty Companies – Responses from
- Amazon
- Applied Materials
- AT&T
- Beats
- Cisco
- Hitachi
- HP
- KLA Tencor
- Tandem
- Teradata

Methods

- Developed hypothesis on what could lead to agility in HW/Large programs
- Developed interview guide and construct data collection approach
- Constructed interview guide with key questions
- Gathered information via interviews, probing on emerging best practices
- Analyzed data, extracted common characteristics & outstanding practices
- Consolidated findings to create novel process model and recommend agile development organization



Research Findings

- Hardware companies used different Agile techniques
- None had a full Agile process
- Still emerging, and still experimenting, but some successes

- Several best practices emerged
- Not slave to identical Sprints, but adhere to identical durations
- Overlapping prototyping of circuit boards – not strictly serial
- Planning with ‘sticky notes on a wall’
- Hardware Sprints of longer than SW (typically 2x duration)
- Creative use of Burn down charts – repurposed for Hardware
- Burn down metrics change by Sprint
- Ranking of features by value

Key Differences Hardware Versus Software

- Flexibility is inherently less
 - Can't do an update over the web for physical product
- Cost of change is higher for hardware
- Architectural work is more front-loaded for hardware
- Availability of standard components constrains hardware design more than software design
- In software development, variation in type of work done over time is small
 - In hardware development, because of the realization of a physical product, the type of work done changes substantially over time
- Hardware development cost rises towards end of work, while software costs are flat over time
 - Hardware Sprints towards end have significant test costs

Key Similarities between Hardware and Software Development

- The total work of product development can be divided into a large number of small and testable deliverables
 - True in Software and Hardware
- Work can be partitioned over Sprints (short development cycles). Sprint scope can be estimated, progress can be measured, and learning can be integrated
 - Planning poker, Burn down charts, and Retrospectives common to both
- Self-contained, self-organized work teams can be created for even the largest projects
 - Creation of small, self sufficient, cross functional teams with Scrum Masters common to both
- Important distinction:
 - Software: Aggregation of deliverables yields usable features over time
 - Hardware: Aggregation of deliverables yields usable product at the end of development

Recommended: Scrum Process for Hardware Development

1. Story Types: “User” vs “Technical”
2. Sprint Length
3. Work estimation is Time-Based
4. Release Planning is Needed
5. Variation in Sprint Focus during Release Cycle

Story Types

- User Story: Short narrative description of a user experience. Usually written by Product Owners.
- Technical Story: Short description of deliverable that is not user-facing. Usually written by Team Members
- Both types occur in both software and hardware development
- Expect higher ratio of Technical Stories to User Stories in hardware than in software

Sprint Length for Hardware

- Usual Scrum advice is 2—4 weeks for Sprint length in software
- I advocate 2—3 weeks for hardware and software development
- Most hardware clients choose 2 weeks
- Lengthen if necessary to discover optimum length
- Use the same length for all Sprints across all collaborating teams

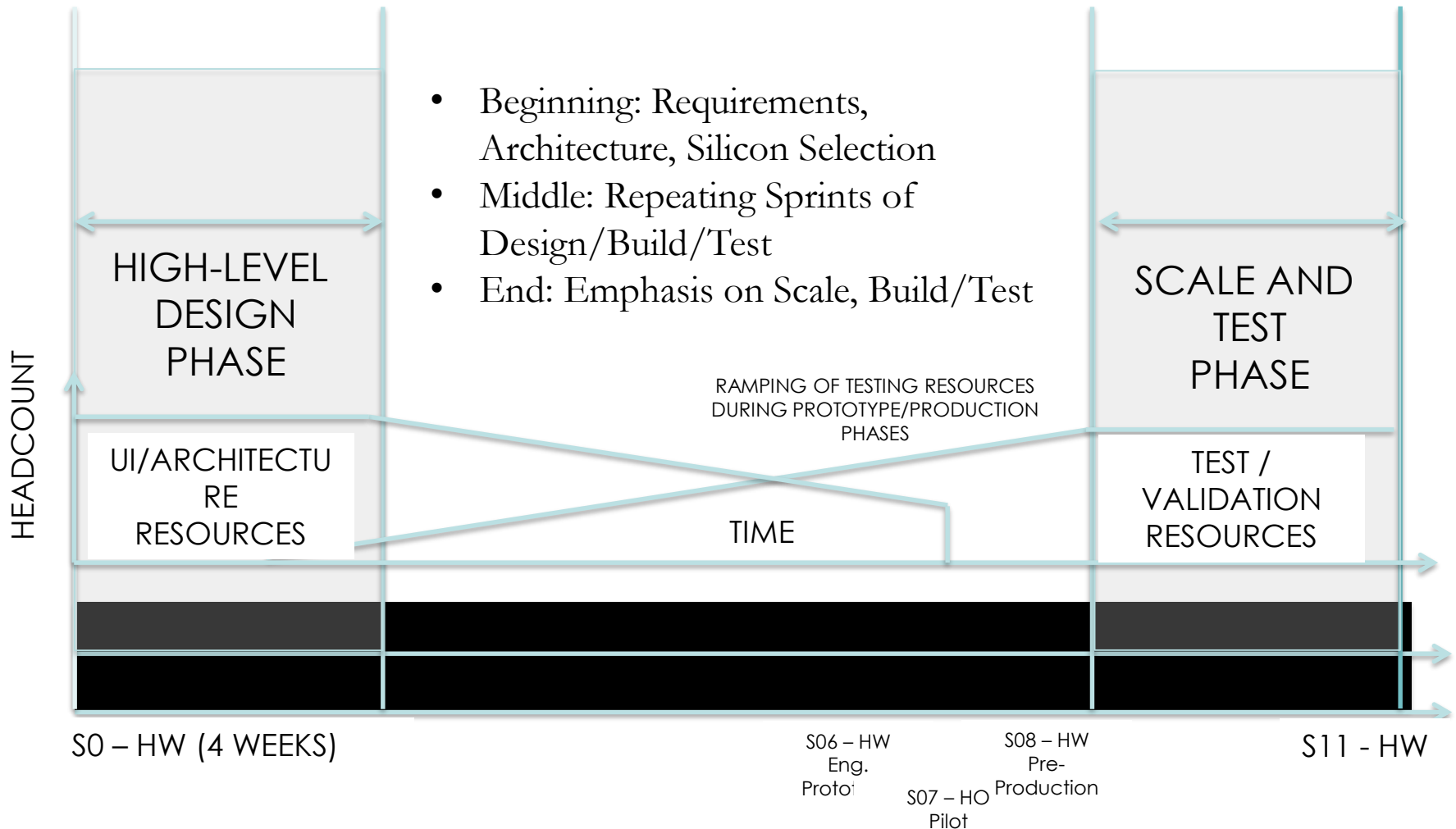
Work Estimates are Based on Time

- Use of relative sizing with Story Points is common in software. This requires Team be able to develop consensus norm of sizing that is not based on time.
- High degree of specialization of hardware Team skills (ME, EE, etc.) means consensus norm not based on time is not possible, because most Stories cannot be sufficiently understood by all Team members
- Time-based estimation (Person-Days, Man-Days, etc.) is understood by all Team members, and tends to be familiar to them already
- Task estimates are in hours (common in software, too)

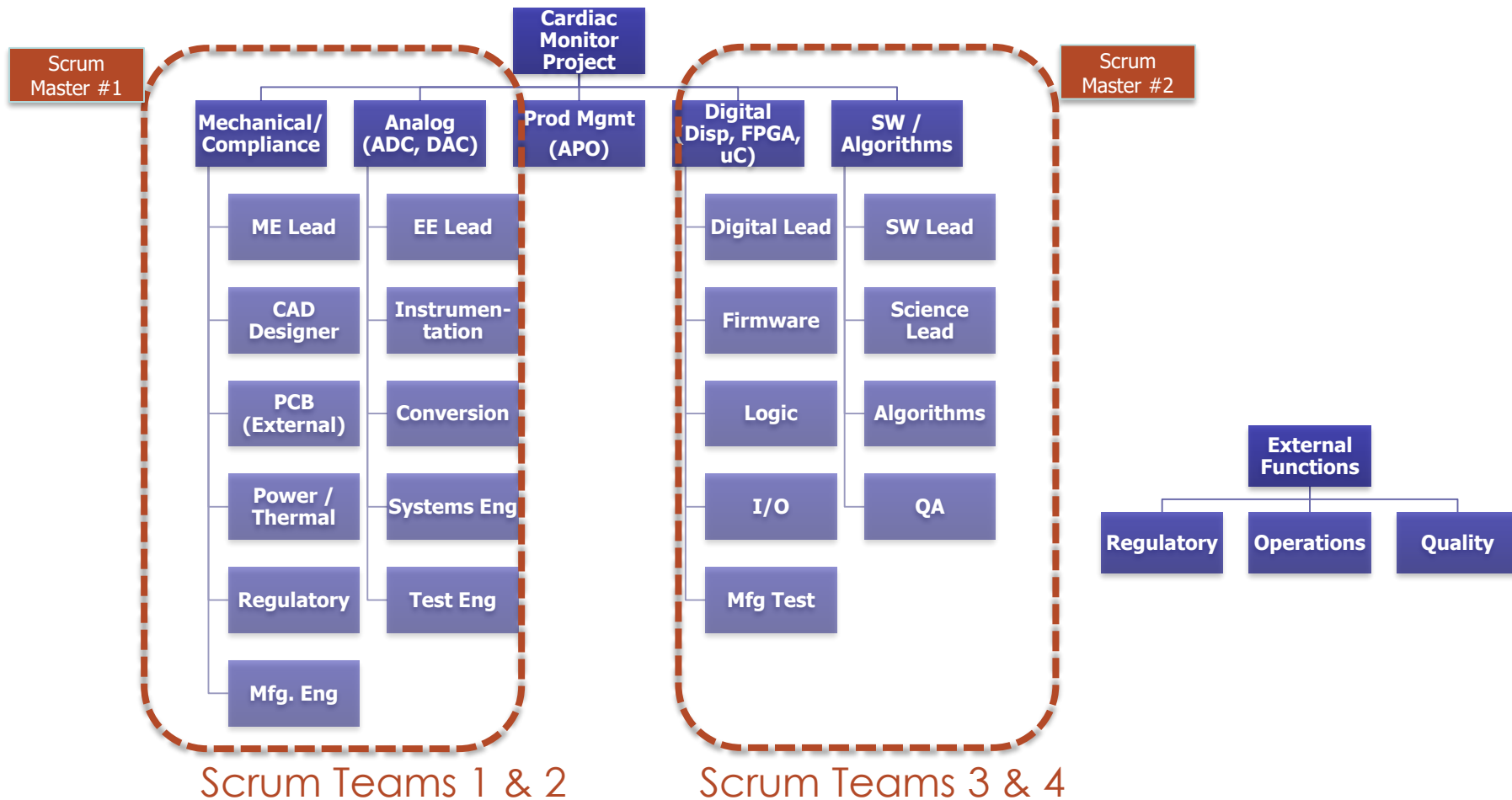
Release Planning

- Release Planning (planning for longer than one Sprint) is often useful in software development, but is optional
- Release Planning is essential for hardware development
 - Higher cost of change implies need to plan complete product development to some degree (more than a Sprint)
 - Variation in type of work over time must be considered in the longer-term plans

Variation in Skills Over Release Cycle

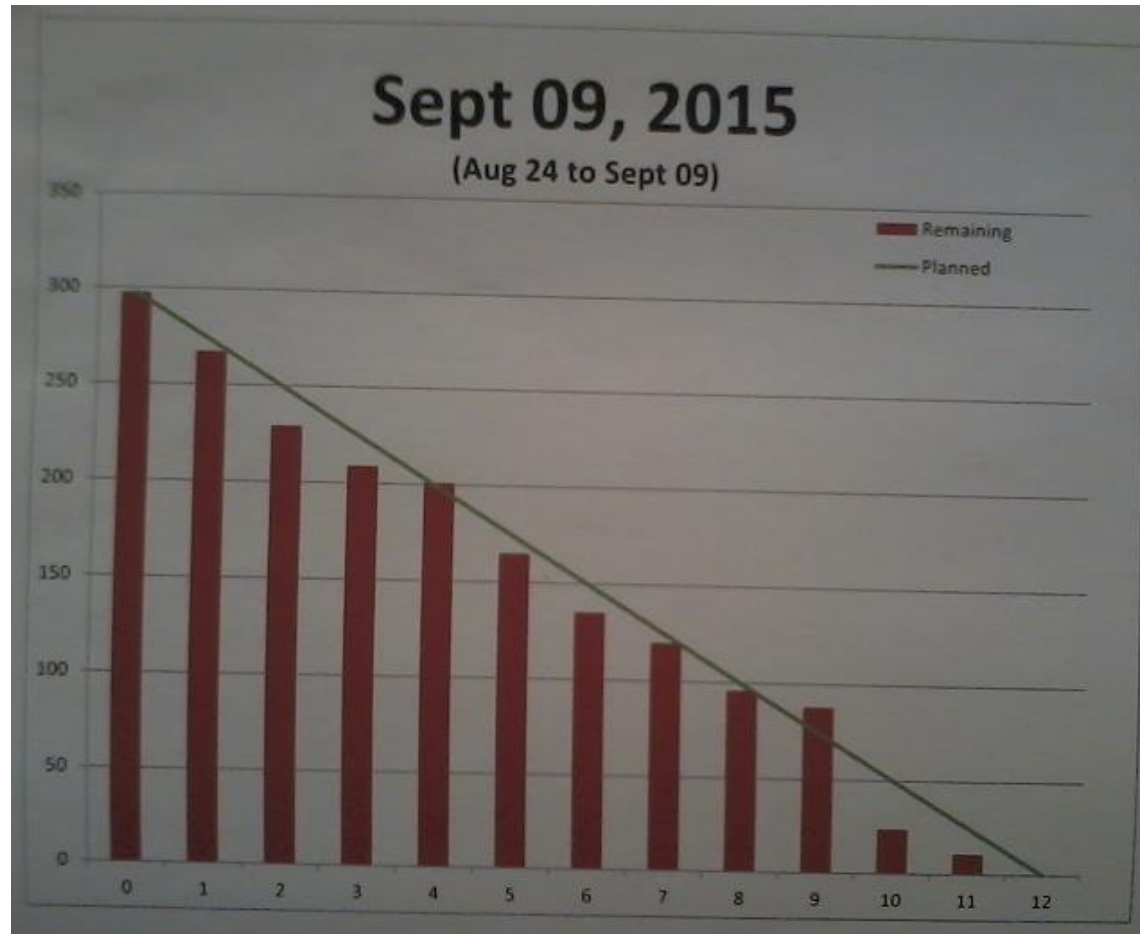


Example Scrum Organization for Hardware Development



Product:
Device for
analyzing
chemical
samples

Image is
Burndown
chart for first
Sprint



1. Hardware development can be done in an Agile way
2. Scrum is well-suited for hardware development
3. Hardware requires time-based estimation, Release Planning, and a shift in types of Stories relative to software

Visit www.cprime.com for white papers and links to Dr. Thompson's book, *Solutions for Agile Governance in the Enterprise (Sage): Agile Project, Program, and Portfolio Management for Development of Hardware and Software Products*

Where It's Happening*



Chromatography and Mass Spectrometry devices for laboratory and medical use



Isotope and Trace-Gas Analyzers for field use



Water-treatment equipment for home and industrial use



Unmanned aircraft, ground stations, and related equipment



High precision, high reliability radio communication and test equipment



Audio and communications products

* Cprime clients

Questions & Comments