

車載機器開発へのインクリメンタル反復開発プロセスの適用

～不確実性の早期低減による新規開発の品質確保～



株式会社デンソー

熱機器事業部

熱エレクトロニクス開発部

○蛸島 昭之

akiyuki_takoshima@denso.co.jp

1. 対象製品 ／ 背景
2. 新規開発における課題
3. 課題解決へのアプローチ
4. 実施方法
5. 実施上の工夫
6. 実施結果
7. 考察と今後の課題

1. 対象製品 ／ 背景

1-1. デンソーの主要製品と発表の対象

3 / 25

● 環境

ガソリンエンジンマネジメントシステム、
ディーゼルエンジンマネジメントシステム、
ハイブリッド車・電気自動車用製品、
スタータ、オルタネータ、ラジエータ、など

● 快適

カーエアコンシステム、
バス用エアコン、空気清浄器、など



● 安全

走行支援システム用センシングシステム、
ABS/ESC用アクチュエータ&コンピュータ、
ヘッドランプコントロールシステム(AFS)、
エアバッグ用センサ&コンピュータ、
車両周辺監視システム、コンビネーションメータ、
ワイパシステム、など

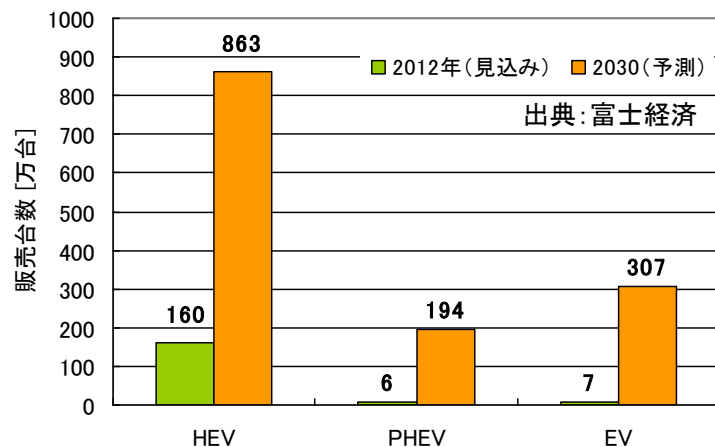
● 利便

カーナビゲーションシステム、ETC車載器、
リモートセキュリティシステム、
リモートタッチコントローラ、スマートキー、
車両運用システム(AVOS)、など

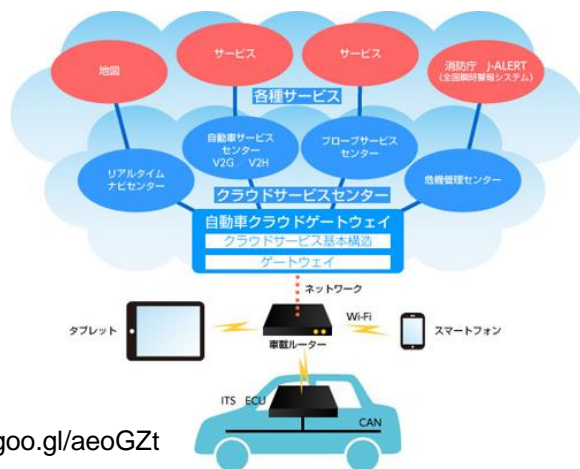
1-2. 本事例の背景

■ HEV/EVの市場予測

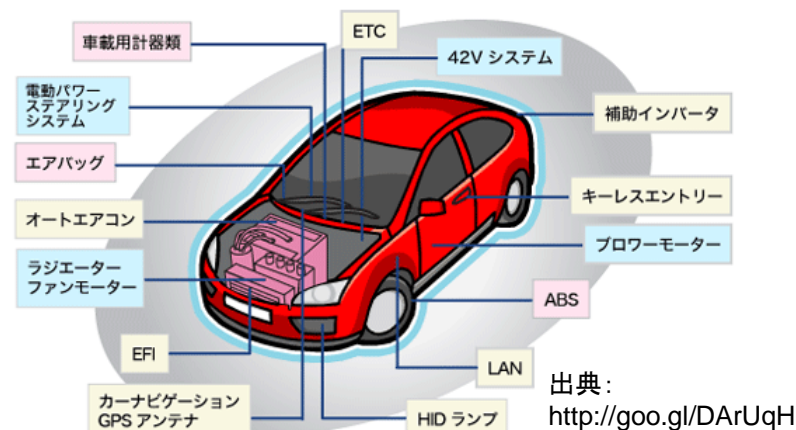
HEV/PHEV/EV市場予測



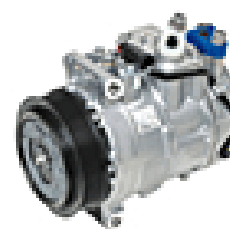
■ クルマは「走る・曲がる・止まる・つながる」へ



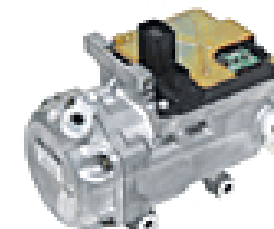
■ ますま進むクルマのエレクトロニクス化



■ 空調制御も更なる省エネ・高機能化のため電動化



ベルト駆動コンプレッサ



電動コンプレッサ

派生開発ではない「新規開発」の比率が増加すると予想される

2. 新規開発における課題

2-1. 派生開発と新規開発の特性の違い

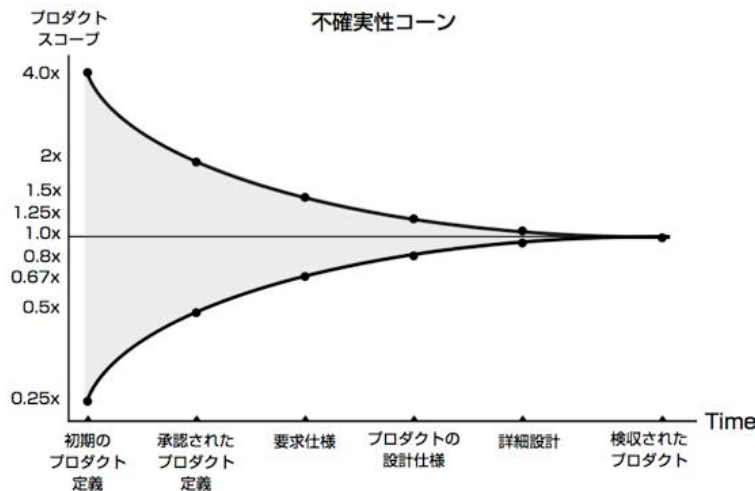
最大の違いはプロジェクト開始時点での**不確実性**の多さ
PMでは不確実性から発生する事象を**リスク**として扱う

- HWと結合評価するまで仕様の妥当性・成立性を判断できない
- 開発開始時にすべての要件が固まっていない
- 新規マイコン/ICが仕様通りの動作をするかわからない
- 新規マイコン/コンパイラの性能がわからない



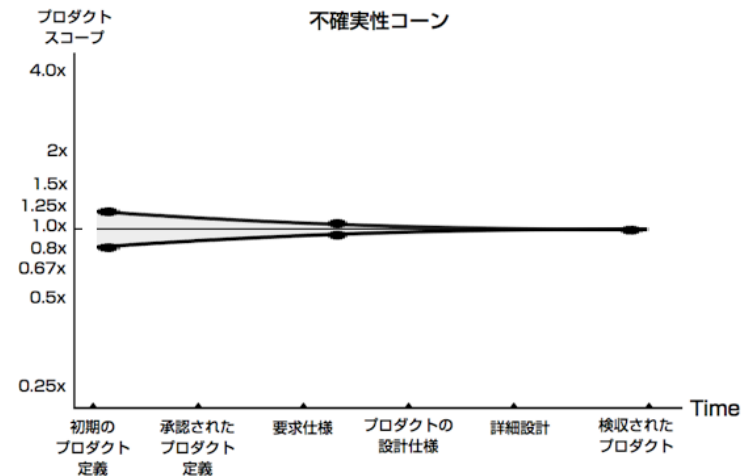
出典: <http://goo.gl/RE0zgZ>

新規開発では**見積りに16倍の差**



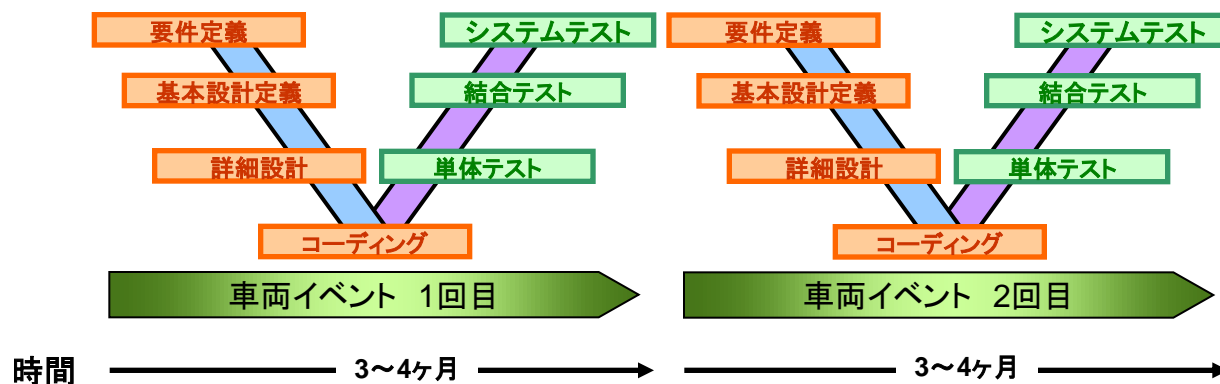
出典: ティーブ・マコネル著『ソフトウェア見積もり』

派生開発では**差分のみ**から不確実性が発生



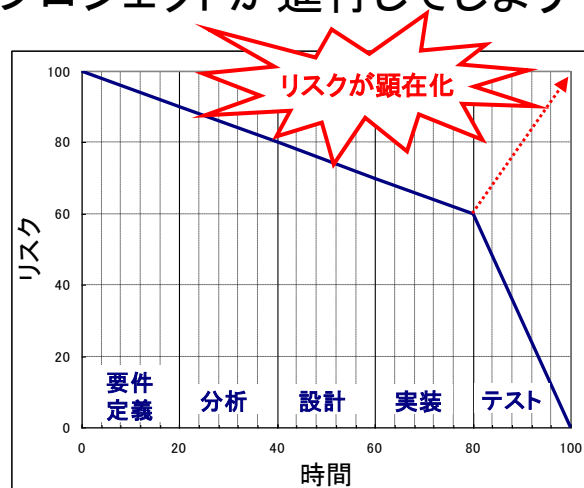
2-2. 従来プロセスの問題点

■ 従来プロセスでは車両イベントごとのV字モデル



■ フェーズ毎の不確実性減少カーブの特性が問題

従来プロセスではV字モデル終盤のテストフェーズまで不確実性が低減できないままプロジェクトが進行してしまう



テストを実施して不確実性が解消された時点で
想定と結果が一致せず**リスクが顕在化**する



分析・設計・実装・テストの再実施による**大幅な手戻り**
により納期遅延やソフト品質の低下が発生

【課題 1】段階的リリース

新規ハードウェアの動作を確認するために一部機能を実装した評価用ソフトウェアを段階的かつ頻繁にリリースできること

【課題 2】不確実性の早期低減

不確実性の多くをマイルストーンの終盤まで残さないようなるべく早期にターゲットハードウェア上でのテストを実施できること

【課題 3】要件追従性

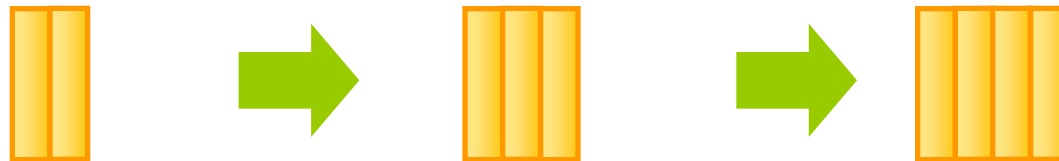
各種評価からのフィードバックによるマイルストーン中の要件変更にも大きな後戻りをせずに追従できること

3. 課題解決へのアプローチ

3-1. 開発プロセスの比較と選択

課題	課題解決に適した開発プロセス
1. 段階的リリース	インクリメンタル型
2. 不確実性の早期低減	インクリメンタル型
3. 要件追従性	反復型

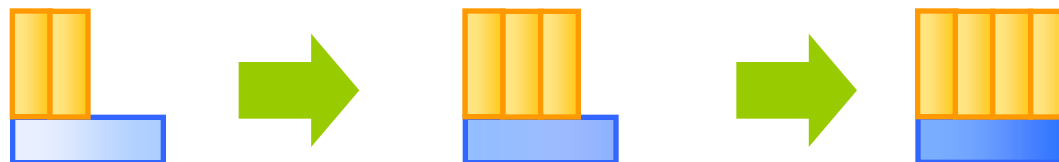
インクリメンタル型プロセス: 段階的なリリースが可能



反復型プロセス: 開発を進めながら要求の精度を高められることが可能



インクリメンタル反復型プロセス: インクリメンタル型と反復型のハイブリッド

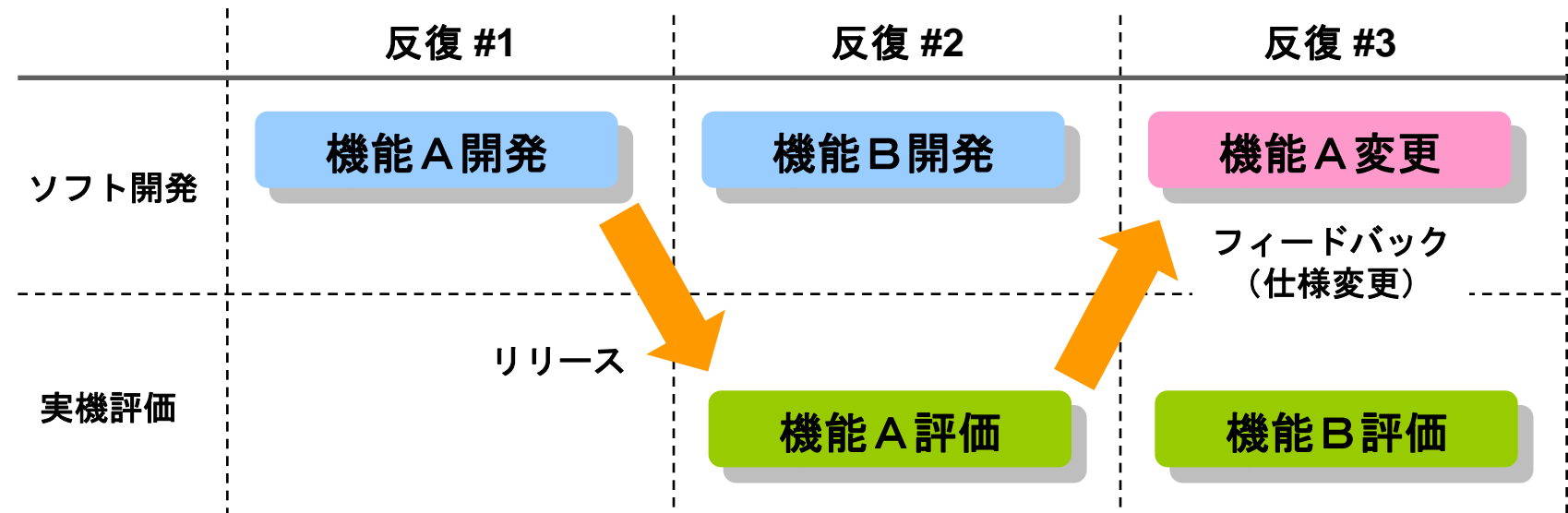
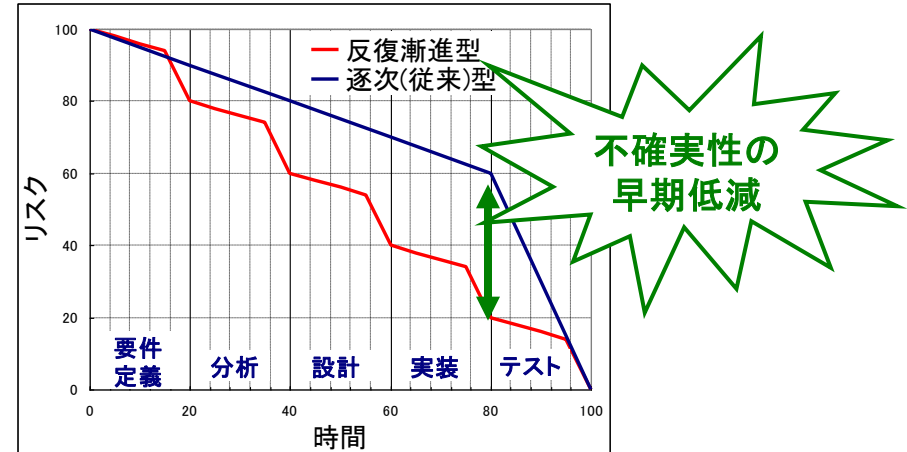


3-2. 効果の事前検証

<検証結果>

3つの課題をすべて解決できる

- ・課題1 段階的リリース
- ・課題2 不確実性の早期低減
- ・課題3 要件追従性



4. 実施方法

4-1. インクリメンタル反復開発プロセスの実施方法

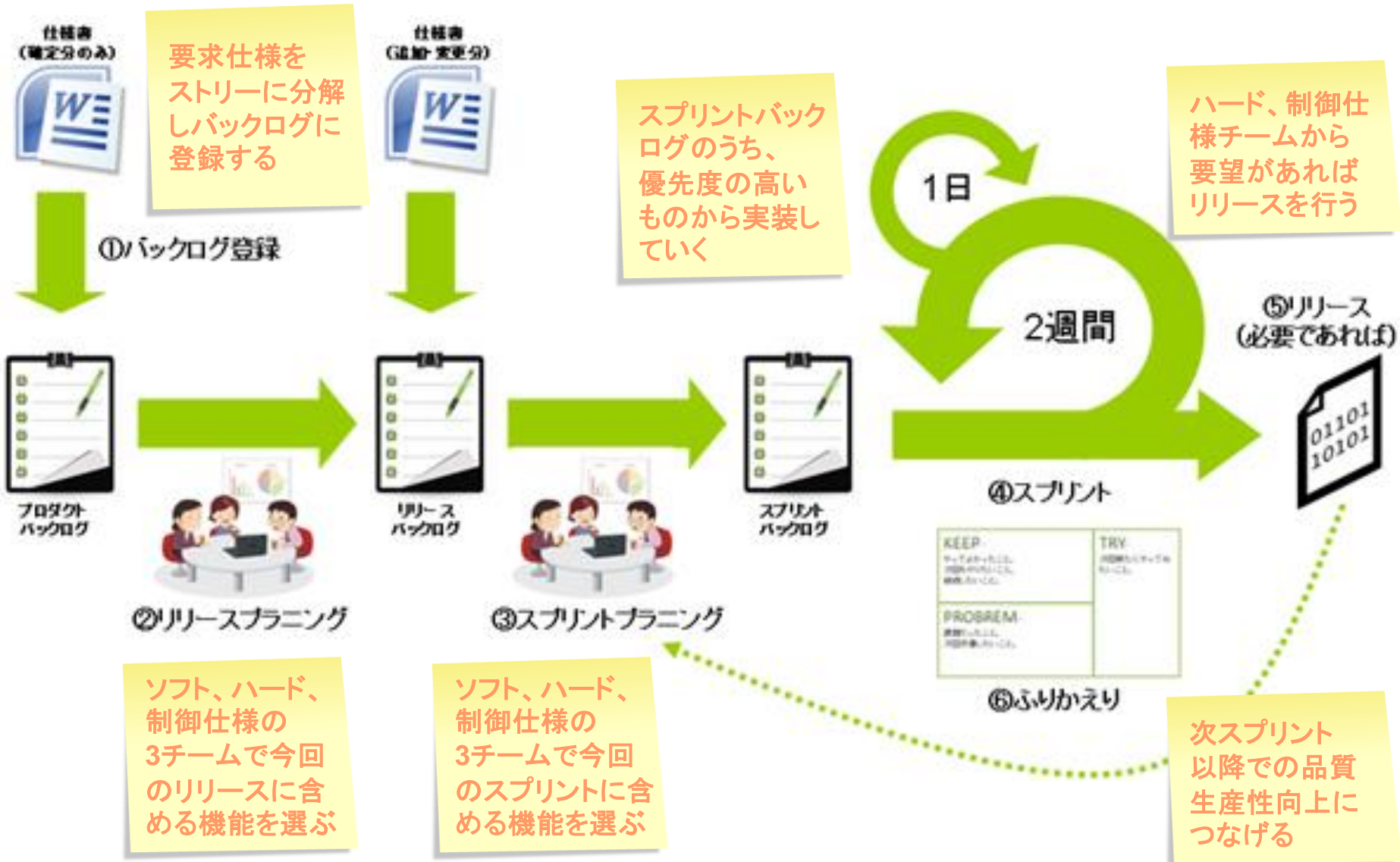
■ 基本方針

- 最も適用事例が多く情報を得やすいため、アジャイル開発手法の1つであるスクラムのフレームワークをベースとする

■ オリジナルのスクラムからの主な変更点

- 専任のプロダクトオーナーは置かず、ソフト、ハード、制御の合議制とする
- 専任のスクラムマスターは置かず、ソフトチームのリーダーとサブリーダーが兼任

4-2. 開発プロセスの概略



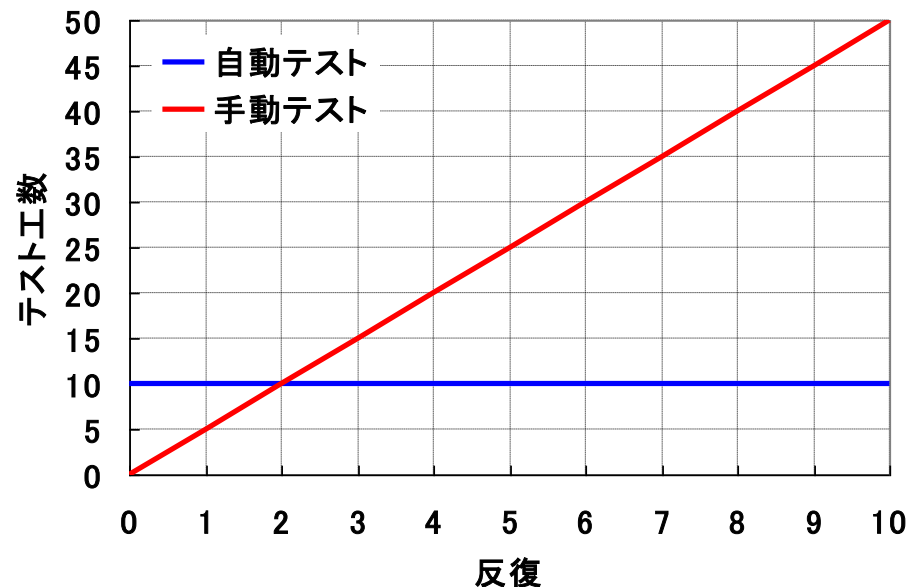
5. 実施上の工夫

5-1. 回帰テストの自動化

16 / 25

■ 課題 I

- 機能追加を行うために既存機能のソースコードを変更する必要がある
- 欠陥の25%はコードの変更または修正によって発生する
- 機能追加によって既存機能がデグレードしていないかを常時確認する必要がある



■ 解決策

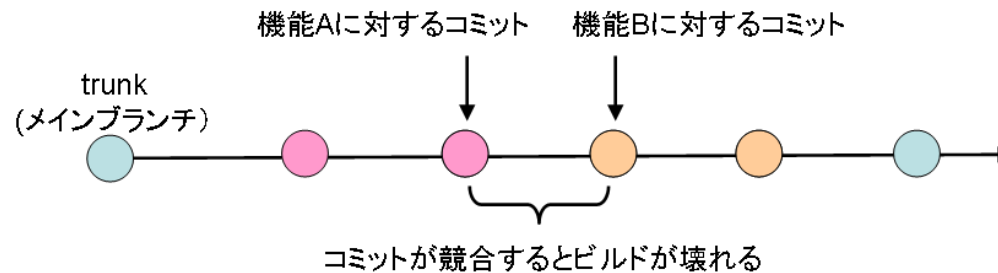
- テスト駆動開発を採用することで単体テストを自動化
- システムテストも可能な限り自動化することでテスト工数を削減
→ 手動テストに対して工数 1/50



5-2. ビルドブレイクの防止

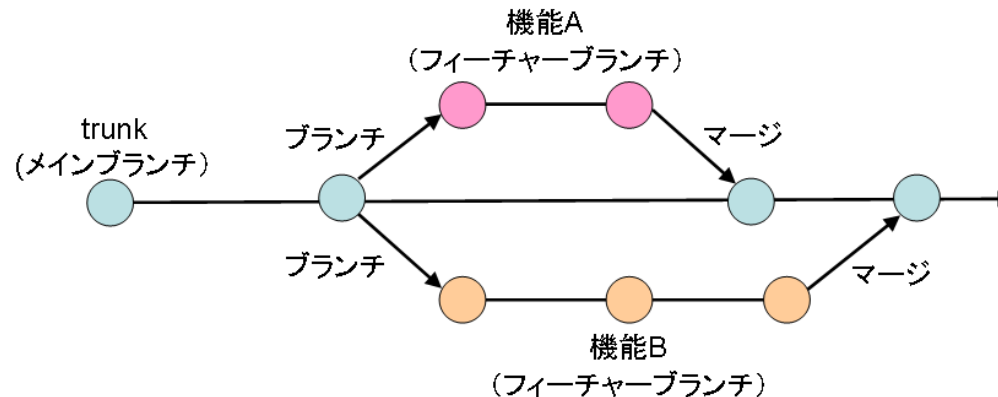
■ 課題 II

- 1つのブランチ上で複数の開発者が同時に開発を行うと、変更の競合によるビルドブレイクが多発しメインブランチをリリース可能な状態に保つことが出来ない



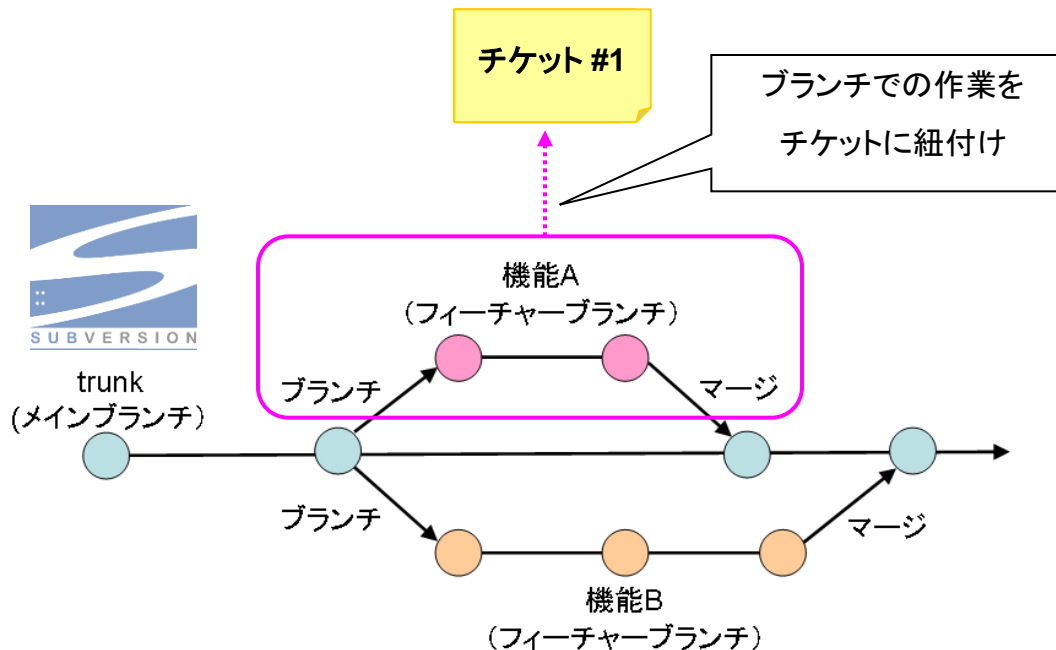
■ 解決策

- プロダクトバックログアイテム(追加する機能)ごとに新たなブランチを作成し機能の実装とテストが完了した時点でメインブランチへマージする



5-3. 成果物品質のばらつき防止

- バックログアイテムごとにSubversionのブランチを作成(チケット駆動開発)
- ブランチへのコミットをチケットに紐付けトレーサビリティを確保
- チケットクローズ条件(DONE)を定義することで作業の完了基準を統一
- マージのたびに発生する繰り返し作業を自動化(継続的インテグレーション)



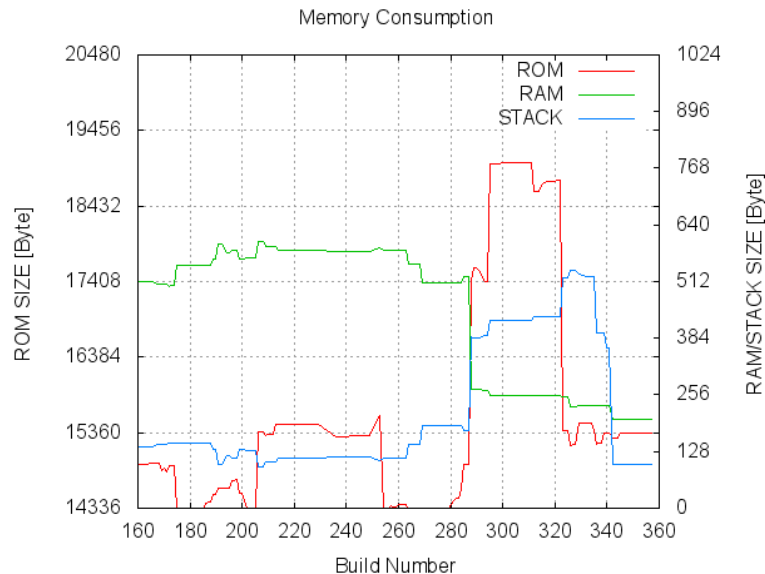
Jenkins

- 静的解析(MISRA-C)
- 単体・結合テストの実行
- ソースコードへのビルド情報の埋込
- オブジェクトコードのビルド
- コンパイラ警告の収集
- ROM/RAM使用量のグラフ化
- API仕様書の生成
- ビルド成果物のコミット

6. 実施結果

6-1. プロジェクトに潜む不確実性早期低減の事例

1. マイルストーン初期のスプリントでリリースしたソフトウェアにより実機動作確認を行うことで、新規ICの過電流保護に関する欠陥が見つかった
2. 単機能ごとに実装を行っていくため、非機能要件(パフォーマンス、ROM/RAM使用率)上のボトルネックが発生しても即座に検出することができた



プロダクトバックログ #444

更新 時間を記録 ☆ウォッチ 複製 コピー 移動 削除

アナログチップ保護機能の無効化

Akiyuki Takoshima が11ヶ月前に追加. 8ヶ月前に更新.

ステータス:	終了	Start date:	2012/06/26
優先度:	今すぐ	期日:	2012/06/29
担当者:	Akiyuki Takoshima	進捗 %:	100%
カテゴリ:	-	作業時間の記録:	8.57時間
対象バージョン:	Sprint #28 - SRVM040	予定工数:	8.00時間
種別:	試作対応	10ms処理負荷[ms]:	5.28
優先順位:	3	0.4ms処理負荷[us]:	207

説明

SPI通信によりアナログチップICの保護機能を無効化できるソフトを試作する

- ・ 起動シーケンス内で無効化コマンドを発行する(コンフィグON/OFF対応)
 - 無効化する対象の保護機能を選択できる(コンフィグ対応)
- ・ LINコマンドにより無効化・無効化解除を実施できる

📎 120625_レジスタ_テストモード設定.ppt (556 KB) 📄 Akiyuki Takoshima, 2012/06/26 10:58:23

子チケット

追加

関連するチケット

追加

計5回のマイルストーンに対して納期遅延なしでリリースを達成

6-2. インクリメンタル反復開発を行ったことによる効果

■ 主な効果

- ソフトウェア、ハードウェア、制御仕様のコンカレント開発が可能となった
- 試作ソフトも最新の安定バージョンをベースに開発を行うことが可能となり、低品質なソフトウェアと未検証のハードウェアの組み合わせによるソフト・ハード同時デバッグを避けることができた

■ 副次的な効果

- 毎スプリント終了のたびにふりかえりを行うことで、プロジェクト期間を通じた絶え間ない改善のフィードバックループが生まれた
- 機能の追加・変更を強く意識することで、疎結合・高凝集な設計が導かれた

7. 考察と今後の課題

7-1. 考察

- 従来プロセスで新規開発を行った場合に予想される問題点を挙げ、その解決策としてインクリメンタル反復開発プロセスを導入した
- 実際の新規開発にインクリメンタル反復開発プロセスを適用し、想定通りにプロジェクトの不確実性を早期低減できることを確認した
- 今後の新規開発にも繰り返し適用可能な開発プロセスの基盤ができた

7-2. 今後の課題

実際にインクリメンタル反復開発プロセスで新規開発を行う中で見つかった以下の課題に対する解決策を検討し、開発プロセスの更なる改善を行っていく

- プロダクトバックログアイテムが着手可能な状態になっているかの判断基準(Readyの定義)がないため、設計フェーズに入って仕様の検討不足が発覚し次スプリント以降へ機能実装が先送りになる
- バックログアイテムの見積り精度が甘く、スプリント内で予定していた作業の全てを完了することができず、次スプリント以降へ機能実装が先送りになる
- バックログの優先順位付けを行う際に、システムの外部的な振る舞いを検証する要件を重視しすぎて、デバイスドライバ層の成熟が遅れる
- 新たな評価ソフト作成依頼による工数増加のリリーススケジュールに対する影響が可視化できていない

ご清聴ありがとうございました