



組込み概論、下から見るか 横から見るか

桐畑 鷹輔



自己紹介

- 桐畑 鷹輔



- 社会人6年目

- 京都の片田舎で組込みエンジニアやっています。
- 内3年は自社SI、4年目～親会社製品ファーム開発
- 分析装置、医用機器、航空機器、産業機器

目的

- 本セッションは名前こそ「組込み概論」ですが、今回は目的を下記としています。
- ①参加者の皆さんが組込み技術者として押さえておきたいポイントを理解する
- ②ポイントを押さえながら開発をできるようになる
- =>バグや不具合を作りこまないでイイ！

目標設定

- 目的達成(ポイント理解)のため、組込み開発で発生したトラブル(事前学習事例)について、原因は何だったか、どうすれば防止できたかを議論してもらいます。



図1.目標達成の様子

事例 1 のおさらい

- 事例 1 を朗読します。



図 2 .事例を読んだ時の気持ち

事例1：メモリリーク

- チームで問題の真因を検討してみましょう！(5min)
- チームで対処方法を検討してみましょう！(5min)
- 結果を後ほど発表してもらいます！(各30sec~1min)
- ※本文以外で疑問に思う箇所、可能性(こんな問題があったのでは?)についても自由に考えてください

事例2のおさらい

- 事例2を朗読します。



図3.事例を読んだ時の気持ち

事例2：時間内処理負荷

- チームで問題の真因を検討してみましょう！(5min)
- チームで対処方法を検討してみましょう！(5min)
- 結果を後ほど発表してもらいます！(各30sec~1min)
- ※本文以外で疑問に思う箇所、可能性(こんな問題があったのでは?)についても自由に考えてください

まとめ

- 未然防止のための施策
 - 事例 1 : 実機検証フェーズの設定、言語・ライブラリ導入の判断
 - 事例 2 : 新規技術のトレーニング、知見ある方へ教えを請う
処理時間測定、コードレビュー
- リスクマネジメント
 - リスクの洗い出し→リスクインパクトの見積→アクション合意形成
 - テストでのバグ潰しは本質的には違う→最終手段
 - 品質は上がるが元々品質の高い（バグが入らない設計なら）
コードならバグ自体が出ない

おまけ：トラブルキーワード

- ハードウェア(電気系)要因による誤動作
 - IC端子がオープン→端子が正しい電圧レベルとならず、所望の動作をしない
- 時間内処理負荷
 - デバッグ出力有無による処理時間差
- 割り込み干渉
- バージョン管理・差分チェック漏れ