自動車制御ソフトウェアにおけるアーキテクチャの構築

2002MT071 太田 将吾　　2002MT073 坂本 樹美　　2002MT096 安江 基規
指導教員　野呂 昌満

1　はじめに

自動車制御ソフトウェアのような，高い信頼性が要求される組み込みソフトウェアでは，耐故障性を考慮する必要がある．耐故障性などの非機能特性は，ソフトウェア全体に散在する処理となり，ソフトウェアの構造を複雑にする．ソフトウェア全体に散在する処理を分離する技術として，アスペクト指向技術 [1] が注目されている．

一方で，本研究室では，組み込みソフトウェアのアスペクト指向ソフトウェアアーキテクチャスタイル (以下，E-AOSAS) を提案してきた．E-AOSAS では，アスペクト指向ソフトウェアアーキテクチャ (以下，AOSA) を並行に動作する状態遷移機械の集合として規定する．並行に動作する状態遷移機械は，並行処理，状態遷移，アプリケーションロジックのアスペクトで構成される．

E-AOSAS では，耐故障性に関する処理をアスペクトとしてモジュール化する仕組みを提供していない．耐故障性に関する処理は，横断的な処理としてソフトウェア全体に散在してしまう．

本研究の目的は，E-AOSAS に，耐故障性に関する処理をアスペクトとしてモジュール化する仕組みを提供し，応用可能性について考察することである．耐故障性に関する処理をアスペクトとしてモジュール化し，アーキテクチャ構築の工程で適正に取り扱うことを可能にする．

本研究は，以下のように進める．

- E-AOSAS に基づき，自動車制御ソフトウェアの AOSA の構築
- 構築した AOSA から，耐故障性に関する処理をアスペクトとしてモジュール化し，AOSA の再構築
- 再構築した AOSA を一般化して，E-AOSAS+ を提案
- E-AOSAS+ の応用可能性について考察

本研究では，E-AOSAS に耐故障性に関する処理をアスペクトとしてモジュール化する仕組みを提供した．E-AOSAS+ を用いることで，アーキテクチャ構築の工程で，耐故障性に関する処理をアスペクトとして取り扱うことが可能となる．

太田はおもに自動車制御ソフトウェアのアスペクト指向実現，坂本はおもにアーキテクチャの再構築，安江はおもに E-AOSAS+ の提案を担当した．

2　組み込みソフトウェアのアスペクト指向ソフトウェアアーキテクチャスタイル

E-AOSAS は，組み込みソフトウェアのアーキテクチャを，並行に動作する状態遷移機械の集合として規定している．E-AOSAS は，本研究室で組み込みソフトウェアのアーキテクチャを構築してきた経験から提案されている．

並行に動作する状態遷移機械

複数の状態遷移機械は，たがいにメッセージを送り協調動作する．それぞれの状態遷移機械を並行処理実体として実現する．状態遷移機械は，待機状態と活性状態があり，システム起動時に待機状態になる．待機状態の時に，他の状態遷移機械からメッセージを受け取ると活性状態になる．活性状態となると，状態遷移機械はその動作を実行し，再び待機状態にもどる．並行に動作する状態遷移機械を実現する工程において，以下のコンサーンが確認されている．

- 主要コンサーン
  - 並行処理コンサーン
- 二次コンサーン
  - 状態遷移コンサーン
  - アプリケーションロジックコンサーン

並行に動作する状態遷移機械において，並行処理は記述言語などのプラットフォームに依存した処理となるので，状態遷移機械に関する処理と分離した．分離した状態遷移機械では，状態遷移に関する処理と，振る舞いに関する処理が横断的に関連する．状態遷移に関する処理と，振る舞いに関する処理の再利用性を高める目的で，状態遷移に関する処理と，振る舞いに関する処理を分離した．

アスペクトの関連を**図 1** に示す．アーキテクチャの記述には，本研究室で提案されているアスペクト指向ソフトウェアアーキテクチャの図式表現 [3] を用いた．

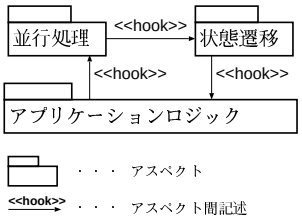


図 1　E-AOSAS のアスペクトの関連

並行処理は状態遷移と，状態遷移はアプリケーションロジックと，アプリケーションロジックは並行処理と，それぞれ関連しあう．各アスペクトに記述する処理を，以下に示す．

- 並行処理アスペクト
  状態遷移機械を並行に動作させる処理
- 状態遷移アスペクト
  状態遷移機械の状態の遷移に関する処理
- アプリケーションロジックアスペクト
  状態の遷移時の振る舞いに関する処理

3　耐故障性を考慮したアーキテクチャの構築

E-AOSAS に基づき，並行に動作する状態遷移機械の集合として，自動車制御ソフトウェアの AOSA を構築する．

3.1　耐故障性に関する処理のソフトウェア上での実現

耐故障性とは，システムに故障が発生したさいに，正常な動作を保ち続ける特性である．本研究では故障とは，誤ったイベントを受け取ることと定義する．

耐故障性を保証する方法は，ハードウェア上で保証する方法と，ソフトウェア上で保証する方法がある．ソフトウェア上で耐故障性に関する処理を実現する技法 [4] として，以下の技法がある．

- N バージョンプログラミング：非縮退型
- リカバリブロック：継続性縮退型
- N セルフチェッキングプログラミング：機能・性能縮退型

上記の技法は，故障発生時にシステムにおよぼす影響が異なる．非縮退型である N バージョンプログラミングは，許容範囲内の故障の発生に対して，機能の低下や時間の遅れを生じることなく耐故障性を保証できる．継続性縮退型であるリカバリブロックは，故障発生時に機能の低下を防ぐが，機能を復旧するための時間の遅れが生じる．機能・性能縮退型である N セルフチェッキングプログラミングは，故障発生時に時間の遅れはないが，一部の機能の停止や，性能の低下が生じる．

本研究では，ハードウェアが故障を起こした場合，ソフトウェア上で耐故障性を保証する方法を取り上げる．実現した自動車制御ソフトウェアでは，N バージョンプログラミングとリカバリブロックを適用した．本稿では，自動車制御ソフトウェアに N バージョンプログラミングを実現した場合の，アーキテクチャの構造について述べる．

3.2　ルームライトシステムの AOSA の構築

本研究では，自動車制御ソフトウェアの例としてルームライトシステムをとりあげる．車内のライトの制御を行うルームライトシステムは，入力装置にスイッチとドア，出力装置にライトをもつ．E-AOSAS に基づいて構築したルームライトシステムの AOSA を**図 2** に示す．

構築したルームライトシステムの AOSA は，入力，出力を管理するサブシステムと，ソフトウェア全体を管理するサブシステムで構成した．各サブシステムを並行に動作する状態遷移機械として実現し，サブシステム間の関連をアスペクト間記述に記述する．入力，出力を管理するサブシステムは，ハードウェアの状態を管理する．ソフトウェア全体を管理するサブシステムは，入力を管理するサブシステムの状態をもとに，ソフトウェア全体の状態を管理する．

3.3　耐故障性を考慮したルームライトシステムの実現

構築した AOSA をもとに，耐故障性を考慮したルームライトシステムを実現する．システムに耐故障性に関する処理を実現する方法として，以下の方法が考えられる．

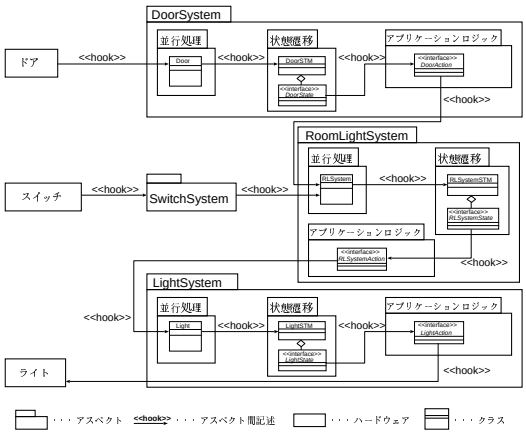


図 2　ルームライトシステムの AOSA

- 状態遷移機械にイベントを送る処理の前に追加
- 状態遷移機械にイベントを送る処理と置き換える

イベントを送る処理の前に追加して実現する場合，耐故障性に関する処理の追加を前提とした構造でソフトウェアを実現する必要がある．一方で，イベントを送る処理と置き換えて実現する場合，送信後のイベントを中継する構造として，耐故障性に関する処理を追加できる．イベントを送る処理と置き換えることで，耐故障性に関する処理を前提としない構造で実現できる．本研究では，耐故障性に関する処理を，システムが状態遷移機械にイベントを送る処理と置き換えて実現する．

実現したルームライトシステムでは，ハードウェアのアプリケーションロジックがシステムに入力を行う部分に N バージョンプログラミングを適用した．N バージョンプログラミングをオブジェクト指向実現するコードの構造を表現したクラス図を**図 3** に示す．

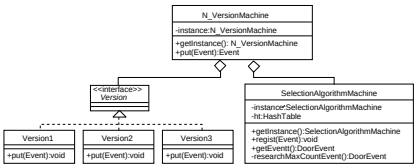


図 3　N バージョンプログラミングのクラス図

N_VersionMachine クラスは，Version オブジェクトと SelectionAlgorismMachine オブジェクトを管理する．Version クラスは，入力されたイベントの成否をチェックする．SelectionAlgorismMachine クラスは，Version オブジェクトの出力したイベントを比較し，結果を返す．

3.4　耐故障性に関する処理の分離

実現したルームライトシステムにおいて，耐故障性に関する処理が横断的に存在していることを確認した．

耐故障性コンサーンの確認

耐故障性に関する処理は，ハードウェアが状態遷移機械にイベントを送る処理と横断的に関連している．耐故障性に関する処理は，イベントを送る処理と置き換えて実行する．実現したルームライトシステムにおいて，耐故障性に関する処理が，複数のオブジェクトに横断してい

る様子を図 4 に示す．

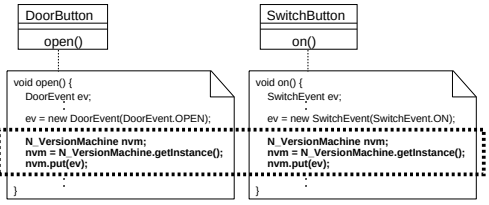
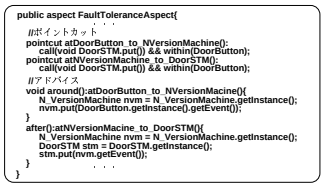


図 4　耐故障性に関する処理の詳細

耐故障性に関する処理の分離

耐故障性に関する処理を，システムが状態遷移機械にイベントを送る処理から，アスペクトとして分離する．アプリケーションロジックから，N_VersionMachine オブジェクトにイベントを送る処理は，アスペクト間記述に局所化した．アスペクト間記述と，N バージョンプログラミングに関する処理のシーケンス図を図 5 に示す．



アスペクト間記述

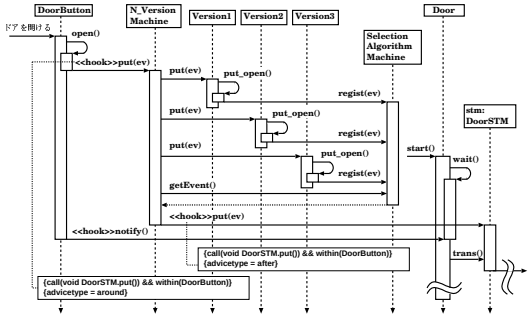


図 5　N バージョンプログラミングに関する処理

3.5　耐故障性アスペクトを取り入れた AOSA の構築

構築した AOSA に耐故障性アスペクトを取り入れ，ルームライトシステムのアーキテクチャを再構築する．耐故障性アスペクトを取り入れ再構築したアーキテクチャを図 6 に示す．

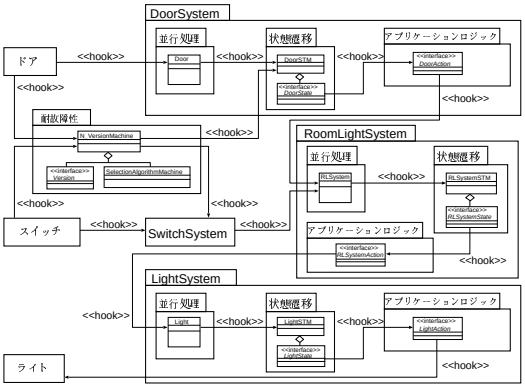


図 6　耐故障性アスペクトを取り入れたルームライトシステムの AOSA

構築したルームライトシステムの AOSA では，ハードウェアがシステムに入力を行う部分に耐故障性アスペクトを取り入れている．入出力システムとソフトウェア全体を管理するシステムで構成される自動車制御ソフトウェアのアーキテクチャは，図 6 のように耐故障性アスペクトを適用できる．

4　E-AOSAS+ の提案

耐故障性アスペクトを取り入れた E-AOSAS として，E-AOSAS+ を提案する．構築したルームライトシステムの AOSA において，耐故障性に関する処理はアスペクトとしてモジュール化している．E-AOSAS+ のアスペクトの関連を図 7 に示す．

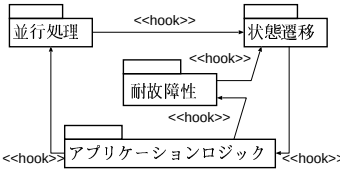


図 7　E-AOSAS+ のアスペクトの関連

耐故障性アスペクトは，主要コンサーンである並行処理アスペクトから独立しており，提案する E-AOSAS+ において二次コンサーンと位置づける．耐故障性アスペクトは，アプリケーションロジックアスペクトから状態遷移アスペクトにイベントを送るメッセージにオペレーションフックさせる．耐故障性アスペクトが実行される処理の流れを図 8 に示す．

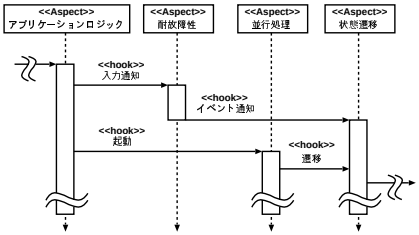


図 8　E-AOSAS+ のシーケンス図

耐故障性アスペクトは，アプリケーションロジックアスペクトで行う，状態遷移アスペクト内の構成要素にイベントを送る処理を中継する．耐故障性アスペクトには，アプリケーションロジックアスペクト内の構成要素からのメッセージを受け取るさいの，耐故障性に関する処理を記述する．耐故障性アスペクト内の構成要素は，耐故障性に関する処理を行った結果を，状態遷移アスペクト内の構成要素に送る．

5　考察

提案した E-AOSAS+ の応用可能性について考察する．E-AOSAS+ を用いて，他の自動車制御ソフトウェアに向けた AOSA と，他の応用領域に向けた AOSA を構築する．

5.1　他の自動車制御ソフトウェアに向けた AOSA の構築

提案した E-AOSAS+ をもとに，他の自動車制御ソフトウェアの AOSA を実現する．他の自動車制御ソフト

ウェアの例として，ABS(Anti-lockBrakeSystem) を実現する．自動車のブレーキを制御する ABS は，入力装置に車速センサー，車輪速センサー，ブレーキペダルをもち，出力装置にブレーキ圧力調整装置をもつ．耐故障性アスペクトは，車速センサー，車輪速センサーからシステムへの入力に取り入れた．実現した ABS の AOSA を**図 9** に示す．

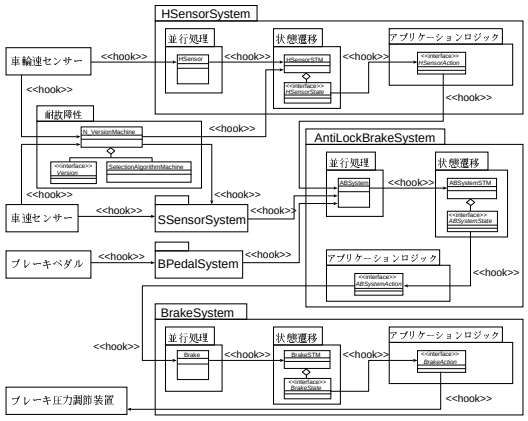


図 9　ABS の AOSA

提案した E-AOSAS+ に基づいて，耐故障性アスペクトを取り入れた ABS の AOSA を構築できた．ABS は，システムと入出力装置のそれぞれが状態遷移機械をもつ構造とし，入力の一部に耐故障性アスペクトを取り入れた．提案した E-AOSAS+ は，他の自動車制御ソフトウェアの AOSA の構築においても応用可能性が高いと予想できる．

5.2　他の応用領域に向けた AOSA の構築

提案した E-AOSAS+ をもとに，他の応用領域に向けた AOSA を構築する．他の応用領域の例として，携帯電話制御ソフトウェアの通話システムを取り上げる．通話システムとは，携帯電話の通話機能を実現するシステムである．通話システムは，入力にボタン，マイクロフォン，アンテナレシーバーをもち，出力にスピーカー，ディスプレイ，アンテナセンダーをもつ．耐故障性アスペクトは，ボタンとアンテナレシーバーからシステムへの入力の記述に取り入れた．構築した携帯電話制御ソフトウェアの AOSA を**図 10** に示す．

提案した E-AOSAS+ に基づいて，耐故障性アスペクトを取り入れた携帯電話制御ソフトウェアの AOSA を構築できた．携帯電話制御ソフトウェアの AOSA においても，アーキテクチャ構築の段階で耐故障性アスペクトを取り扱うことができる．提案した E-AOSAS+ は，他の応用領域に向けた AOSA を構築するさいにも，応用可能性が高いと予想できる．

6　おわりに

本研究では，E-AOSAS に基づいた自動車制御ソフトウェアの AOSA を構築し，耐故障性に関する処理を取り入れた．耐故障性に関する処理はアスペクトとしてモジュール化している．耐故障性アスペクトにおいて，

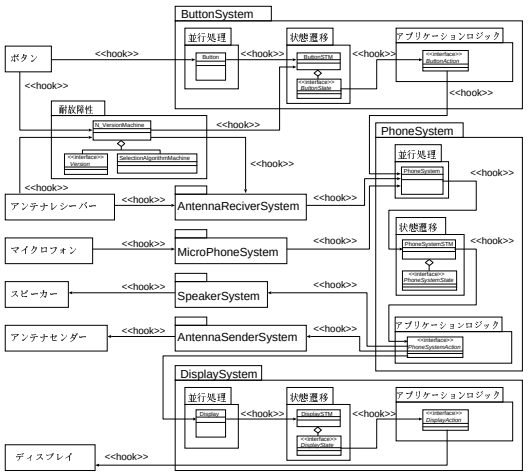


図 10　携帯電話制御ソフトウェアの AOSA

耐故障性に関する処理の実現技法に変更があった場合，変更箇所を耐故障性アスペクトに局所化できる．構築した自動車制御ソフトウェアの AOSA を一般化して，耐故障性アスペクトを取り入れた E-AOSAS として，E-AOSAS+ を提案した．提案した E-AOSAS+ の応用可能性の考察として，他の自動車制御ソフトウェアの AOSA と，携帯電話制御ソフトウェアの通話システムの AOSA を構築した．

今後の課題として，コンフィギュレーションの切替えの実現があげられる．例として，入力装置の状態によってシステムの構成を切替えることを考える．自動車制御ソフトウェアにおいてコンフィギュレーションの切替えを行うことで，システムの構成を整理する．

謝辞

本研究を進めるにあたり，二年間熱心に御指導をいただいた野呂昌満教授，有益なアドバイスをいただいた張漢明助教授，大学院生の石見知也さん，八木晴信さん，小久保佳将さん，石川智子さん，坂野将秀さん，本多克典さん，久松康倫さん，水野耕太さんに深く感謝いたします．また，二年間をともに励まし支え合い頑張ってきた野呂研究室一同の方々にも感謝いたします．

参考文献

[1] AspectJ，http://eclipse.org/aspectj/，Dec. 2005.

[2] T. Elrad, M. Aksits, G. Kiczales, K. Lieberherr, and H. Ossher, “Discussing Aspects of AOP,” *COMMUNICATION of the ACM*, Vol. 44，No. 10，pp. 33-38，2001.

[3] Java，http://java.sun.com/，Dec. 2005.

[4] 森貴彦，“アスペクト指向ソフトウェアアーキテクチャの図式表現に関する研究,” 南山大学大学院経営学研究科経営学専攻修士論文，p，67，2004.

[5] 向殿政男，フォールト・トレラント・コンピューティング，丸善株式会社，p. 248，1989.

[6] UML，http://www.uml.org/，Dec. 2005.