

The Agile Hardware Research Project

Kevin Thompson, PhD

December 10, 2015

www.kevinthompsonphd.com

The Researchers



Kevin Thompson

- Former Chief Scientist for Cprime, Inc.
 - Creator of Sprime's Agile Curriculum and Consulting standards
- Author
 - *Solutions for Agile Governance in the Enterprise (Sage): Agile Project, Program, and Portfolio Management for Development of Hardware and Software Products*
- PhD in Physics, Princeton



Scott Elliott

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



John Carter

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

The Research Project

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



Interview Questions

Background

- Describe your role and the products/solutions you deliver.

Establishing Requirements & Planning Estimates

- How are requirements developed and what do they look like?
- How do you go from requirements (the what) to task definitions (the actions)?
- What methods do you use to estimate the work?
- What does a plan look like? (e.g., Gantt chart, Sprint Backlog, something else)?

Managing Changes

- How do you accommodate changes in requirements & scope during development?
- How do you update estimates as the project changes?
- Who makes the decision to change scope?

Future Wishes

- What changes do you plan to make in the future to make your process more agile?

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

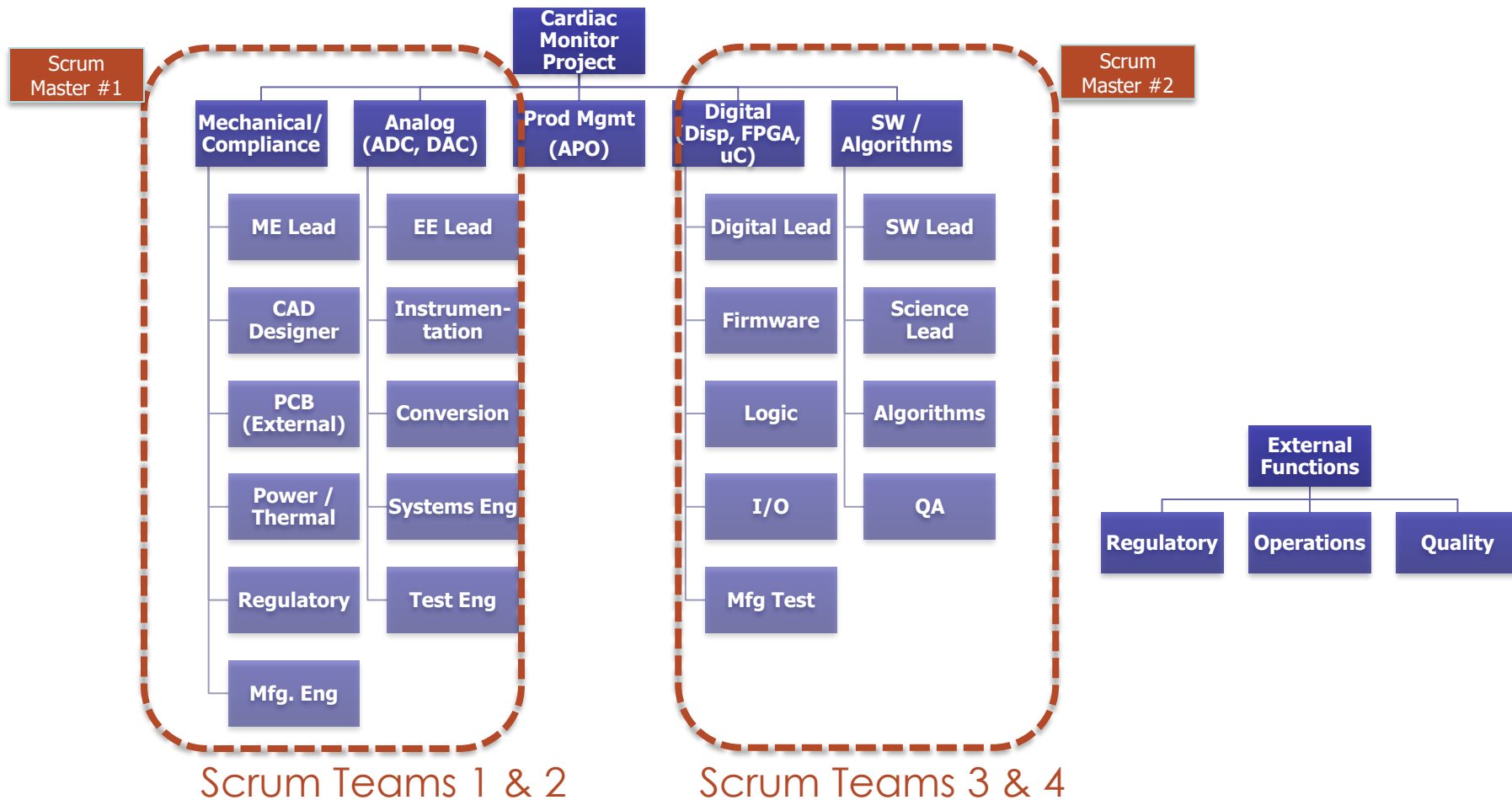
Key Similarities between Hardware and Software Development

- 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: Few User Stories, mostly Technical Stories
2. Sprint Length
 1. Recommend 2-3 weeks, based on duration of design/build/test cycle time
 2. Keep standard length across development
3. Release Planning
4. Variation in Sprint Focus during Release Cycle
 1. Beginning – Requirements, Architecture, Silicon Selection
 2. Middle – Repeating Sprints of design/build/test
 3. End – Emphasis on Scale – Build/test

Example Scrum Organization for Hardware Development



Mechanical Team

Mechanical Team		Team Member	Primary Skills
Product Owner	Anthony	Anthony	Mech. Engineering Lead and Team PO
Scrum Master	Caroline	Alex	CAD Designer
The Mechanical Team focuses on product packaging and compliance.		Mahesh	CAD Designer
		Chun	Power & Thermal Engineer
		Brenda	Regulatory & Compliance
		Todd	Manufacturing Engineer

Analog Team

Analog Team		Team Member	Primary Skills
Product Owner	Franklin	Franklin	Electrical Engineering Lead and Team PO
Scrum Master	Caroline	Yuri	Instrumentation Engineer
The Analog Team handles Sensors, Analog-to-Digital and Digital-to-Analog conversion.		Krish	Instrumentation Engineer
		Sheldon	Conversion Engineer
		Erica	Conversion Engineer
		Dwight	Systems Engineer
		Yani	Test Engineer

Digital Team

Digital Team		Team Member	Primary Skills
Product Owner	Heinrich	Heinrich	Digital Lead and Team PO
Scrum Master	Fred	Chuck	Firmware Engineer
The Digital Team is responsible for the Display, FPGA, instrument control, network communications, and printed circuit development.		Asha	Firmware Engineer
		Heidi	Firmware Engineer
		Jing	Digital Logic/Glue Logic
		Neeraj	IO - Input/Output (Bus)
		Melissa	Manufacturing Test

Software Team

Software Team		Team Member	Primary Skills
Product Owner	Jerry	Jerry	Software Lead and Team PO
Scrum Master	Fred	Mike	Lead Scientist
The Software Team develops algorithms and implements them in software.		Steve	Instrument SW Engineer
		Padma	Instrument SW Engineer
		Kelly	Algorithms SW Engineer
		Gabe	Algorithms SW Engineer
		Tatiana	QA
		Chris	QA

Story Types: Content

User vs Technical

- 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
- Expect higher ratio of Technical Stories to User Stories in hardware than in software

User Story

Title	Buyer views statistics for transactions with Vendors				Rank	3
ID	22		Estimate	8	Total Task Est.	13
Narrative						
As a Buyer, I can view my statistics about my transactions with Vendors, so that I can understand how my history looks to Vendors. When I click on a buyer-statistics link, I see my statistics. (This link is on the home page, under account information.) <u>Prototype / UI References:</u> Landing Page: Attached (landingPage.html) <u>External Documents:</u> Company Style Guide for UI: http://wiki/UI/UIStyleGuide.doc Statistics to be computed: http://wiki/apps/buyerstats.doc						
Acceptance Criteria						
• When the user clicks on the link, the application should display the statistics. • User can create fictitious buyers and suppliers for use in testing. • When the user submits or responds to RFPs, report shows updated statistics that reflect the user's activities.						

Technical Story

Title	16-bit Digital-to-Analog Converter Motherboard Integration				Rank	9
ID	25		Estimate	8	Total Task Est.	60
Narrative						
Design into the motherboard a Digital-to-Analog converter (DAC), in order to provide the high-level analog voltage required to drive the analog display. Assume that the digital drive presented at the input of the DAC module is of sufficient amplitude to activate the unit properly, but not so large as to damage the input gates. Since no DAC has been chosen for this purpose yet, select an appropriate one for incorporation into the motherboard design.						
Acceptance Criteria						
• 16-bit output with +/- 0.5 bit nonlinearity • Single supply operation: 2.7 to 5.5V • The maximum current to drive the unit is 500mA. • The maximum clock rate is 3.4 MBit/sec. • The output will be a time-varying analog signal varying from 0V to 5V at steps of 7.5 μV, with a maximum output impedance of 1 Ohm. • Device meets the other requirements of the I2S serial bus						

Story Boundaries are Firm

Stories in Scrum have “Firm” boundaries—work on a Story should not cross a Sprint boundary

Multiple interviewees objected that Stories could not always be completed within a Sprint, because it is not always possible to build a working device in one Sprint

- Contrary to common belief, it is not necessary to build working product functionality within a single Sprint
- Larger deliverables can always be broken down into multiple smaller deliverables that can be completed within a Sprint

Sprint Length for Hardware

Usual Scrum advice is 2—4 weeks for Sprint length in Software

We recommend 2—3 weeks for Sprint lengths. Longer Sprints increase risk and make tracking of progress less reliable.

- When implementing Agile, start with two-week Sprint lengths
 - Discover whether length is appropriate
- Lengthen if necessary
- Select and use optimum length for all Sprints and all future products

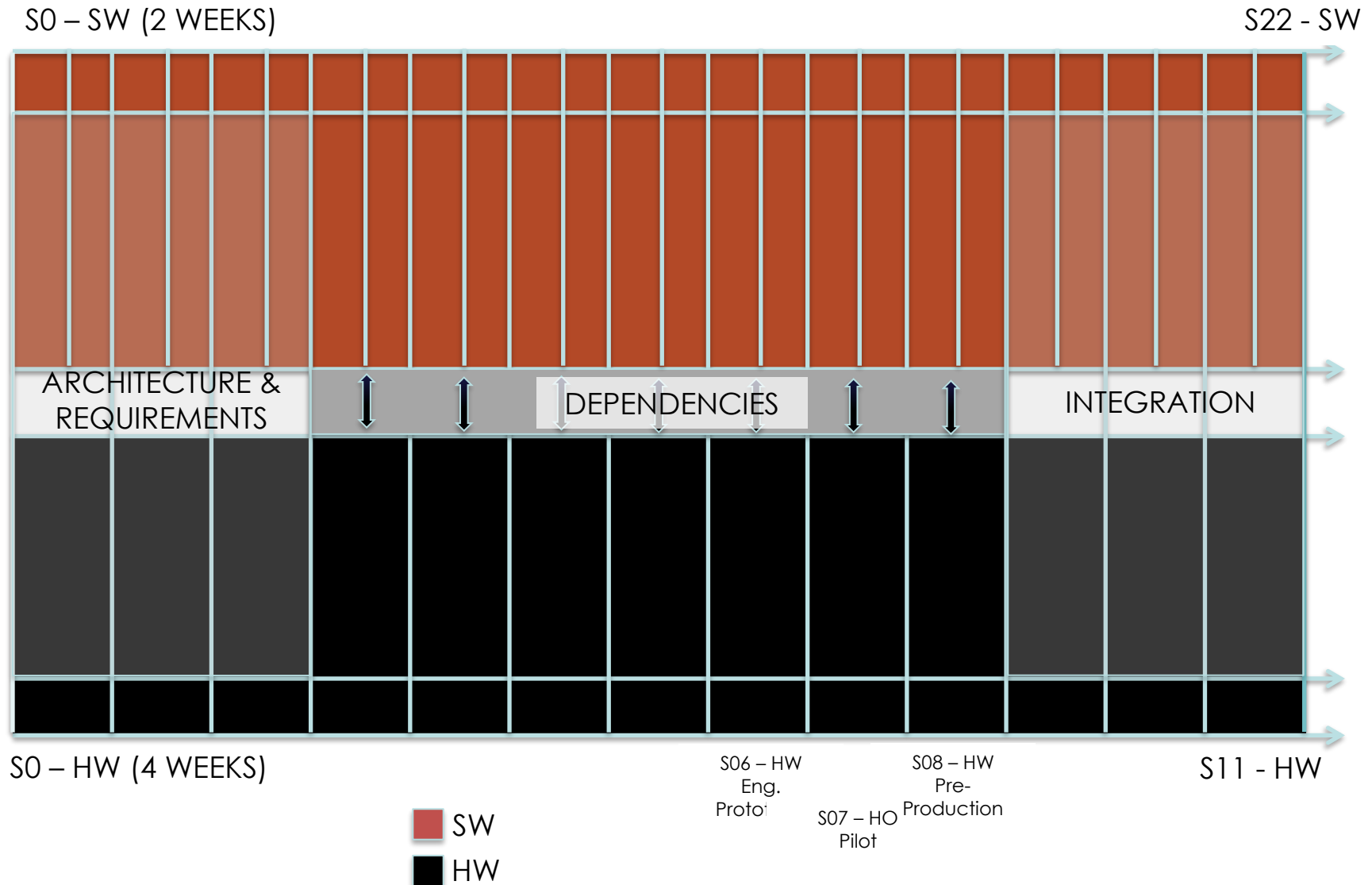
Release Planning

Release Planning is often useful in software development, but is optional

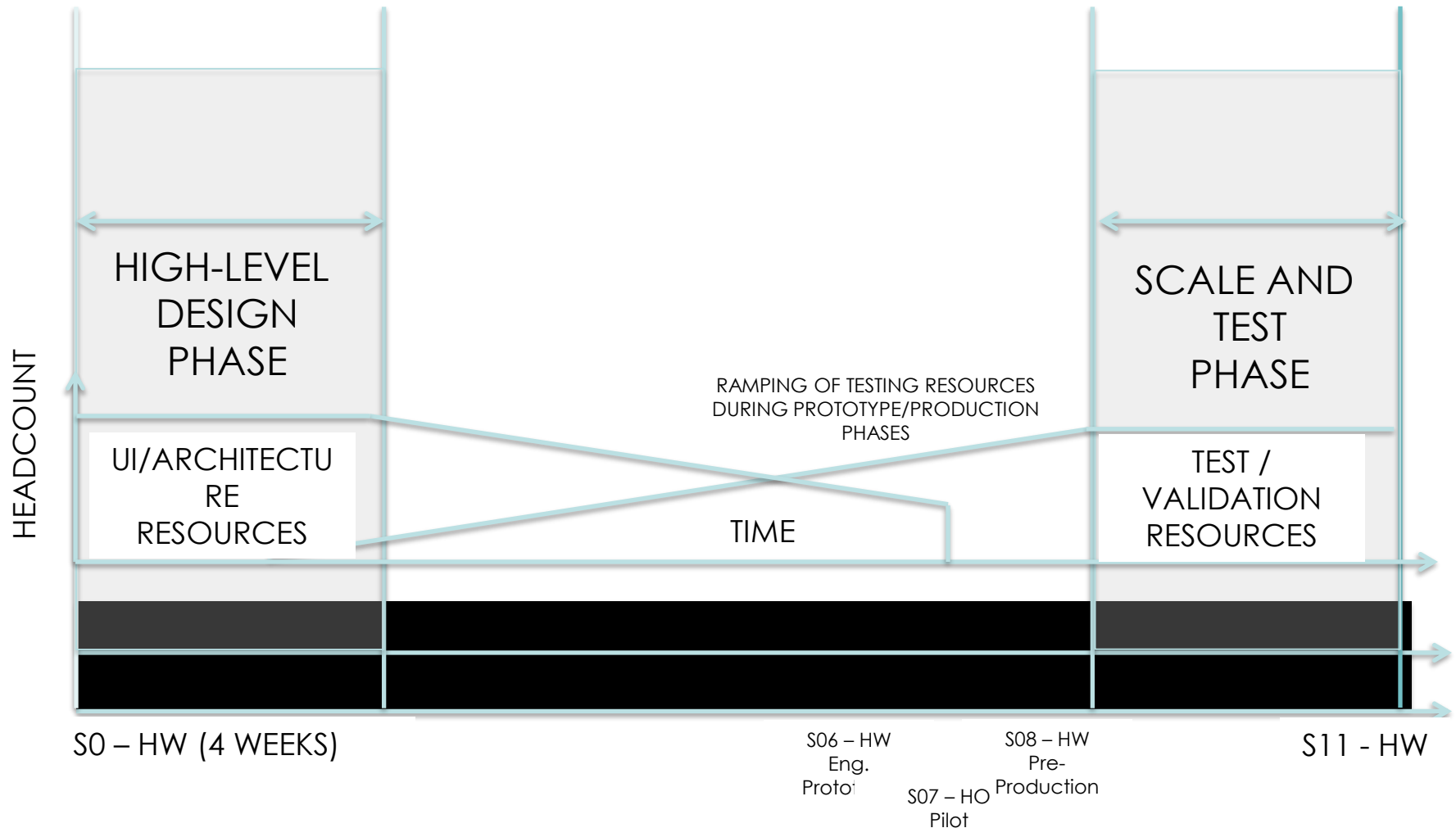
Release Planning is essential for hardware development

- Higher cost of change implies need to plan complete product development to some degree (more than a Sprint)

Variation in Focus over Release Cycle



Variation in Skills Over Release Cycle

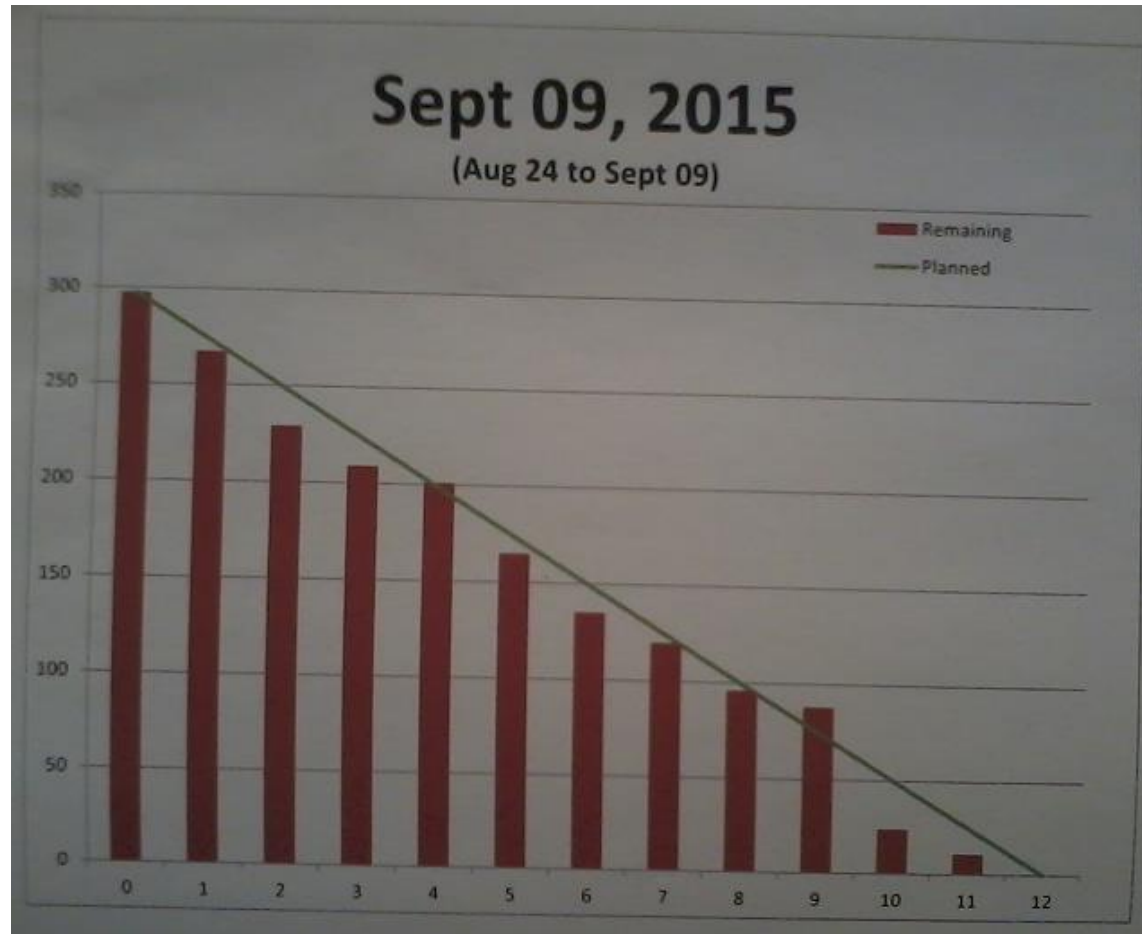


Client Results

Product: Device for analyzing chemical samples

Image is Burndown chart for first Sprint. It shows work hours remaining for unfinished tasks compared to ideal progress line.

Progress is very good.



Conclusions

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

Download our white paper, “Agile Processes for Hardware Development”