

組込みシステム概論

第14章 システムズエンジニアリング

システムズエンジニアリング
学習のポイント

- ・ 組込みシステム開発では、電気電子、ソフトウェア、機械など多くの技術が必要
- ・ このような複雑なシステムを開発する体系

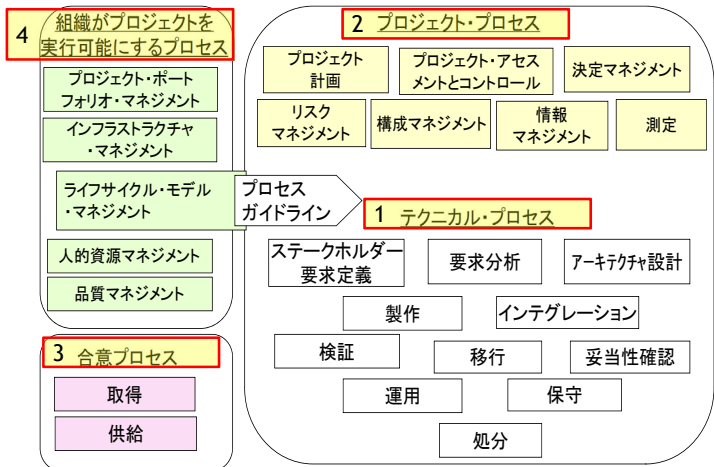
システムズエンジニアリング

- ・ 問題を把握し、要求を分析し、システムの構造を設計し、製造し、検証する手順

表14.1 一般的なライフサイクル段階とその目的と決定関門
(ISO/IEC 15288:2002)

ライフサイクル段階	目的	決定関門
研究	ステークホルダのニーズの識別 アイデアと技術の探索	決定の選択肢： － 次の段階に進む － 次の段階に進むと同時に要処理事項に対応する － 現段階を継続する － 前段階に戻る － プロジェクト活動を停止する － プロジェクトを中止する
コンセプト	ステークホルダのニーズを洗練化する 実現可能なコンセプトを探索する 見込みのあるソリューションを提案する	
開発	システム要求を洗練化する ソリューションを記述（設計）する システムを構築する システムを検証し妥当性確認する	
製造	システムを製造する 点検し検証する	
利用	ユーザーのニーズを満足するよう運用する	
支援	永続的にシステム能力を提供する	
廃棄	システムの記録、保管または処分	

図14.1 システムのライフサイクル・プロセス



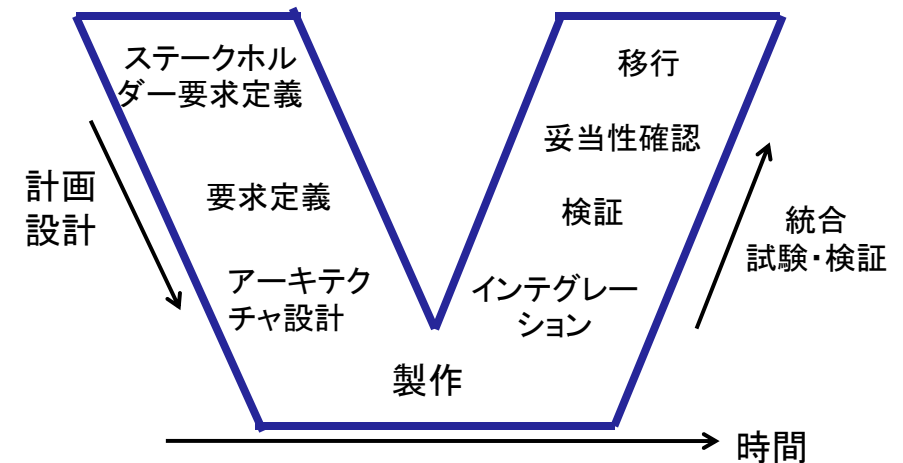
ISO/IEC 15288-2008 [5]を元に作成

14.2.3 テクニカルプロセス

- | | |
|----------------|----------|
| 1. ステークホルダ要求定義 | 6. 検証 |
| 2. 要求分析 | 7. 移行 |
| 3. アーキテクチャ設計 | 8. 妥当性確認 |
| 4. 製作 | 9. 運用 |
| 5. インテグレーション | 10. 保守 |
| | 11. 処分 |

5

図14.2 システムのV字プロセス



コラム：検証と妥当性確認

- 検証 (verification)
 - 正しくシステムを作っているか
 - Are we building the system right?
 - 要求どおり設計したか、設計どおり実装したか (開発者サイドの視点)
- 妥当性確認 (validation)
 - 正しいシステムを作っているか
 - Are we building the right system?
 - 顧客、利害関係者の要求に合ったシステムか (ステークホルダ全体の視点)

7

コラム：トレーサビリティ (Traceability, 追跡可能性)

- 製造段階
 - 全てのシステム、機器、部品材料が対象
 - 設計、部品メーカー、製造ロット、検査記録、試験記録等の履歴の追跡ができるようにすること
- 開発段階
 - ステークホルダ要求定義から、要求分析、システムアーキテクチャ設計、最終的には、要求に合致したプロダクトを製造し、運用するまでの過程の追跡ができるようにすること
 - 要求を行項目に記述し、設計仕様を列項目として記述し、相互の対応させたトレーサビリティ・マトリクスを使う

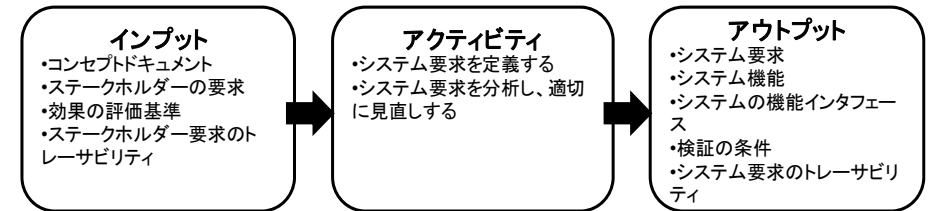
8

14.4 アーキテクチャ設計

- システムアーキテクチャ
 - システムは構成要素の有機的な結合
 - 構成要素の相互接続の記述した**システムの構造がシステムアーキテクチャ**
- アーキテクチャの定義
 - 構成要素の**構造**，構成要素間の**関係**，構成要素の設計や進化を決める**原理**や**指針**（IEEE STD 610.12）

9

図14.3 要求分析プロセスの インプット，アクティビティ， アウトプット



INCOSE, Systems Engineering Handbook Ver.3.2, 2010. [1]を参考に作成

システムの階層構成

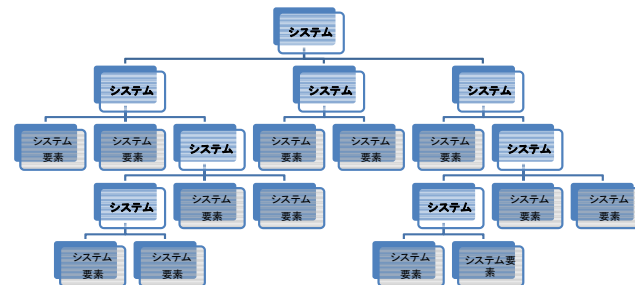


図14.5 システムの階層構成

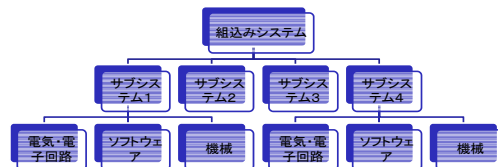


図14.6組込みシステムの階層構成[6]

図14.7 階層的なシステムのV字プロセス （設計，製作，試験の段階を拡大）

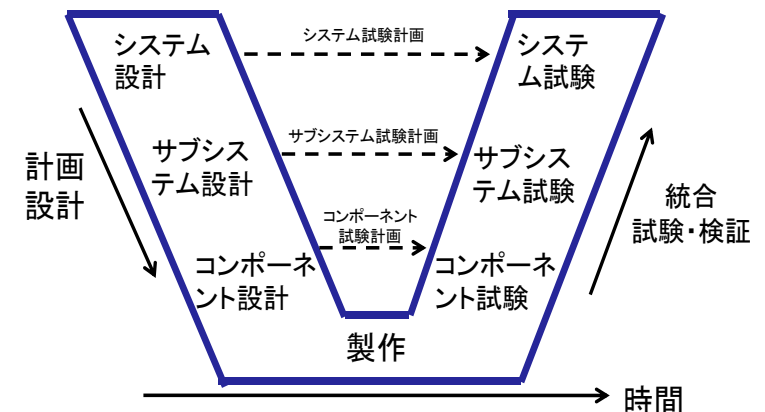
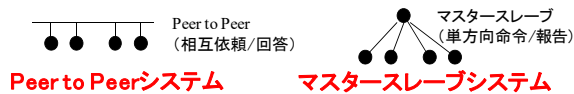


図14.8 アーキテクチャの例

－ 分散システムと集中システム



－ 階層システム

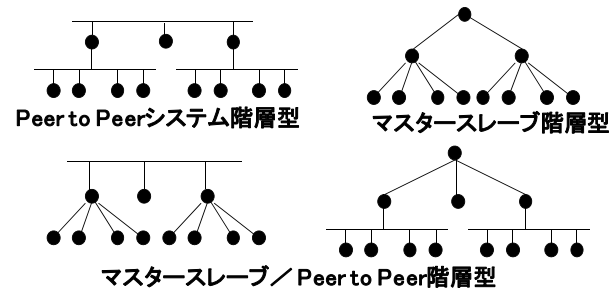


図14.9 組込みシステムの開発プロセス例

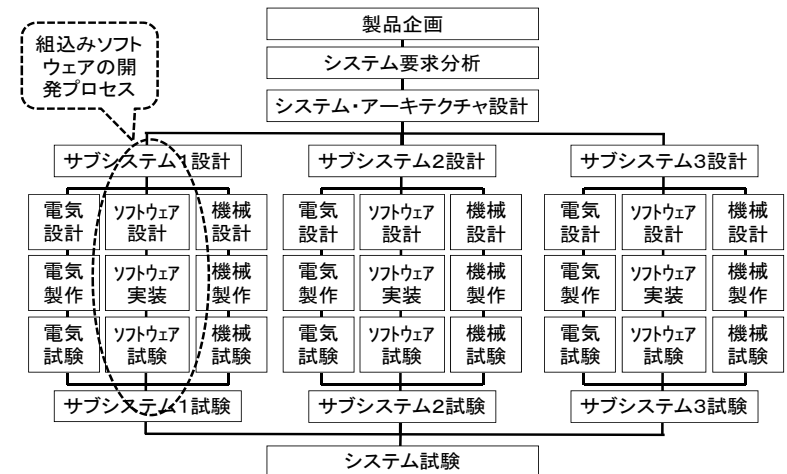
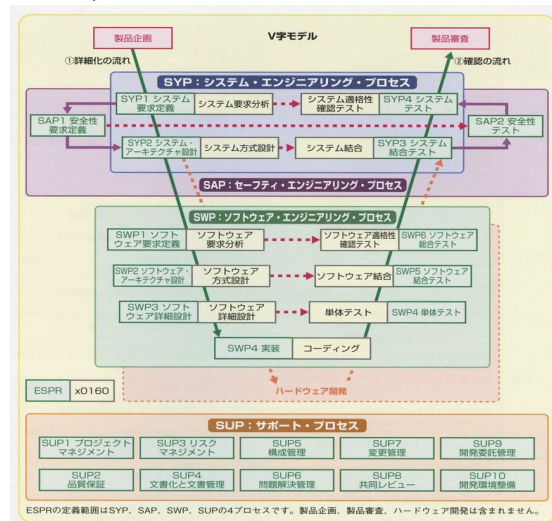


図14.10 組込みソフトウェアの開発プロセス

(IPA/SEC, 改訂版 組込みソフトウェア向け開発プロセスガイド [7]より引用)



14.5 システムのモデリング

- システム開発のなかで**ソフトの比重**が増大
- ソフトの世界では**UML** (Unified Modeling Language) で静的構造や振る舞いを記述
 - 設計の初期段階で検証し、設計品質を向上し、開発コストを削減
- UMLはソフト向けで機械や電気仕様など**組込みシステム全体を表現できない**.
- SysML** (Systems Modeling Language) などが提案されている

図14.11モデリング技法の発展

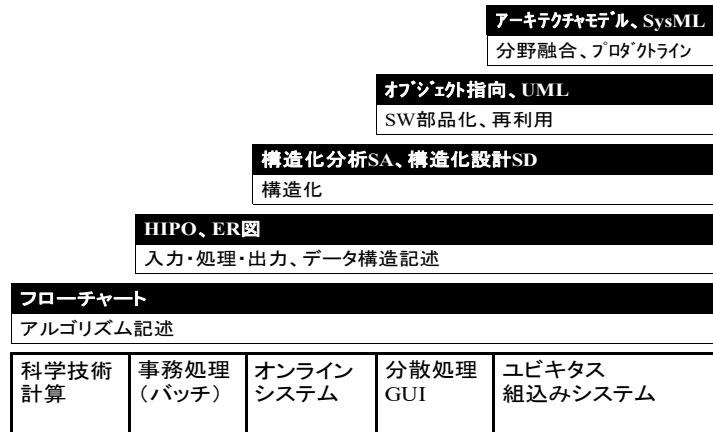


図14.12 SysML OMGシステム
モデリング言語の4つのビュー

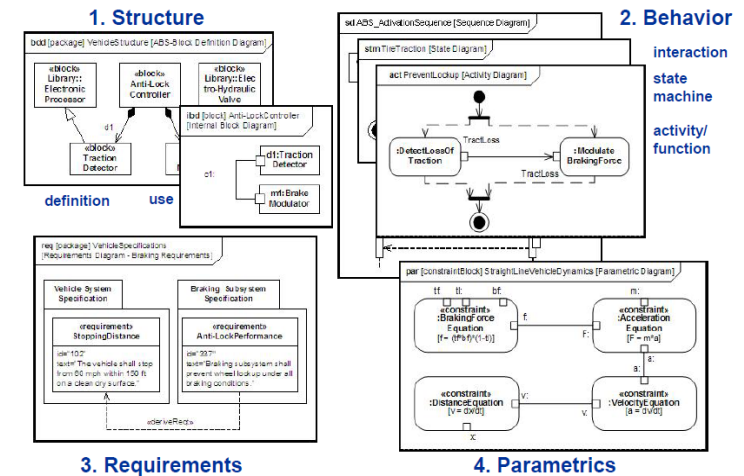


図14.13 Systems Modeling
Language

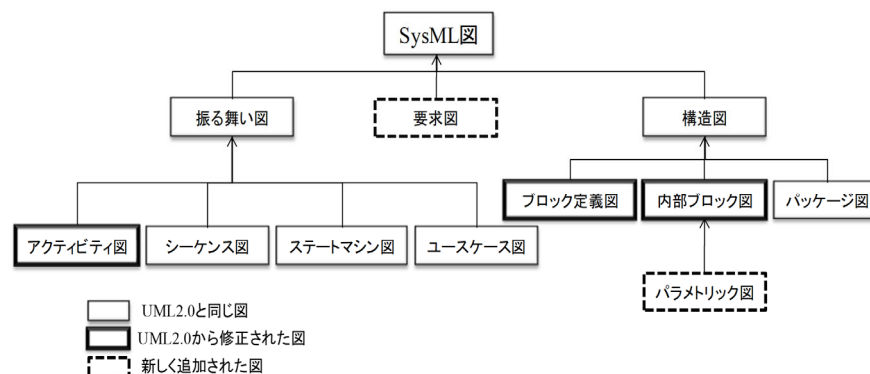


図14.14 飛行船自律航行システム

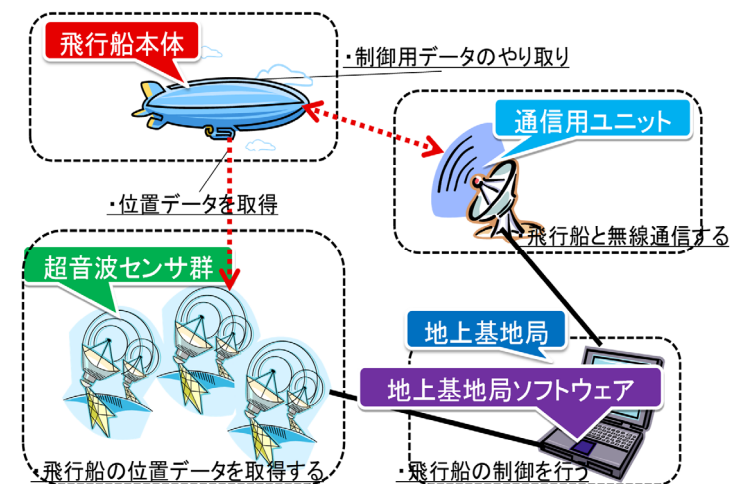
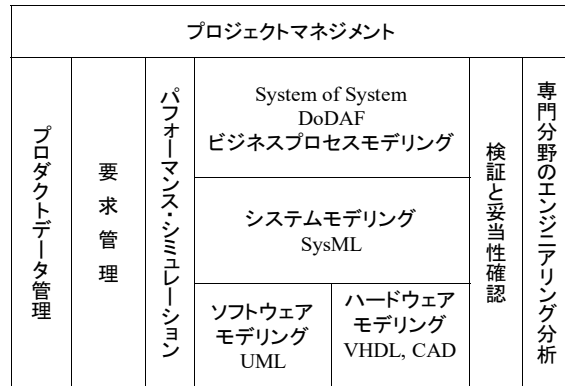


図14.19 統合エンジニアリング環境



注: CAD: Computer Aided Design, DoDAF: Department of Defense Architecture Framework, SysML: Systems Modeling Language, UML: Unified Modeling Language, VHDL: VHSIC (Very High Speed Integrated Circuits) Hardware Description Language
参考文献: OMG Systems Modeling Language (OMG SysML™) Tutorial [6].

システムズエンジニアリング まとめ

- 組込みシステム開発では、電気電子、ソフトウェア、機械など多くの技術が必要
例：飛行船，ETロボコン
- このような複雑なシステムを開発する体系
システムズエンジニアリング
- 問題を把握し、要求を分析し、システムの構造を設計し、製造し、検証する手順

教科書207ページの設問1,3,4に答えよ

演習問題

設問1 システムのライフサイクル・プロセスを構成する4つのプロセスは何かを示し、それぞれを簡潔に説明せよ

設問3 検証と妥当性確認は何が異なるか説明せよ

設問4 機械、電気、ソフトウェアの統合的モデリング言語とは何か。また、その言語を構成する図法を示し、説明せよ