

**組み込みソフトウェアにおける
ソフトウェアアーキテクチャの事例研究**
副題: 車型ロボット制御ソフトウェアの設計, 実現
99B588 長野雄二
指導教員 野呂昌満

1 はじめに

組み込みソフトウェアをオブジェクト指向で開発した場合、並行処理や同期処理などの、ハードウェアコントロールや時間制御に関する処理が混在した構造になる場合がある。このような処理の混在するシステムをモジュールに分割することは困難であり、汎用性、再利用性を低下させる要因となっている。オブジェクト指向では、複数のオブジェクトにまたがる並行処理や同期処理などの異なる大域的な特性を、オブジェクトとして矛盾なく取り扱うことは困難である。アスペクト指向[1]を用いて、これらの特性をアスペクトとして整理することで、より良いモジュール分割が得られると考えられている。

本研究の目的は、組み込みソフトウェアにアスペクト指向を適用し、ハードウェアコントロールと時間制御に注目したソフトウェアアーキテクチャを提案することである。組み込みソフトウェア開発にソフトウェアアーキテクチャを用いることで、構造整理、実現手段、開発プロセスが規定されるので、生産性の向上につながることを期待できる。本研究では、事例として車型ロボット制御ソフトウェアのアーキテクチャを提案し、アーキテクチャをもとにアスペクト指向を用いて試作する。試作したソフトウェアから組み込みソフトウェア特有のアスペクトを抽出し、モジュール化する。研究の結果、組み込みソフトウェアのハードウェアコントロールと時間制御に注目したアーキテクチャを提案し、アスペクト指向を用いてモジュール化することができた。

組み込みシステムの題材として、Javaを用いてオブジェクト指向開発が可能なlejOS[2]が提供されているロボット制御実験キットのLegoMindStorms[3]を用いる。

2 アーキテクチャの提案

車型ロボット制御ソフトウェアの構造を整理し、アスペクト指向を用いたアーキテクチャを提案する。ソフトウェアアーキテクチャはソフトウェアの構造、実現方法、実現プロセスを示唆するものである。

アスペクト指向では、分割を規定するモジュールの基準をコンサーンと呼ぶ。ハードウェアコントロールと時間制御に注目して、時間に関するコンサーン、入力に関するコンサーン、出力に関するコンサーンとし、全体の制御をおこなうロボットをアプリケーションロジックコンサーンとして分割した。

組み込みソフトウェアにおいて、ハードウェアコントローラや非同期なメッセージ通信が必要であると考え、構成要素間の関係を整理し、組み込みソフトウェアのアーキテクチャとしてあげる。

組み込みソフトウェアのアーキテクチャ

- ハードウェアと仮想ハードウェア
構成要素の物理的なハードウェアを直接制御せずに、仮想的なハードウェア(仮想ハードウェアとよぶ)を用いて制御することでハードウェアを制御する構造を考えた。仮想ハードウェアオブジェクトは、制御の中心となるオブジェクトのコンポジットオブジェクトとなる。
- 仮想ハードウェアオブジェクトを並行オブジェクトとする
制御の中心となるオブジェクトと複数の仮想ハードウェアでは、並行処理、同期処理が必要であると考え、非同期なメッセージ通信を可能にする能動的な並行オブジェクトとして実現する。
- 仮想ハードウェアごとに状態遷移機械を用いる
並行オブジェクトである制御の中心のオブジェクトと仮想ハードウェアごとに状態遷移機械をもたせることでハードウェアごとの状態を管理する。状態遷移機械はハードウェアによって状態を遷移し、状態に応じた処理をおこなう。

上記より提案したアーキテクチャを図1に示す。

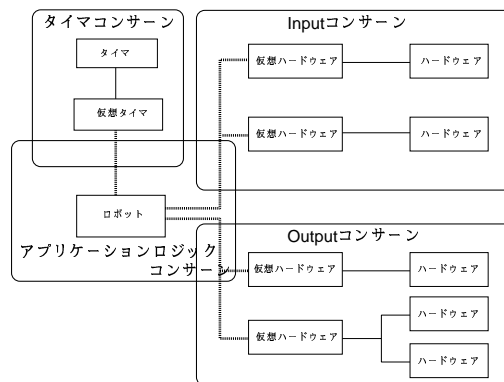


図1: 車型ロボット制御ソフトウェアのアーキテクチャ

3 障害物をよけて走行するロボット制御ソフトウェアの開発

車型ロボット制御の例として“障害物をよけて走行する制御”を考えた。提案したアーキテクチャをもとに、アスペクト指向を用いて分析、設計、実現する。

“障害物をよけて走行する”車型ロボット制御ソフトウェアに対するシステムの要求は、障害物をよけて走行すること、リモコンが押されたら一時停止、離されたら再発進をおこなうことである。ほかにハードウェア制約、環境条件を分析した。提案したアーキテクチャにもとづいて、分析した構成要素の仮想ハードウェアを考えた。関係を図2に示す。

図2に示した結果にもとづき設計、実現する。ロボット、仮想ハードウェアに状態遷移機械をもたせ、オブジェクトごとに状態を管理した。各仮想ハードウェアにアクティブオブジェクトパターン[4]を用いることで、非同期なメッセージを処理できる能動的な並行オブジェクトを実現する。ロボット、ゴール検知器、障害物検知器、リモコン、走行タイマを能動的な並行オブジェクトとして実現した。

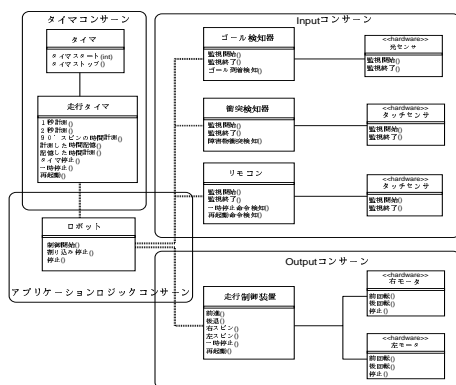


図2: クラス図

4 考察

提案したアーキテクチャと試作した事例をもとに、抽出したアスペクトについて考察し、組み込みソフトウェアのアスペクト指向開発について考察する。

車型ロボット制御のアスペクト

今回の事例では、提案したアーキテクチャをもとに車型ロボット制御ソフトウェアのハードウェアの制御目的に応じて仮想ハードウェアを用意し、アスペクトとして、出力と入力、時間制御アスペクトを抽出した。本研究で提案したアーキテクチャでは、ハードウェアの制御目的に応じて仮想ハードウェアを用意することで、ハードウェアを直接制御せず、仮想ハードウェアで制御する構造をつくった。制御目的に応じて、必要な仮想ハードウェアを追加、変更することで、ハードウェアの制御目的に応じた変更にも柔軟に対応できる。この構造で仮想ハードウェア同士に協調関係をもたさないようにすることで、ハードウェアの変更や追加があった場合に、他のハードウェアが影響を受けにくい構造にすることができた。

アスペクトとして抽出して構造を整理することで、ハードウェアや仮想ハードウェアに変更、追加があった場合、アプリケーションロジックを変更することなく、アスペクト間のアドバイスを変更することで対応できる。例えば、今回の事例でロボットの状態を表示する機能を追加することを考える。出力のハードウェアとしてモニターを追加し、仮想ハードウェアを追加する。ロボットとモニター間のメッセージ通信、すなわち出力アスペクトのアドバイスを追加することで実現できる。同様に時間に関する制御の追加がおこなわれた場合、時間制御アスペクトのアドバイスを追加することで実現できる。提案したアーキテクチャにもとづいたソフトウェアにおいて、ハードウェア制御の追加、変更をおこなう場合、抽出したアスペクトによって、変更箇所を局所化することができ、変更が容易である。したがって変更に強い構造のソフトウェアをつくることのできる。

アスペクト指向を用いた組み込みソフトウェア開発

本研究の事例にアスペクト指向を用いることで、コンサーンとして分割し、モジュール化できた。オブジェクト指向を用いた場合、モーターの起動処理、走行時間の計測処理、各センサからのイベントの待ち受け処理が混在する。結果、モジュールとして分割することが困難であり、構造として整理できない部分がある。事例では障害物を検知すると、障害物の右側からよけるように設計したが、左側からよけるように設計した場合にも、コンサーンによって分割されたモジュールのインターフェイスを利用することで異なる動作を設計することが可能である。オブジェクト指向ではオブジェクトの変更が必要であるが、アスペクト指向ではアドバイスの記述の変更をおこなうことでモジュール内の構造を変えることなく再利用できる。モジュール化することでオブジェクト単位での再利用性が向上し、生産性、信頼性の向上が期待できる。

5 おわりに

本研究では、車型ロボット制御ソフトウェアのアスペクト指向アーキテクチャを提案した。アーキテクチャをもとにハードウェアコントロールと時間制御のアスペクトとして整理し、モジュール化することで汎用性、再利用性を考慮した組み込みソフトウェアの構造を得ることができた。

参考文献

- [1] 野呂昌満: “プログラミング言語から見たアスペクト指向開発技術”, ソフトウェアシンポジウム2002.
- [2] leJOS Java for the RCX, <http://lejos.sourceforge.net/>.
- [3] Lego Mindstorm, <http://mindstorms.lego.com/>.
- [4] Douglas Schmidt, Michael Stal, Hans Rohnert, Frank Buschmann: “Pattern-Oriented Software Architecture, Volume2: Patterns for Concurrent and Networked Objects”, John Wiley & Sons.