

ハードウェア出身のマネジャーに 分かっておいてほしい7つのこと

Ver.1.02

SESSAME

組込みソフトウェア
管理者・技術者育成研究会

メカ・エレ・ソフトの各工程

メカ

- 要求仕様
- 構想設計
- 部品選定と機構設計
- 機構干渉のチェックなど（設計デバッグ）
- 部品図・部品表作成と出図
- 加工
- 加工チェック
- 組み付け
- 機構干渉チェック, 組み付け後のチェック
- 追加工・再加工
- 総合試験

エレキ

- 要求仕様
- 構想設計
- 部品選定と機能ブロック設計
 - シミュレータで検証の場合も
- 部品表作成と部品手配
- 基板加工手配, 敷線図作成
- 基板チェック（両面基板程度まで）
- 組立
- 動作チェック
- 設計修正・部品変更
- システムテスト

ソフト

- 要求分析
- 方式設計
- 基本設計
- デザインレビュー
- モジュール設計
- コーディング（プログラミング）
- 単体テスト・デバッグ
- 結合（結合テスト）
- 機能テスト
- デバッグ
- システムテスト

ハードウェアとソフトウェアの違い

● SOFTWARE IS DIFFERENT FROM HARDWARE

- General Principles of Software Validation;
Final Guidance for Industry and FDA Staff

- ▶ Section 3.3 / Document issued on: January 11, 2002

- ▶ <http://www.fda.gov/cdrh/comp/guidance/938.html>

Another related characteristic of software is the speed and ease with which it can be changed. This factor can cause both software and non-software professionals to believe that software problems can be corrected easily. Combined with a lack of understanding of software, it can lead managers to believe that tightly controlled engineering is not needed as much for software as it is for hardware. In fact, the opposite is true. Because of its complexity, the development process for software should be even more tightly controlled than for hardware, in order to prevent problems that cannot be easily detected later in the development process.

- ソフトウェアには素早く容易に変更できるという性質もあり、ハードウェア技術者は、あるいはソフトウェア技術者であっても、ソフトウェアは簡単に変更できると信じている。するとソフトウェアに対する理解不足も相まって、マネジャーはハードウェアほど厳密に管理しなくてもよいと思うようになってしまう。しかし実際は逆だ。ソフトウェアは複雑なため、ハードウェアよりも厳しく開発プロセスを管理すべきである。そうしないと、市場に流出してしまうような見つけにくい不具合を防ぐことができないのだ。

ハードウェア出身のマネジャーに 分かっておいてほしい7つのこと

- ソフトウェアの開発はお金がかかる
- ソフトウェアの工程は人間そのものである
- ハードウェアと「協調」することが必要である
- ソフトウェアは見えなかったり自由度が大きいから難しい
- ソフトウェアは工程などがハードウェアよりも未成熟である
- ソフトウェアは詳細を決めないで作ることが多い
- ソフトウェアを知ったかぶりしたり頼被りしてはいけない

ソフトウェアの開発はお金がかかる

- ハードにおける治具や計測器のように、ソフトでもツールなどを購入し、かつ修得し教育して使いこなさねばならない
 - 人とコンピュータがあれば十分なわけではなく、自分では作れないツールなどがないと品質の高い開発は難しい
 - 修得や教育を怠って使いこなせず宝の持ち腐れになっている組織も多い
- テスト機や実機は1人に複数台必要である
 - ソフトも「試行錯誤」する必要があるにある
- CPUやメモリサイズが変更になると意外に作業がかさみ、かなりの変更コストがかかってしまう
 - ハードと違って目に見えないが、部品を取り換えるようなわけにはいかない

ソフトウェアの開発はお金がかかる

- ソフトは費用が見えにくいからといって、ハードのおまけではない
 - 人件費は固定費というのは詭弁である
 - 「変更はすぐにできる」「頑張れば期間も短くなる」というのは間違い
- ハードの問題を何でもすぐに吸収できるわけではない
 - ハードの修正のコストはすぐにイメージでき、ソフトの修正のコストはイメージしにくいかもしれないが、きちんと期間やコストを見積もってトレードオフをする必要がある
- 誰でもソフトが作れるというわけではない
 - パソコンが好きだからといってソフト開発はできないので、きちんと教育や育成をする必要がある
 - 生成AIに適切なコードを生成させるにはスキルが要る

ソフトウェアの工程は人間そのものである

- 生産ラインではスムーズに流れるようよく考えて工程設計をするように、ソフト開発でもきちんと考えて開発者の作業環境を設計する必要がある
 - 机がそれなりに広くないとドキュメントが広げられず、とても生産性や品質が落ちる
 - ディスプレイは大きければ大きいほどよい、できれば複数
 - マシンが遅いと集中力を阻害する
 - 電話や上司の声などで邪魔をされると集中力が切れる
 - 開発環境は好みを入れられるよう幅を持たせてあげる
 - テストで音声や動作音が出る場合は、設計と分けて集中させてあげる
 - 実験室と居室を分けてストレスを和らげてあげる

ソフトウェアの工程は人間そのものである

- ソフト開発の生産性は個人により大幅に異なる
 - ハードと違い、10倍以上の差が出ることもあるが、傍目からは分からない
 - 仕事の量と質が比例しないので質で評価する必要があるが、非常に難しいので、ソフトが分かるマネジャーと相談して進める
 - できない奴ばかりに囲まれていると「できる」奴は腐るので、ジョブローテーションやチーム編成をきちんと行う
 - 日々のプロジェクトに追われず教育や育成をきちんと計画立てて行う必要がある
 - できるエンジニアをこき使っていると知らぬ間に生産性が落ちるので、適宜休ませてやらないといけない
 - 社内ばかりだと向上心が落ちるので、外部の研究会やコミュニティ、メーリングリストに参加し、発言や発表を奨励するようにする
 - 生成AIにコードを作らせても、最終的には人間による確認が必要である

ソフトウェアの工程は人間そのものである

- ソフトウェアチームの人数は、多すぎても少なすぎてもだめであり、後から気軽に投入すると逆効果になることもある
 - 当たり前だが、しかるべきスキルを持った技術者をしかるべき役割に割り振ることが重要である
 - 多人数のマネジメントは難しいので、マネジャーが育っていないとすぐに限界を超えてしまう
 - 人数が多すぎても、インターフェースと会議が増えるばかりでかえって混乱の元となる
 - 後から人を投入すると、追いつくための教育や意思疎通にえらい時間を取られる
 - 協力会社のエンジニアを管理することは結構難しい
 - 仕様や設計、作業内容を確定できず曖昧に進めざるを得ないのでコミュニケーションを徹底的に重視しないとイケない
 - 意志疎通ツールやシミュレーションツール、再利用基盤としてのドキュメントが本当は重要である
 - 人数が少なすぎるとは、作業負荷が多すぎて、いつ終わるかわからず、あげくの果てには心の病気になってしまう

ハードウェアと「協調」することが必要である

- とにかくソフトで何とかしろ、と言わない
 - ハードの軽微な変更が大幅なソフトの変更をもたらすこともあるので、ハードとソフト両方のQCDを十分検討して判断する
 - あおればソフトが早くできるように思えるが、その後バグを取ったりテストをするのにとっても時間がかかることも考慮に入れる
 - ソフトの基本設計や基本データ構造に関わる変更は、ほぼ作り直しになると覚悟するくらいでよい
 - 土壇場でソフトを大きく修正するとメンテナンスが難しくなるので、製品寿命が長く、複数のリビジョンをサポートするような場合は、ハードで解決した方がよいかもしれない

ハードウェアと「協調」することが必要である

- ソフトにもハードの知識がかなり必要である
 - ROM焼き、チューニング、デバッグ環境の設定などは、ハードと協調して進めないと上手くいかず、プロジェクト開始時にも見落とされがちである
 - 結局メカやエレキと組み合わせてみないとソフトがきちんと動作するかどうかは分からない
 - ハードは劣化するがソフトは劣化しない、と思われているが、ソフトが動作するハードが経時変化する事と、対向装置の変化、動作環境や通信量の変動によって隠れていた不具合が露出することがある
 - こうした知識やノウハウは、なかなか形式知になりにくいいため、マネジャーや組織として知識の伝承をサポートしないといけない

ソフトウェアは見えなかったり自由度が大きいから難しい

- ハードウェアに対して操作対象の自由度があまりに大きく、規模が大きい上に入り組んでいるため複雑さが桁違いであるということを理解しないと
いけない
 - 見えなくても内部は込み入っているため、手抜きや仕様変更はそのまま品質に跳ね返る
 - 設計の変更によって、最終的な評価(テスト)のフェーズまでのスケジュール変更が必要になる
 - LSIや回路設計などと異なり、ソフトウェアは全てのケースや入力パターンを洗い出して完璧にテストする、というのは極めて難しく、テストに莫大な工数がかかってしまう
 - **ハードウェア同様に要求分析(要求仕様)も設計も重要であり、それらなしに作成やメンテナンスができると考えてはいけない**

ソフトウェアは見えなかったり自由度が大きいため難しい

● ソフトウェアは見えない

- アーキテクチャの崩れや歪みが捉えにくいので、作業をあおるとメンテナンスが極めて難しいソフトができてしまう
- 関係している部分が把握しにくいので、仕様変更や設計変更でどこに影響が出るかが分からず、デグレードバグが頻繁に作り込まれてしまう
- どこまでテストしたかが分かりにくいので、完成したのかどうか判断が難しい
- ソフトから「進捗90%」と言われても、何に対する進捗か、残り作業は何か、などをしっかり確認せずに、そのまま信じてはいけない
- ソフト開発は全体が見えにくいというのに、多人数で外注込みで開発を進めるので全体の意識合わせが難しく、協調が大変である

ソフトウェアは工程などがハードウェアよりも未成熟である

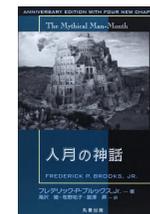
- メカ・エレキは設計・製造プロセスがそれなりに確立しているが、ソフトは課題によって最適な開発プロセスが変わりうる
 - メカ・エレキは加工や部品手配があり、外注や別部所に渡すドキュメントを作成することが多く、工程の把握や設計品質の初期確認が容易であるが、ソフトは設計とコーディングの境が体制的にあいまいになりやすく、ずるずると工程が進んだり設計評価をおろそかにしやすい
 - メカ・エレキは購買部署が「たたけば安くなる」が、ソフトで外注をたたいて値切ると結局は高く付く場合が多い
 - ソフトの設計やプログラム作成はひたすら「人」に依存しているため、人月規模だけではプロジェクトの難しさは単純に押し量れない
 - メカ・エレキのような信頼性があり再利用性の高い機能部品がそうそうころがってはいないので、テストや評価の工数がかさむ

ソフトウェアは詳細を決めないで作ることが多い

- ハードは確定的に設計や実装を進めるが、ソフトは臨機応変(成り行き任せ)にする余地を残したまま設計や実装を開始せざるを得ないことが多い
 - 仕様変更が下流工程での大きな手戻りの原因となるのはハードウェアと同じだが、ソフトでは詳細を作りながら決めていくのでトレーサビリティが確保しにくく、仕様変更を代表とするメンテナンス時に混乱をきたす
 - 工程を踏んでおり計画が重要という点ではハードウェアと同じだが、詳細が初めに定めにくいので管理が難しい
 - 臨機応変にする余地が必要だから分析や設計は無意味、ではなく、だからこそ分析も設計もしっかり行わないとあとで品質問題を多発するようになることはハードウェアと同じである

ソフトウェアを知ったかぶりしたり頼被りしてはいけない

- ハード出身のマネージャは、ソフトを理解しようとせずに知ったかぶりして無理やり理解したことにするか、頼被りしてハードと同じような管理を押し付けることがある
 - 気持ちはよく分かる
 - ハードの理屈を押し付けても上手くいかない
 - だからといって、ソフト部隊の言うことを鵜呑みにしてもいけない
 - ソフト開発がよく分かっている人間を副官におき、よく考えて議論しながら真理を見極めて進めいく
 - ソフトウェア関係の軽い読み物だけでも読んでおく
 - ▶ 「人月の神話 **新装版**」, F. ブルックス, **丸善出版**
 - ▶ 「ピープルウェア **第3版**」, T. デマルコ, 日経BP社
 - ▶ 「ワインバーグのシステム思考法:ソフトウェア文化を創る〈1〉」, G. M. ワインバーグ, 共立出版



本ドキュメントのご利用に際して

- 本著作物の著作権は作成者または作成者の所属する組織が所有し、著作権法によって保護されています
 - SESSAMEは本著作物に関して著作者から著作物の利用※を許諾されています
 - 本著作物はSESSAMEが利用者個人に対して使用許諾を与え、使用を認めています
 - SESSAMEから使用許諾を与えられた個人以外の方で本著作物を使用したい場合は query@sessame.jp までお問い合わせください
-
- ※ SESSAMEが著作者から許諾されている権利
著作物の複製・上演・演奏・公衆送信及び送信可能化・口述・展示・上映及び 頒布・貸与・翻訳・翻案・二次的著作物の利用
 - ドキュメント中には Microsoft社, Adobe社等が著作権を所有しているクリップアートが含まれています



<http://www.sesame.jp/>