

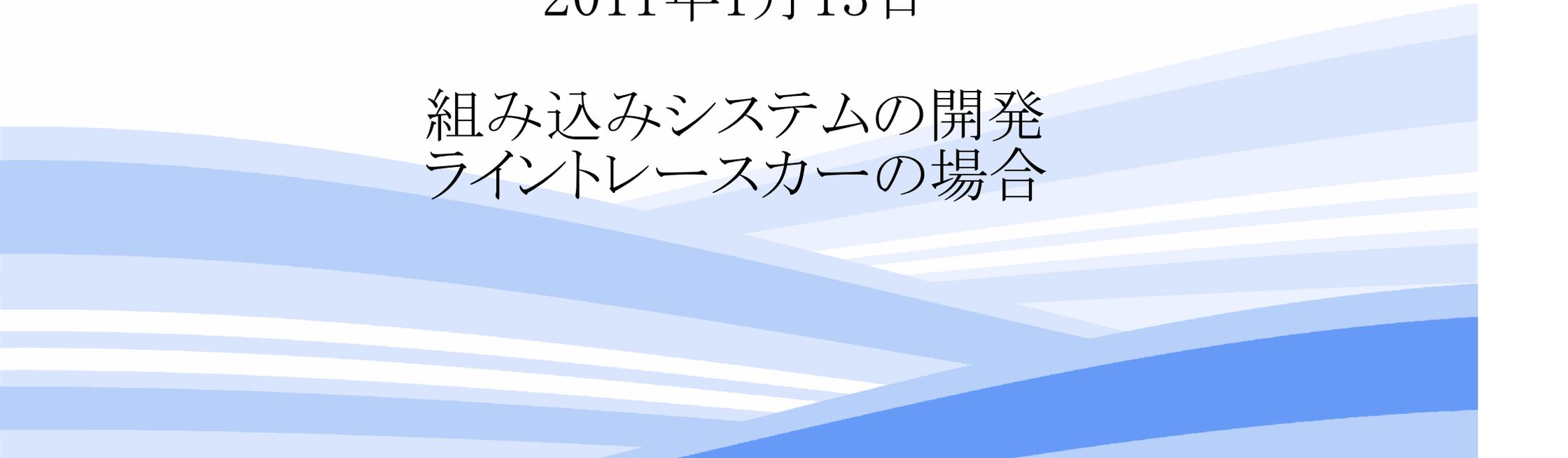


ハードウェア実験

組み込みシステム入門 第14回

2011年1月13日

組み込みシステムの開発
ライトレースカーの場合



組み込みシステムの開発(1)

- ▶ ライトレースカーという題材を用いて、組み込みシステムの開発の流れを理解する。
- ▶ 「仕様策定」 -- どんなものを作るか明確にする
 - 企画書/要求仕様書など、概要がわかる書類
- ▶ 「基本設計」 -- ハードウェア、ソフトウェアの概要を策定する。
 - XMOS+ライトレースカーという前提で作る。
 - ハードウェアの仕様書から、できることやできないこと、利点、価格などの概要を見積もる。
 - 比較して、最適なCPUなどを決める。

組み込みシステムの開発(2)

- ▶ 「詳細設計」 - 実際にものを作る場合の、詳細を決めていく。
 - 配布資料 - 「ライトレースカー設計仕様V1.0」参照のこと
 - まだプログラムはせずに、理論的に詰められるだけ煮詰めて検討しておく。
- ▶ 「プログラム設計」 - プログラムの構成を決める
 - モジュール単位でインターフェースを決める
 - データの受け渡し、モジュールごとの処理決定
- ▶ 「コーディング」 - プログラムを書く
 - 書きながら、モジュール単体のテストを行う。

組み込みシステムの開発(3)

- ▶ 「テスト・検証」 -- 動作のテスト、検証
 - テスト手順書の執筆/テスト結果報告書の作成
 - 「上限」テストを忘れない。
 - データ点数が最大になった場合の処理
 - 日付をまたぐ場合の処理(2000問題?)
- ▶ 「オペレーション」 - 運用 / 結果のフィードバック
 - 作ったシステムの改良点などを明確にする。

開発環境とデバッグ

- ▶ プログラム本体にテスト記述を組み込む場合
- ▶ 表示を切り替えられるように配慮する。
 - C言語系(≒XC)の場合の例

```
24
25 #define DEBUG_MESSAGE 0x0020 // デバッグ用のメッセージ出力
26 // 0x0001 -- alarmTimerの動作確認
27 // 0x0002 -- 受信したキーコードを表示する
28 // 0x0004 -- ボタン反応を表示する。
29 // 0x0008 -- ライン反応データの受信
30 // 0x0010 -- ボタンの状態変化を表示
31 // 0x0020 -- 動作モードの変化を表示
32 // 0x0040 -- モーター駆動制御の表示
33 // 0x0080 -- アラーム設定の表示
34 // 0x0100 -- ライントレースモードの判定出力
35
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285 #if (DEBUG_MESSAGE & 0x0020)
286     printf( "stateMachine: changed to LINE TRACE mode.\n" );
287 #endif
288     ledMode = 0x01;
289     led = (ledLines|ledMode);
290     ledOut <: led; // 送信
291
if( key == MANUAL_BTN_EXIT ){ // 5秒入力なし長押し
mode = MODE_LINE_TRACE; // ライントレースモードに移行
move <: 0; // 最初は車輪を止めておく
modeChange <: mode;
setBlink <: ALARM_OFF; // 点滅の中止

// 動作記録
if( recording ){
recording = 0; // 動作記録の終了
recordTimer :> endTime;
if( endTime < startTime ){
duration = INTEGER_LIMIT - startTime + endTime
} else {
duration = endTime - startTime;
}
eventPeriod[traceCounter] = duration;
traceCounter++;
}
}
```

🚩 掲示されているプログラムについて

Itcpmain-8.xc

- ▶ 仕様書通りに作成したプログラムですが、プログラムモード(記録された動作を逆にたどる)の部分のデバッグが完了していません。
 -
- ▶ 仕様書通りで、全ての「サブプログラム」をスレッドとして作成しましたが、スレッドにする必要のない(同時並行の動作をさせる必要がない)モジュールがいくつかあります。
 - パフォーマンスに影響するため、「関数」に書き換える必要があります。



最終回の課題

- **【これまでの課題】**
- 掲示されたプログラムを参考にして、これまでの課題で提出できるものを提出して下さい。

- **【今回の課題】**
-
- 第11回の課題を発展させて、自分の設計による「ライトレースカー」の動作仕様書を作成し、プログラムして、その動作を検証して下さい。
-

課題の実施において...

- ▶ 掲示のプログラムは、部分的あるいは、全体をそのまま再利用しても構いません。
- ▶ ただし、個々の部分の動作などについては、説明できる限り説明して下さい。
- ▶
- ▶ オリエンテーリングの動作では、「ライトレースモード」の動作で、直線の最終端の処理で改良の余地があります。



最終回分の別課題

- ▶ **【別課題1】**
- ▶ 掲示されたサンプルプログラムをインストールしたライトレースカーで、「取扱説明書」を作成して下さい。

- ▶ **【別課題2】**
- ▶ 掲示されたサンプルプログラムで、「テスト手順書」と「テスト結果報告書」を作成して下さい。