

# 組み込みシステム開発用多機能 behavior モデラーの開発（第一報） ~Statecharts、シナリオ及び機能モデルの連携による 組み込みシステムの挙動仕様記述手法の研究~

漆原 映彦<sup>†</sup> 金井 理<sup>†</sup> 岸 浪 建 史<sup>†</sup>

北海道大学大学院工学研究科

## 1. はじめに

PDA、携帯電話の様な組み込みシステムの開発において開発プロセスの効率化が強く望まれている。開発プロセスの効率化の一方法として UI(User I/F)部の設計の効率化が考えられる。

現在、Rapid<sup>[1]</sup>や PlayMO<sup>[2]</sup>の様な商用プロトタイプングツールがあるが、これらは挙動の仕様モデルをそれぞれのシステムのみで独自に使用しておりそれらは他システムでの再利用は困難である。また、ROSE<sup>[3]</sup>、Rhapsody<sup>[4]</sup>といった UML ベースの商用 CASE ツールでは、挙動仕様モデルは Statecharts で記述できるが機能仕様は Usecase 図しかない。そのため、機能仕様と挙動仕様間の論理的な関係づけが得られないため、製品機能の変更に対応した機能仕様や挙動仕様の管理が十分に行えない。また、Statecharts を設計者がゼロから記述する必要があるため作業量が大きくなってしまふ。

本研究では UML の開発方法論に製品機能のモデル化を行うための機能関係図を導入し、機能関係図と HI の挙動仕様を表す Statecharts との論理的関係づけを明らかにすることによって体系的な変換手順を明らかにする。

## 2. 機能仕様モデルから挙動仕様モデルへの挙動仕様記述手法の概要

図 1 に提案する挙動仕様記述手法の概要を示す。手順としては、まず 商品企画から製品の仕様が与えられる。企画書の製品の機能を詳細化を繰り返し、機能関係図で表す。機能関係図に示された機能間関係に基づいて Statecharts の状態階層性の枠組みを自動生成する。機能の詳細化と同時に UI の構成要素やその属性の定義を行う。詳細化した各機能に対して UI の構成要素を割り当てていく。割り当ては拡張した Usecase 図で表す。最も詳細化された機能を Statecharts 化するため、各詳細機能ごとにシーケンス図を記述する。シーケンス図上で動作シナリオを作成し、そのシーケンス図上で状態定義ルールを用いて状態を推定する。Statecharts の枠組みと で作成された詳細化された機能の動作シナリオを合成する事によって Statecharts が完成し、これに状態 - UI 対応表を組み合わせてシステム挙動仕様として出力する。また、機能関係図も製品機能仕様として同時に出力し、後のシステム開発に利用できるようにする。

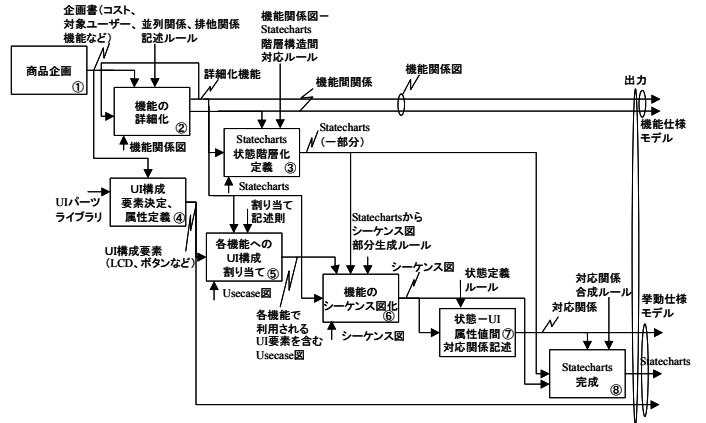


図1 本仕様記述手法の概要

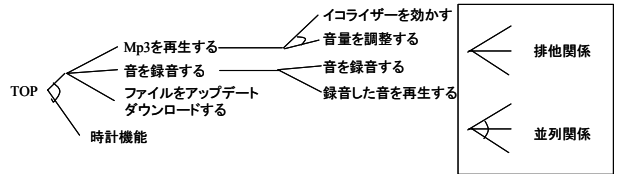


図2 mp3 プレイヤーの機能関係図

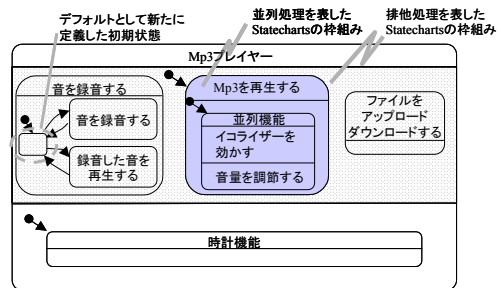


図3 図2の機能関係図から生成された一部の Statecharts

## 3. 機能の詳細化と Statecharts の状態階層化定義

まず商品企画で企画された機能を詳細化する。TOP の機能から下位機能にかけて機能の詳細化を繰り返し、これを機能関係図として木構造で表していく。ただし、おのおのの階層でそれぞれの機能の関係を並列機能、排他機能、親子機能関係で分類し、定義していく。mp3 プレイヤーの機能関係図の例を図 2 に示す。

この機能関係図を用いて Statecharts の状態階層の枠組みを定義していく。i)機能関係が親 - 子関係の場合・・・機能の階層性と 1 対 1 対応の階層状態を定義する。ii)同階層の機能関係が並列の場合・・・並列機能の状態は並行サブ状態の形を作る。iii)同階層の機能が排他の場合・・・いくつかの機能の中から一つを選択して動作を行うことになる。そのため Statecharts の形は別に分類された機能の中からイベントにより一つを選んで遷移する形になる。

A Research on Multi-Functional Behavior Modeler for Embedded Systems Development  
 Akihiko Urushihara, Satoshi Kanai and Takeshi Kishinami  
 Hokkaido University, kita-13 nishi-8, Kita-ku, Sapporo,  
 Hokkaido 060-8628, Japan

上記のような定義方法で図2の機能関係図から状態階層構造を自動決定し記述したものが図3である。

#### 4. UIの構成要素の決定と各機能への割り当て

商品の企画書などを参考にしてUIの構成要素を決定する。その際、UIのパーツライブラリを利用しそれぞれの構成要素がどのような属性を持つかも定義しておく。(例えば記憶式ボタンではONとOFFという値を持つ属性等。)

詳細化した機能に対して、上で定めたUIのどの構成要素を使用するかを割り当てる。割り当てには製品モデルにUIを加え、階層化した拡張Usecase図を使用する。Mp3プレイヤーにおける拡張Usecase図の例を図4に示す。

#### 5. Usecase図とStatechartsを利用した機能のシーケンス図化

3で定義された製品モデルのStatecharts、4で定義したUIを機能に割り当てたUsecase図を利用して詳細化された機能をシーケンス図で表す。

詳細化された機能を実行するシナリオの一部は3で定義したStatecharts上の遷移の経路となり得るため、機能のシーケンス図の一部はStatechartsを利用して自動決定することが出来る。

#### 6. 定義したシーケンス図による状態の定義

5で定義したシーケンス図に対して状態-UI属性間の対応関係を記述した表を利用して状態を定義していく。出力UI(LCDやスピーカーなど)はUIの属性値の組み合わせにより一意に状態を定めることが可能となる。実際にmp3プレイヤーの記述例で定義した状態-UI属性値間対応関係は表1のように表すことが出来、状態がS1~S5のように定義される。

この表を用いて5で作成したシーケンス図上でシステムコントローラの状態を識別する。また、UIの各パーツの属性を利用して遷移(入力と出力)も定義する。mp3プレイヤーの例(音を録音するという機能)で実際に定義した例が図5となる。

#### 7. Statechartsの完成

シーケンス図に状態、遷移の概念を追加したことによりシーケンス図で表現した詳細機能をStatecharts化することが可能となる。3で作成した枠組みに詳細機能のStatechartsを付加していくことによりStatechartsが完成する。また、排他関係の部分においてお互いの状態間の不足している遷移についてもいくつか加えておく。mp3プレイヤーについて完成したものが図6である。

#### 8. まとめ

本報告では、機能仕様のモデル化のために機能関係図を導入し、機能関係図と挙動仕様を表すStatechartsとの論理的関係を明らかにすることにより体系的な挙動仕様記述手法を提案した。また、具体的な例により有効性を確認できた。今後は本手法を自動的に行うことが可能なツールの開発を行ってきたい。

[参考文献]

- 1, <http://www.e-sim.com/> e-SIM Ltd.
- 2, <http://www.playmo.co.kr/> INUS Technology ,INC
- 3, <http://www.rational.com/> Rational Software Corp.

- 4, <http://www.ilogix.com/> i-Logix
- 5, Bruce Powel Douglass "Doing Hard Time ~ Developing Real-Time Systems with UML, Objects, Frameworks, and Patterns ~"
- 6, 渡辺他 "組み込みUML ~eUMLによるオブジェクト指向組み込みシステム開発~" 翔泳社(2002)
- 3, Bruce Powel Douglass "Real-Time UML ~ Developing Efficient Objects for Embedded Systems ~"

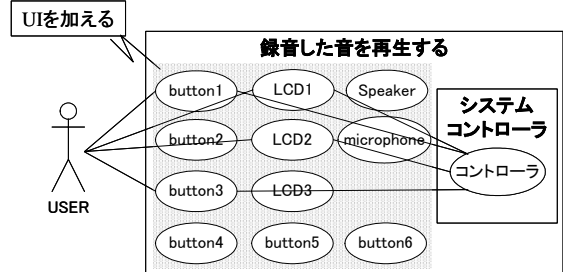


図4 Human I/Fの構成要素を割り当てたUsecase図例  
表1 状態とUI属性の対応関係

	LCD1	LCD2	LCD3
S1(Default)	Date	なし	Time
S2	"PLAY"	なし	Time
S3	"LOAD"	なし	なし
S4	Date	"MEMO"	Time
S5	"REC"	"MEMO"	Time
...	...	...	...

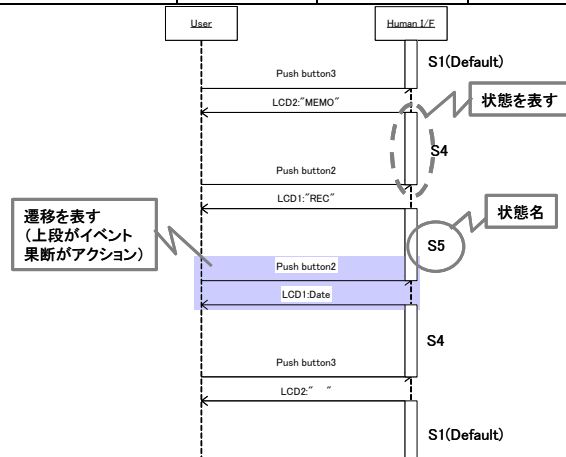


図5 シーケンス図にUI構成要素を加えた例

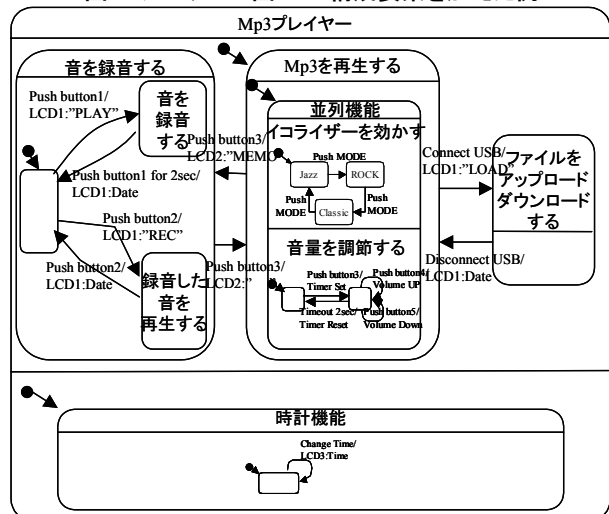


図6 完成したStatecharts例