

## 調査 5 モデルカリキュラムの提言 コースウェア

### 27. 組み込みシステム最適化に関するスキル

I. 概要	組み込みシステムの性能最適化のためのソフトウェア、ハードウェア、構成、設計などの必要な機能と、その手順、方法に関して、実際の開発・運用の際に必要な管理知識・手法の種類と特徴、内容を学ぶ。
II. 対象専門分野	職種共通
III. 受講対象者、 受講前提	本カリキュラムの「組み込みシステムに関するスキル」を受講済みであること。または、同等のスキルレベルの知識を有すること。
IV. 学習目標	<ul style="list-style-type: none"><li>・ 組み込みシステムの最適化にはどのような手法があるか、その方法や手順はどのようなものか、を理解する。</li><li>・ 組み込みシステムの最適化を考慮したシステムの活用シーンとその目的、メリット、実際の適用内容を理解する。</li><li>・ 組み込みシステムの最適化を実際のシステム開発に導入する作業の留意点を理解する。</li></ul>
V. 使用教科書、 教材等	『H8 マイコン+C 言語効率アップテクニック』 鹿取祐二著、オーム社刊
VI. 習得スキル の評価方法	講義終了後の受講レポート、定量アンケート、知識確認ミニテスト、演習問題の取り組み状況を総合的に判断して評価を行う。
VII. カリキュラム の構成	レベル1 第1回～第6回 レベル2 第7回～第15回

## 講座内容

---

### 第1回 マルチプロセッサシステム(講義 90分)

---

マルチプロセッサの種類、活用方法について理解する。

#### (1)マルチプロセッサの分類

1. 従来からの分類
  - ・ SISD (Single Instruction stream Single Data stream)
  - ・ SIMD (Single Instruction stream Multiple Data stream)
  - ・ MISD (Multiple Instruction stream Single Data stream)
  - ・ MIMD (Multiple Instruction stream Multiple Data stream)
2. 新しい分類方法
  - ・ SMP
  - ・ クラスタ構成
  - ・ MPP

#### (2)マルチプロセッサシステムの活用方法

1. 複数事象の制御
2. タスクのスケジューリング
  - ・ 最適の分割
  - ・ ハードウェア処理との整合性
3. プロセッサ間の同期
  - ・ 論理的なつながり
  - ・ 時間的な順序
4. プロセッサ間の通信

#### (3)ハード面の考慮点

1. プロセッサ間通信
2. マルチプロセッサシステムの資源の共有

#### (4)マルチプロセッサシステムの構成例

---

## 第2回 ハードウェアによる最適化(講義 90分)

---

マルチプロセッサを使用したシステムの最適化について理解する。

(1)性能の最適化

(2)CPU 高速化の仕組み

1. パイプライン処理
2. マルチプロセッサ構成
  - ・ SMP
  - ・ AMP
  - ・ DMA

(3)入出力方式

1. 各種入出力機器の接続法の原理とタイミング
2. 動作速度が遅いメモリとの接続方法

(4)低電力処理

1. 使用電力の見積り
2. 電力の配分とその設計

---

## 第3回 リアルタイムシステムの設計(講義 90分)

---

リアルタイムシステムの設計に必要な要素について理解する。

### (1)リアルタイムシステム設計

1. モジュール分割
2. 割り込み処理とアプリケーション
3. 最適化のための処理分割
  - ・ 処理要求の確定
  - ・ デバイスドライバ
  - ・ カーネル
  - ・ アプリケーション

### (2)アプリケーションタスクの設計

1. 処理要求の発生と応答時間の分析
2. アプリケーション処理の分割
3. 処理横断的な分割
4. 処理待ちのキューの設計

### (3)デバイスプログラミング

1. 高速応答処理の実装
2. カーネルのシステムコール代替
3. カーネルインタフェースの設計

---

## 第4回 リアルタイムソフトウェアの条件と最適化(講義 90分)

---

リアルタイムソフトウェアを利用したシステムの性能向上、最適化を行うためのソフトウェア作成方法、設計の方針について理解する。

### (1)カーネル処理とドライバ

1. デバイスドライバ
2. アプリケーション

### (2)タスク設計

1. タスク分割
2. キューの作成とタスクの同期
  - ・ タスク分割とデータのやりとり
  - ・ 入力スピードと出力スピードの調整
  - ・ キューの役割
  - ・ キューを作成するエリア
  - ・ 処理依頼の扱い
  - ・ キューと同期
3. イベントフラグによる同期
  - ・ 並行動作の単位
  - ・ 処理要求の発生の仕方
  - ・ 処理要求に沿ったタスク分割
  - ・ 互いに関連のないデータの流りに沿ったタスク分割

### (3)入出力単位

1. 入出力単位のタスク分割
  - ・ 入力と出力の差
  - ・ 受信データの出力処理
2. 受信タスクと処理タスクへの分割

### (4)タスク関連設計による最適化のアプローチ

1. タスク関連図の作成
  - ・ タスク番号と優先順位
  - ・ タスクの起動関係
  - ・ 起動タイミング
  - ・ タスク間のキュー
  - ・ タスク間の同期と排他制御
  - ・ 重要なフラグの更新と参照の関連

---

## 第 5 回 性能最適化の評価項目(講義 90 分)

---

リアルタイム処理の設計を導入したシステムの性能評価について理解する。

### (1)評価項目と留意点

1. パフォーマンス
  - ・ タスクのスイッチング時間
  - ・ カーネルによる全体のスケジューリング方式
  - ・ 機能とスピードのトレードオフ評価
2. 全体の優先順位
  - ・ プリエンプションの発生タイミング
  - ・ 処理の遅延
3. デバイスドライバの処理方式
  - ・ イベントを発生させる性能
  - ・ 割り込み処理の制御仕様

### (2)タスク分割の設計評価

1. キューイング
2. 排他制御
3. 同期

### (3)MPU の評価指標ごとの評価方法

1. CPI (Clock Cycles Per Instruction)
2. MIPS (Million Instructions Per Second) 値
  - ・ 計測条件
  - ・ MPU のアーキテクチャとの依存関係
  - ・ 異なる命令セット
3. MPU の性能に相反する場合
  - ・ 割り込みを多用するシステム
  - ・ 受付後に多数のレジスタの退避と復帰をハードウェアで行うシステム
  - ・ レジスタを命令で退避するシステム

---

## 第6回 ソフトウェアの最適化(講義 90分)

---

ソフトウェアによるシステムの性能向上、最適化の基本概念、設計方針について理解する。

### (1)ソフトウェア最適化の方法

1. 冗長なコードの排除
2. コードの最適化
  - ・ ループ
  - ・ 内部変数
  - ・ 処理ブロック

### (2)プログラムと処理

1. プログラムのローディング時期や方法
2. プログラムのリンク方式
  - ・ すべてのモジュールを一本にリンク
  - ・ カーネルとデバイスドライバを一本にリンク
  - ・ アプリケーションの仕様に応じて個別にリンク
3. 常駐と非常駐
4. 常駐化の検討
5. 非常駐の検討
6. リューザブル化

### (3)最適化コンパイラ

1. 仕様
2. 最適化のパターン
3. 最適化の利点／欠点

---

## 第 7 回 MPU の性能最適化設計(講義 90 分)

---

MPU を実装するシステムの性能向上を目指した設計の方針、最適化の基本概念を理解する。

### (1)MPU のアーキテクチャとリスク

1. 基本ハードウェア構成とリスク
  - ・ MPU の基本構成とリスク
  - ・ 演算装置のリスク
  - ・ 制御装置のリスク
  - ・ レジスタのリスク
2. ノイマン型の MPU のボトルネック
3. メモリや I/O の接続
4. アドレスバス
  - ・ データバス
  - ・ 制御信号
  - ・ 読み出しサイクル
  - ・ 書き込みサイクル
  - ・ タイミング

### (2)MPU のアーキテクチャ

1. MPU の評価
  - ・ MPU の性能
  - ・ 割り込みの発生回数
  - ・ DMA の発生回数の変化
2. MPU の性能を示す指標
  - ・ CPI
  - ・ MIPS
  - ・ MFLOPS
  - ・ SPEC



---

## 第 8 回 システムの性能要件と評価項目(講義 90 分)

---

システム性能の要件、考え方、評価方法を理解する。

### (1)システムの性能要件定義

1. 処理時間の計算
2. オーバラップ時間
  - ・ リアルタイム OS の持つシステムの性能計測
  - ・ イベントの発生から終了までの時間の計測

### (2)アクセス速度の主な評価項目

1. 待ち時間
2. MPU
3. MPU のクロック
4. メモリ、I/O のアクセス時間
5. バンド幅

### (3)データ転送時間の高速化

1. 高速のメモリデバイスを使用
2. キャッシュメモリの搭載
3. データバス幅の拡大
4. 高速転送モード

---

## 第9回 システム性能の評価方法(講義 90分)

---

システム性能の考え方、計測、量産時の制約事項とのトレードオフを理解する。

### (1)システムの性能要件定義

1. システム設計からの制約
  - ・ ソフトウェアに許される容量
  - ・ ハードウェアの制約
  - ・ マイコン構成の全体条件
  - ・ マイコンによる連動、情報共有の要件
2. システム要件としての括りだし
  - ・ ソフトウェアの性能
  - ・ ハードウェアの性能
  - ・ ネットワークの性能
  - ・ 全体最適化

### (2)システムとしての制約条件の確認

1. コスト面からの検討
  - ・ 量産価格
  - ・ 部品の流用とのトレードオフ
  - ・ 材料費用
2. 拡張性からの検討
  - ・ ハードウェア耐用年数とのトレードオフ
  - ・ MPU、メモリ等の拡張性
  - ・ 筐体サイズとのトレードオフ

### (3)トレードオフのポイント

---

## 第 10 回 性能評価手法の分類(講義 90 分)

---

性能評価の要件、評価方法、評価手順について理解する。

### (1)性能評価の解析的手法

1. 確定的なモデル化
2. 確率的なモデル化
3. 解析の手順
  - ・ モデルの構築
  - ・ モデル上での解析
  - ・ モデルの検証

### (2)シミュレーション

1. シミュレーションによる性能評価
2. 手順
  - ・ システム構成をモデル化
  - ・ シミュレーションで評価する項目の決定
  - ・ 実際の構成との差分の確認
  - ・ 実験計画立案
  - ・ シミュレータの準備
  - ・ シミュレーション実施
  - ・ 実験結果の評価
3. ハードウェアシミュレータ
4. ソフトウェアシミュレータ

### (3)モニタリング

1. ハードウェアモニタ
2. ロジックアナライザ
3. プロトコルアナライザ
4. ICE

### (4)CPU の性能評価

1. CPU アーキテクチャの性能評価項目
  - ・ CPU のビット数
  - ・ 実行サイクルのクロック周波数
  - ・ CISC か RISC
  - ・ アドレス空間
2. CPU の処理速度
3. メモリアドレッシング方式
4. 資源規模に関わる性能

5. MIPS と MFLOPS
6. ベンチマークテスト
7. 応答性の評価

---

## 第 11 回 拡張性の評価(講義+ワークショップ 90 分)

---

拡張性評価の要件の実現方法・確認方法について理解する。

### (1)拡張性の要求事項

1. 機能の追加
  - ・ CPU の処理範囲の拡張
  - ・ メモリ資源
  - ・ 処理時間の余裕評価
2. ソフトウェアの構成の追加
3. 規模の拡張
  - ・ 処理対象の拡張
  - ・ システムが取り扱う限界の制御対象の拡張

### (2)拡張性トレードオフの評価

- ・ トレードオフの評価
- ・ トレードオフの評価の時期

### (3)評価の視点

1. 個々のシステム資源の評価
  - ・ 機能分担したハードウェア部品
  - ・ プログラムなどシステム資源
2. システム資源間のインタフェース
  - ・ ハードウェアやソフトウェアに関する資源間での情報
  - ・ タイミングの問題
  - ・ インタフェース仕様での誤り

### (4)設計ワークショップ

---

## 第12回 システム資源のトレードオフ(講義+ワークショップ 90分)

---

システム資源のトレードオフの概念と方法論及びそのポイントについて理解する。

(1)ソフトウェアとハードウェアのトレードオフのパターン

(2)性能向上

1. 複雑な制御方式の部分は、ソフトウェアで組み込む。
2. 処理能力に優れるハードウェア部品を使用する
  - ・ 電子部品や、メカ部品に分担
  - ・ 処理時間の厳しいアナログ信号の処理
  - ・ ハードウェア処理化

(3)信頼性向上

1. ハードウェア部品数の削減
2. ソフトウェア処理を単純化
3. 寿命の長い部品の採用

(4)コスト低減

1. 開発コストを削減する
  - ・ 開発規模を、最小限にする構成
  - ・ 多機能な機能デバイス
  - ・ 汎用的なマイコンボードの採用
  - ・ 開発方法の工夫によるコスト削減
2. 部品コストを削減する場合
  - ・ 部品点数を削減
  - ・ 専用のハードウェア部品採用

(5)設計ワークショップ

---

## 第 13 回 基本ソフトウェアと応用ソフトウェアのトレードオフ(講義+ワークショップ 90 分)

---

組み込みシステム最適化でもっとも重要な設計方法である基本ソフトウェアとアプリケーションとの間での拡張性、性能向上のためのシステム構成要素や実現機能、資源のトレードオフの概念と方法及びそのポイントについて方法、手順とその内容を理解する。

- (1)基本ソフトウェアを活用して、目的の機能を実現
  - 1. 汎用的な機能を組み合わせ
  - 2. 抽象化(汎用化)された機能を活用する。
- (2)基本ソフトウェアの活用で要求された機能が得られるように
  - 1. 基本ソフトウェアの処理速度を見積もる。
  - 2. メモリ資源の余裕を持たせる。
- (3)信頼性を向上するために
- (4)開発コストを削減するために
  - 1. ソフトウェアの既存の機能の活用
  - 2. ソフトウェアに付随する開発環境を活用
  - 3. ソフトウェアのライセンス料金の考慮
- (5)設計ワークショップ

---

## 第 14 回 組み込みシステム最適化のための方式設計(講義+ワークショップ 90 分)

---

組み込みシステムの性能向上のためのシステム方式設計の基本概念と設計方針などを理解する。

- (1)システムの要件定義
  - 1. リアルタイム性
  - 2. クリティカル性
  - 3. ハードウェア容積からの問題
  - 4. 使用条件からの問題
- (2)マイコン構成からの方式選択
  - 1. シングルチップ MPU の利用
  - 2. シングルチップマイコンの機能
- (3)ネットワーク機能からの方式設計
- (4)設計ワークショップ

---

## 第 15 回 最適化のための検討項目(講義+ワークショップ 90 分)

---

組み込みシステムを最適化するためのハードウェア、ソフトウェア、ネットワーク、設計方法など、各要素ごとの検討項目などを理解する。

### (1)ハードウェアによる最適化とトレードオフ

1. MPU の選択基準にかかわる諸要因
2. ハードウェア的視点からの検討項目
3. ソフトウェア的視点から検討項目

### (2)ソフトウェア設計方法論の考え方

1. オブジェクトの独立性を基準とした設計
2. カーネルやデバイスドライバの扱い
3. パフォーマンスを中心とした旧来の考え方
4. 資源の能力の配分

### (3)設計ワークショップ

以上