



MONOist > 組み込み開発 > 状態遷移表による設計手法(4):状態遷移表を使用...

状態遷移表による設計手法(4):

状態遷移表を使用した設計モデル(拡張階層化状態遷移表) (1/3)

2012年09月24日 10時00分 公開

[塚田 雄一 キャッツ, MONOist]

印刷

通知

2

Share

21

はじめに

組み込みソフトウェアが抱える一番の課題は「設計品質の向上」です。本連載の主役「状態遷移表」であれば、“イベント”と“状態”の全ての組み合わせを捉えることができるため、「モレ」「ヌケ」のない品質の良い設計が可能です。そして、不具合発生による手戻りコストの削減や開発効率の向上にも役立ちます。

こうした理由から、組み込みソフトウェア開発の世界では、長年、状態遷移系モデルで設計が行われています。

前回 は、“状態遷移表を使用した要求分析モデル”をテーマに、要求仕様から状態遷移表を作成するプロセスを紹介しました。今回は、前回作成した要求分析モデルを基に、「設計モデル」を作成したいと思います。

設計モデルとは、プログラミングを意識し、より良い実装を行うために作成するモデルのことです。

ただし、通常の状態遷移表の表記の場合、プログラミングを意識したモデリングを行うには表現力が足りないので、状態遷移表設計手法では、「拡張階層化状態遷移表」による表記を使用します。

なお、本連載では以下の6つのテーマを順番にお届けしていきます。

1. (連載第1回):状態遷移表設計手法の概要
2. (連載第2回):なぜ状態遷移表を使うと、品質の良い開発ができるのか
3. (連載第3回):状態遷移表を使用した要求分析モデル
4. 状態遷移表を使用した設計モデル(拡張階層化状態遷移表)
5. 状態遷移表からの実装
6. 状態遷移表を使用したテスト手法

今回モデリングするキッチンタイマーの要求仕様範囲

前回 は、基本仕様部分の要求分析を行いました。基本仕様では、「分ボタン」「10秒ボタン」「クリアボタン」の3つのボタンを1つにまとめ、時間設定ボタンとして扱いました。

今回は、時間設定ボタンの詳細部分も設計していくため、「分ボタン」「10秒ボタン」「クリアボタン」をそれぞれ別のボタンとして扱っていきます。つまり、これらのボタンが押された際の各処理に

カスタム検索

Embedded

組み込み開発

見どころ情報や展示会速報、レポート記事をお届け!!

自動車技術展

人とくるまのテクノロジー展

2019 横浜

AUTOMOTIVE ENGINEERING EXPOSITION 2019 YOKOHAMA

MONOist

EE Times

EDN

スマートジャパン

詳しくはこちら

スポンサーからのお知らせ

> 【5月29日 東京開催】MONOist 主催セミナー
失われつつある「日本品質」という強み、
「攻めの品質管理」で生まれる新たな価値

Special Contents

日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ

プラットフォームにならないマイクロソフトの「CASE戦略」

現実を超えた仮想環境へ、自動運転時代に向けた最新モデルベース開発ツール

スマートファクトリー化でCC-Link IE TSNが果たすべき役割

コネクタ、センサーが実現する次世代モビリティ社会、「ホロレンズ」で体験

「ねじレス化」が生み出す価値、盤製作全体の効率化を目指す制御・配電盤革新

リアルタイムOS上でROSが動く、産業用機器へのOPC UAサーバ機能搭載も

ついて、個別に設計していくということです (図1)。

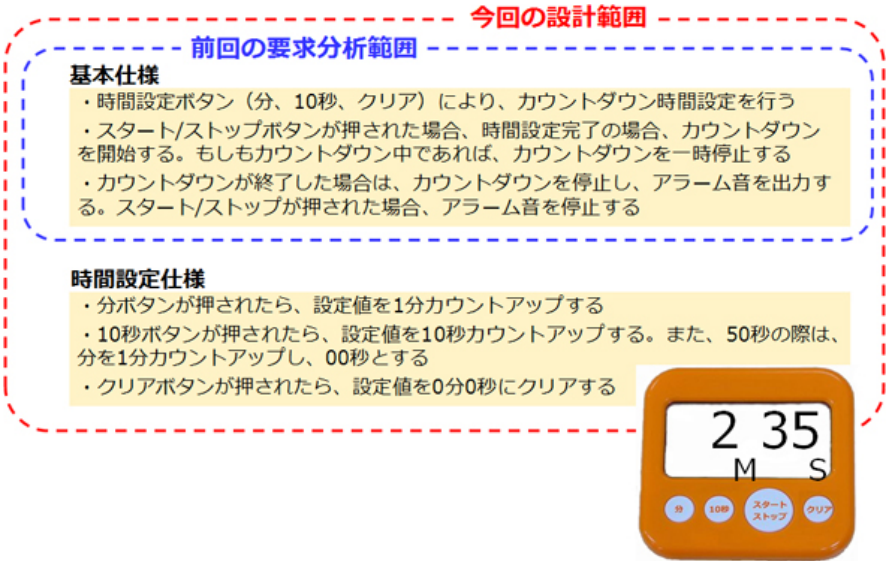


図1 今回の設計範囲について

前回作成したキッチンタイマーの要求分析モデル

図2は、前回作成したキッチンタイマーの要求モデルです。

今回は、この図2の要求分析モデルを基に設計モデルを作成していきます。

□0 キッチンタイマー	S	時間設定	カウントダウン	アラーム音出力
E		0	1	2
時間設定ボタン	0	カウントダウン時間設定 時間設定	/	/
スタートストップボタン	1	カウントダウン開始 カウントダウン	カウントダウン停止 時間設定	アラーム音停止 時間設定
カウントダウン終了	2	×	カウントダウン停止 アラーム音出力 アラーム音出力	×

図2 前回作成したキッチンタイマーの要求モデル

拡張階層化状態遷移表とは

今回使用する拡張階層化状態遷移表とは、組み込みソフトウェアの開発を行う際に必要な表現を、状態遷移表に追加したものです (図3)。

状態遷移表を機能拡張することにより、通常の状態遷移表では表現しきれないようなものに対応できます。例えば、組み込みソフトウェア開発では、C言語などを用いた構造化プログラミングによる実装がよく行われますが、拡張階層化状態遷移表であれば、構造化表現に対応できます。

さらに、通常の状態遷移表では、状態とイベントの全ての組み合わせを表現するため、表が巨大化してしましますが、階層化することにより、これを抑制できます。加えて、C言語の関数コールのようなトップダウン設計も可能です。

次世代Power over Ethernet規格「PoE++」対応機器を実現するチップセット

» Special 一覧

Special Site

- PR -

第4次産業革命をチャンスに
日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ

【Embedded Innovations】
マイコン/アナログ/メモリ最新情報を配信中。
組み込みの最新情報をチェック

LTC6560/LTC6561 TIA
アンプは、LIDARおよび
産業用画像処理向け

出力多重化機能付き
シングルおよび4チャンネル
トランスインピーダンスアンプ

AHEAD OF WHAT'S POSSIBLE™

メーカー公認
ディストリビュータ

詳細はこちら

コーナードリンク

Windows7サポート終了 対策ナビ

Windows 7 サポート終了 対策ナビ

Windows 10 IoT

FPGA

車載ソフトウェア

組み込み開発の記事ランキング

パナソニックがソフト開発体制強化へ
「製品を常にアップデート可能にする」

いまさら聞けないLPWAの選び方
【2019年春版】

HPCとAI性能を両立したポスト「京」の
CPU、ウエハーが初公開

人体通信で医療IoT、加速する医療機器の
モバイル化——MEDTEC Japan 2019レポート

CAN通信におけるデータ送信の仕組み
とは？

CANプロトコルを理解するための基礎知識

ROSロボット開発者向け開発管理環境
ユーティリティを公開

質量分析計のピークピッキングをAIで自動

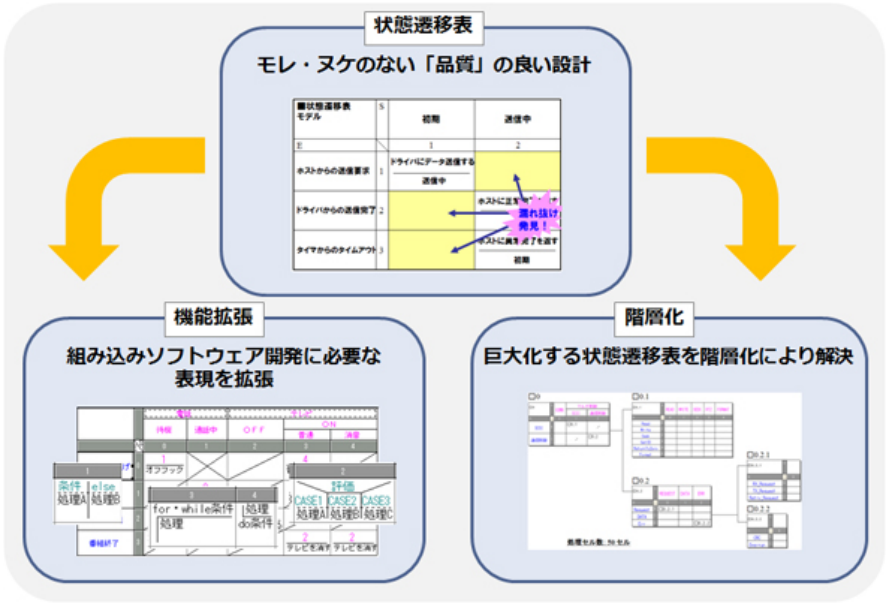


図3 拡張階層化状態遷移表とは

関連商品

拡張階層化状態遷移表設計手法

Ver.2.0-Embedded SEのための設計手法

今回紹介する設計モデル作成手順

図4に、今回紹介する設計モデルの作成手順を記します。流れとしては、イベント部分、状態部分、アクション部分、遷移部分、アクティビティの順に機能拡張を行い、最後に階層化を行います。

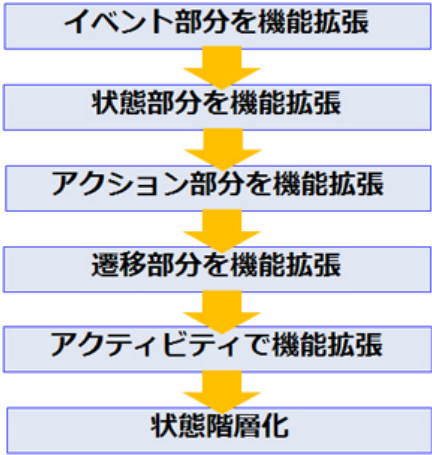


図4 設計モデルの作成手順

イベント部分の機能拡張例

イベントとは、遷移をもたらす外的刺激や内部変化などを表すものです。

イベントの種類には、イベントを待ち続け、イベントが来たらどのようなイベントが来たかを解析するメッセージ型と、常にイベントのポーリング参照を行い、変化を監視するフラグ型が存在します

- 化、島津製作所と富士通が共同開発
- ネクスティが販売パートナーになった QNX、マイコンレベルのプロセッサもカバー
- 日本初のAIプロダクト品質保証ガイドライン、QA4AIコンソーシアムが発行へ

よく読まれている編集記者コラム

- モノづくり魂集の どうして地下鉄のホームにLiDARがあるのか、東京メトロに聞いてきた
- 自分にとっての「ぴったり」とは？ 個人のモノづくりの価値をもう一度問う
- 「LOVOT」のプロダクトデザインから学んだ“仕事の流儀”

» 編集後記一覧

人気記事ランキング

提供 オートモーティブ・ジョブズ AUTOMOTIVE JOBS

【動画で解説】2019年度の採用開始！今年の転職トレンドは？

<高齢者事故対策>ペダル踏み間違い防止や逆走防止など様々なアプローチが登場

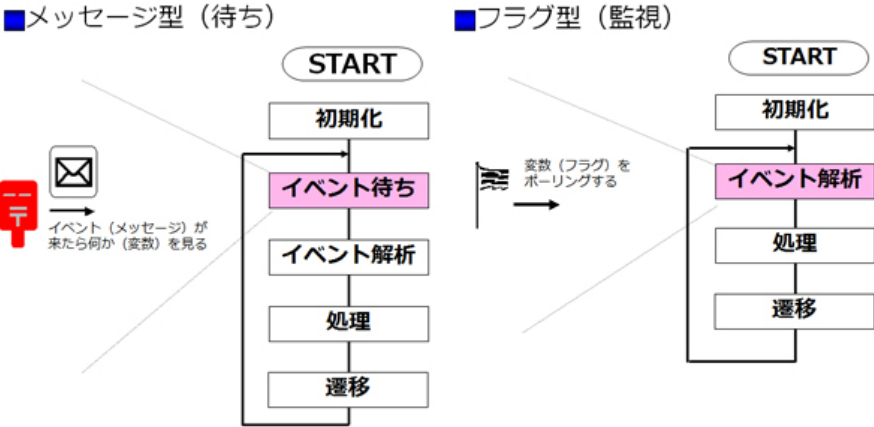
45%が「勝手にブレーキをかけてくれる」と認識 自動ブレーキにまつわる誤解とリスク

【ホンダ】HEV/PHEV用で熱効率40%超を達成したエンジン戦略

男性がつけていたら恥ずかしい「図柄入りご当地ナンバー」ランキング

» 他の記事を見る

(図5)。さらに、割り込みが発生した際のイベントに対応した**割り込み型**、デバイスドライバのシステムコールなどに対応した**関数コール型**などが存在します。



イベント部分では、複数のイベントを1つのまとまりとしてグループ化し、親子関係で表現するなどの機能拡張がなされています。

キッチンタイマーモデルのイベント部分を機能拡張してみよう

それでは、キッチンタイマーモデルのイベント部分を機能拡張してみましょう。

前回作成した要求分析モデルでは、前述の通り、時間設定ボタンを1つのイベントとしてまとめて表現していました。設計モデルでは、各ボタンの処理を全て設計するため、時間設定ボタンを「分ボタン」「10秒ボタン」「クリアボタン」とし、詳細のイベント表現を記述します。図6に変更した状態遷移表を示します。

時間設定仕様

- ・分ボタンが押されたら、設定値を1分カウントアップする
- ・10秒ボタンが押されたら、設定値を10秒カウントアップする。また、50秒の際は、分を1分カウントアップし、00秒とする。
- ・クリアボタンが押されたら、設定値を0分0秒にクリアする

□キッチンタイマー		時間設定	カウントダウン	アラーム音出力
E		0	1	2
時間設定ボタン	分ボタン			
	10秒ボタン			
	クリアボタン			
スタートストップボタン		カウントダウン開始 カウントダウン	カウントダウン停止 時間設定	アラーム音停止 時間設定
カウントダウン終了		×	カウントダウン停止 アラーム音出力 アラーム音出力	×

図6 変更した状態遷移表(イベント部分)

状態部分の機能拡張例

状態とは、システムの状況や特性を表すものです。

状態には、各状態が排他的に存在している「**排他状態**」があります。状態部分では、複数の状態を1つのまとまりとしてグループ化し、親子関係で表現するなどの機能拡張がなされています。

実態状態フレームでは、アクションに遷移すると親のアクションを実行後に子のアクションが実行されます。

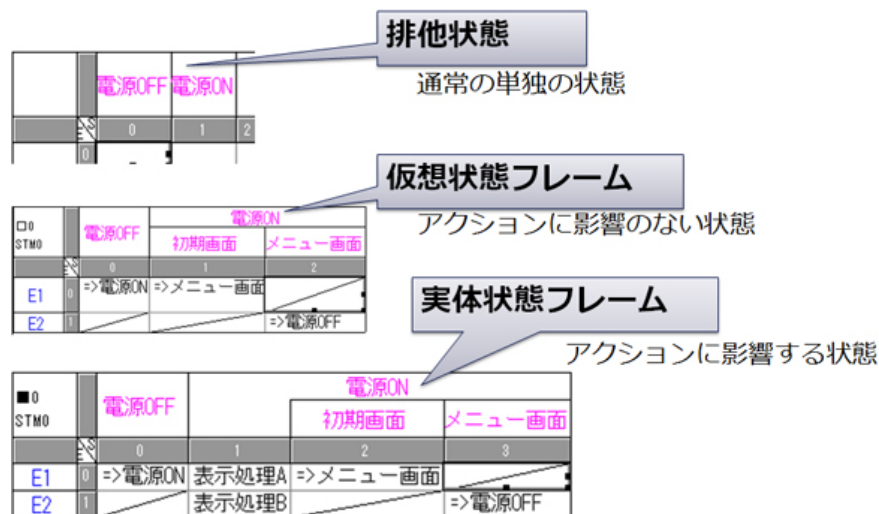


図7 状態の種類について

キッチンタイマーモデルの状態部分を機能拡張してみよう

それでは、キッチンタイマーモデルの状態部分を機能拡張してみましょう。

前回作成した要求分析モデルでは、全て排他状態で、それぞれの状態として表現していました。ここでは、「カウントダウン」状態と、「アラーム音出力」状態を1つのグループとして、仮想状態フレームで表現したものを示します(図8)。

□ キッチンタイマー		S	時間設定	時間計測	
E			0	1	2
時間設定ボタン	分ボタン	0			
	10秒ボタン	1			
	クリアボタン	2			
スタートストップボタン		3	カウントダウン開始 カウントダウン	カウントダウン停止 時間設定	アラーム音停止 時間設定
カウントダウン終了		4	×	カウントダウン停止 アラーム音出力 アラーム音出力	×

図8 変更した状態遷移表(状態部分)

関連キーワード

ソフトウェア | モデルベース開発 | 設計 | 組み込み | 組み込みソフトウェア | UML | 組み込みシステム | モデリング

→ 次ページ アクション部分の機能拡張例

MONOist 主催セミナー

製造業×品質、転換期を迎えるモノづくりの在り方

開催日: 2019/5/29 (水)

会場: 野村コンファレンスプラザ日本橋

無料

MONOist > 組み込み開発 > 状態遷移表による設計手法(4):状態遷移表を使用...

状態遷移表による設計手法(4):

状態遷移表を使用した設計モデル(拡張階層化状態遷移表) (2/3)

2012年09月24日 10時00分 公開

[塚田 雄一 キャッツ, MONOist]

前のページへ

123

次のページへ

印刷

通知

2

Share

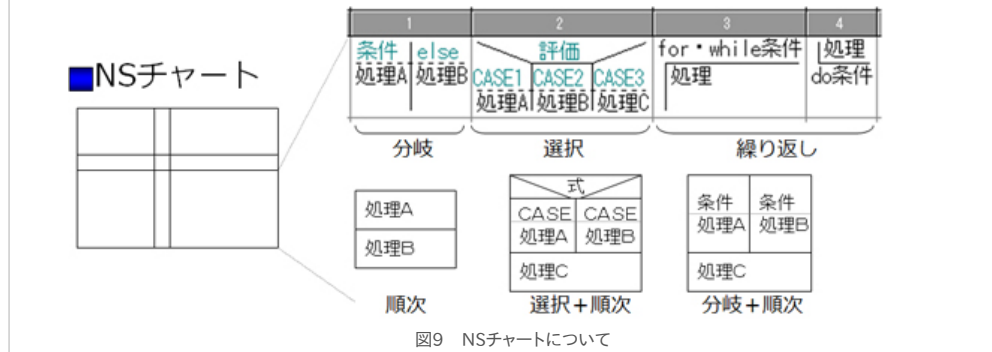
21

アクション部分の機能拡張例

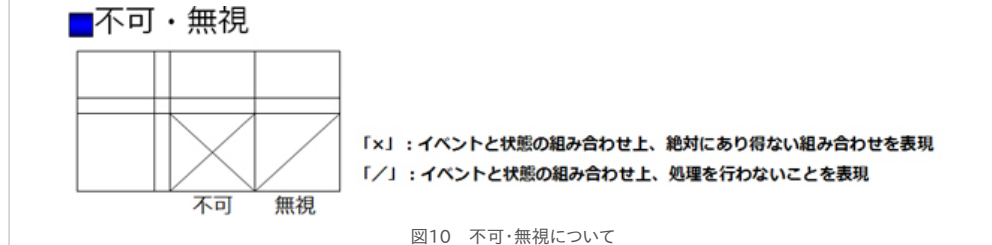
アクションとは、現在の状態にて発生したイベントに対し、実行される処理・機能のことです。

アクションは、日本語などの「自然言語」や、C言語などの「コンピュータ言語」で記述できます。他に「NSチャート」の表現を行うことも可能です。

NSチャートの表現は、C言語の「if-else」などの条件分岐構文や、「switch-case」などの条件種別構文の他に、「for」「while」「do-while」などの繰り返し構文の表現が可能です。また、それらの表現は、順次処理を順番に表現することも、選択処理と順次処理を組み合わせる表現することも可能です。つまり、状態遷移表のアクションで構造化表現をすることが可能となります。



その他、要求分析モデルで既に紹介しましたが、イベントと状態の組み合わせ上であり得ない組み合わせを表現した「×」状態と、イベントと状態の組み合わせ上、処理を行わないで無視してよい「/」状態の表現が存在します。



カスタム検索

Embedded組み込み開発

EE Times Japan × EDN Japan

統合電子版

2019年5月号 発行!

正体不明の異物はあるのか? 最新サーバの搭載チップ事情

ノートルダム大聖堂の再建に必要な技術とは

詳細はこちら

スポンサーからのお知らせ

- PR -

> 【5月29日 東京開催】MONOist 主催セミナー

失われつつある「日本品質」という強み、

「攻めの品質管理」で生まれる新たな価値

Special Contents

- PR -

コネクタ、センサーが実現する次世代モビリティ社会、「ホロレンズ」で体験

スマートファクトリー化でCC-Link IE TSNが果たすべき役割

現実を超えた仮想環境へ、自動運転時代に向けた最新モデルベース開発ツール

リアルタイムOS上でROSが動く、産業用機器へのOPC UAサーバ機能搭載も

次世代Power over Ethernet規格「PoE++」対応機器を実現するチップセット

日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ

「ねじレス化」が生み出す価値、盤製作全体の効率化を目指す制御・配電盤革新

キッチンタイマーモデルのアクション部分を機能拡張してみよう

それでは、キッチンタイマーモデルのアクション部分を機能拡張してみましょう。

時間設定状態で、「分ボタン」イベントが入力された際は、設定値を1分カウントアップします。また、「10秒ボタン」に関しては、もしも設定値が50秒の場合は、分をケタ上げする必要があるため、設定値を1分カウントアップし、秒を0秒に設定します。その他の場合は、設定値を10秒カウントアップするというように条件分岐を使用して表現します。「分ボタン」についても、計測時間の最大値が指定されている場合は、条件分岐を使用して表現する必要がありますが、今回は、最大値が指定されていないため、条件分岐を使用しないこととします。

次に、「スタートストップボタン」イベントが押された際を考えてみたいと思います。

前回作成した要求分析モデルで、初期状態と時間設定状態を1つにまとめたため、時間設定ボタンが押され、時間が設定された状態で「スタートストップボタン」が押された場合と、時間設定がされていない“0分0秒”の状態で「スタートストップボタン」が押された場合が存在します。

現在の状態遷移表では、設定値が“0分0秒”で、「スタートストップボタン」が押された場合は、既にカウントダウン時間であるため、カウントダウンの開始とともに、すぐにアラーム音出力されます。要求仕様(基本仕様)に、「時間設定完了の場合、カウントダウンを開始する」と記述してあります。それ故、0分0秒の際に「スタートストップボタン」が押された場合は、処理を行わない無視「/」としました。

- 時間設定仕様
- ・分ボタンが押されたら、設定値を1分カウントアップする
 - ・10秒ボタンが押されたら、設定値を10秒カウントアップする。また、50秒の際は、分を1分カウントアップし、00秒とする
 - ・クリアボタンが押されたら、設定値を0分0秒にクリアする

キッチンタイマー		時間設定	時間計測		
			カウントダウン	アラーム音出力	
		0	1	2	
時間設定ボタン	分ボタン	設定値を1分カウントアップ	/	/	
	10秒ボタン	設定値が50秒の場合	else		
		設定値を1分カウントアップ	設定値を10秒カウントアップ	/	/
		設定値を0秒に設定			
	クリアボタン	設定値を0分0秒にクリア	/	/	
スタートストップボタン		設定値が0分0秒の場合	else	カウントダウン停止	アラーム音停止
		/	カウントダウン開始	時間設定	時間設定
-		カウントダウン			
カウントダウン終了		×	カウントダウン停止 アラーム音出力	×	アラーム音出力

※時間設定が行われずに、0分0秒でスタートストップボタンが押された場合の処理も追加

図11 変更した状態遷移表(アクション部分)

遷移部分の機能拡張例

遷移とは、ある状態からある状態へと移行変わることを意味します。

機能拡張により、状態を親子表現することが可能となったため、親状態へ遷移した際に、子状態が固定で遷移する場合と、前回の状態を記憶しておき、前回と同じ状態へ遷移する場合が存在します。

DVDデッキを例に説明すると、前回の状態が早送りで終了していた場合、その状態が記憶されているとすると……、次にディスクを挿入したらいきなり早送りが始まってしまいます。これでは困



プラットフォームにならないマイクロソフトの「CASE戦略」

» Special 一覧

Special Site

- PR -



第4次産業革命をチャンスに
日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ



【Embedded Innovations】
マイコン/アナログ/メモリ最新情報を配信中。
組み込みの最新情報をチェック

LTC6560/LTC6561 TIA
アンプは、LIDARおよび
産業用画像処理向け

出力多重化機能付き
シングルおよび4チャンネル
トランスインピーダンスアンプ



ANALOG
DEVICES
AHEAD OF WHAT'S POSSIBLE™



メーカー公認
ディストリビュータ



詳細はこちら

コーナーリンク

Windows7サポート終了 対策ナビ

Windows 7 サポート終了 対策ナビ

Windows 10 IoT

FPGA

車載ソフトウェア

組み込み開発の記事ランキング



パナソニックがソフト開発体制強化へ
「製品を常にアップデート可能にする」



いまさら聞けないLPWAの選び方
【2019年春版】



HPCとAI性能を両立したポスト「京」の
CPU、ウエハーが初公開



人体通信で医療IoT、加速する医療機
器のモバイル化——MEDTEC Japan
2019レポート



CAN通信におけるデータ送信の仕組み
とは？



CANプロトコルを理解するための基礎知
識



ROSロボット開発者向け開発管理環境
ユーティリティを公開



質量分析計のピークピッキングをAIで自動

https://monoist.atmarkit.co.jp/mn/articles/1209/24/news004_2.html

2/5

てしまいますね。新たにディスクを入れたら、停止状態が再生状態になってほしいものです。

一方で、一時停止を例に考えてみたいと思います。早送り状態から一時停止した場合、一時停止を解除したら、早送り状態へ戻るべきです。同じく、再生状態から一時停止を行った場合、一時停止を解除したら、再生状態へ戻るべきです。このように、前回の状態を記憶しておき、前回の状態へ遷移しなくてはならない場合も存在します。

そのため、必ず固定で遷移する**固定型(Fixed)**と、全て親子の状態へ遷移する**深層記憶型(Deep-Memorized)**が存在しています(図12)。

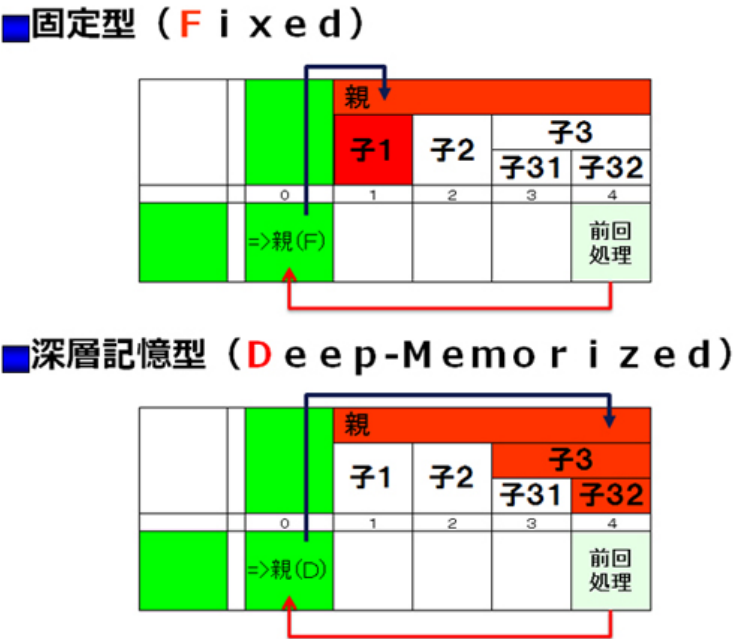


図12 固定型(Fixed)と深層記憶型(Deep-Memorized)について

キッチンタイマーモデルの遷移部分を機能拡張してみよう

それでは、キッチンタイマーモデルの遷移部分を機能拡張してみましょう。

時間計測状態へ遷移する際は、前回の状態がアラーム状態からの遷移であっても、必ず固定でカウントダウン状態へ遷移してほしいので、固定型(Fixed)で遷移させます。

記述方法は、遷移先表現の時間計測の後に、(F)を記述し、「時間計測(F)」と記述します。ちなみに、深層記憶型(Deep-Memorized)で遷移する際は、(D)と記述します。

ID キッチンタイマー		時間設定		時間計測	
				カウントダウン	アラーム音出力
時間設定ボタン	分ボタン	設定値を1分カウントアップ		/	/
	10秒ボタン	設定値が0秒の場合 設定値を1分カウントアップ 設定値を0秒に設定	else 設定値を10秒カウントアップ	/	/
	クリアボタン	設定値を0分0秒にクリア		/	/
	スタートストップボタン	設定値が0分0秒の場合 /	else カウントダウン開始	カウントダウン停止	アラーム音停止
カウントダウン終了		×		カウントダウン停止 アラーム音出力	×

図13 変更した状態遷移表(遷移部分)

関連キーワード

化、島津製作所と富士通が共同開発

ネクスティが販売パートナーになった
QNX、マイコンレベルのプロセッサもカバー

日本初のAIプロダクト品質保証ガイドライン、QA4AIコンソーシアムが発行へ

よく読まれている編集記者コラム

モノづくり魂集の
れ話
ぼ

どうして地下鉄のホームに
LiDARがあるのか、東京メトロ
に聞いてきた

自分にとっての「ぴったり」と
は？ 個人のモノづくりの価値
をもう一度問う

「LOVOT」のプロダクトデザイ
ンから学んだ“仕事の流儀”

» 編集後記一覧

人気記事ランキング - PR -

提供  オートモーティブ・ジョブズ
AUTOMOTIVE JOBS



【動画で解説】2019年度の
採用開始!今年の転職トレ
ンドは?



<高齢者事故対策>ペダ
ル踏み間違い防止や逆走
防止など様々なアプローチ
が登場



45%が「勝手にブレーキを
かけてくれる」と認識 自動
ブレーキにまつわる誤解とリ
スク



【ホンダ】HEV/PHEV用で
熱効率40%超を達成したエ
ンジン戦略



男性がつけていたら恥ずか
しい「四柄入りご当地ナン
バー」ランキング

» 他の記事を見る



MONOist > 組み込み開発 > 状態遷移表による設計手法(4):状態遷移表を使用...

状態遷移表による設計手法(4):

状態遷移表を使用した設計モデル(拡張階層化状態遷移表) (3/3)

2012年09月24日 10時00分 公開

【塚田 雄一 キャッツ, MONOist】

前のページへ [1](#) [2](#) [3](#)

印刷

通知

2

Share

21

アクティビティとは

アクティビティとは、イベントを検出した際や状態が遷移した際など、特定の状況下で呼び出される機能、または処理を表現したものです。

種類は、「状態アクティビティ」「イベントアクティビティ」「トリガアクティビティ」「STMメインアクティビティ」など、さまざまなアクティビティが存在しています。その中から今回は、状態アクティビティを紹介します。

状態アクティビティには、「状態スタートアクティビティ」「状態モードアクティビティ」「状態エンドアクティビティ」「ディスパッチアクティビティ」などが存在します。ここでは、状態スタートアクティビティと状態エンドアクティビティについて紹介します。

状態スタートアクティビティの例

状態スタートアクティビティは、状態遷移が行われた際に、最初に行う処理や機能を表現したものです。

図14のように、状態に遷移した際、最初に行うべき処理は、状態スタートアクティビティによりまとめて表現できます。

カスタム検索

Embedded

組み込み開発

MONOist

MONOist 主催セミナー

製造業×品質、
転換期を迎える
モノづくりの在り方

開催日: 2019/5/29 (水)

会場: 野村コンファレンスプラザ日本橋

無料

スポンサーからのお知らせ

> 【5月29日 東京開催】MONOist 主催セミナー
失われつつある「日本品質」という強み、
「攻めの品質管理」で生まれる新たな価値

Special Contents

Microsoft

プラットフォームにならないマイクロソフトの「CASE戦略」

自動運転時代の自動車開発の必需品! 最新モデルベース開発ツールを一挙紹介

CC-Link IE TSN

スマートファクトリー化でCC-Link IE TSNが果たすべき役割

コネクター、センサーが実現する次世代モビリティ社会、「ホロレンズ」で体験

次世代Power over Ethernet規格「PoE++」対応機器を実現するチップセット

日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ

「ねじレス化」が生み出す価値、盤製作全体の効率化を目指す制御・配電盤革新

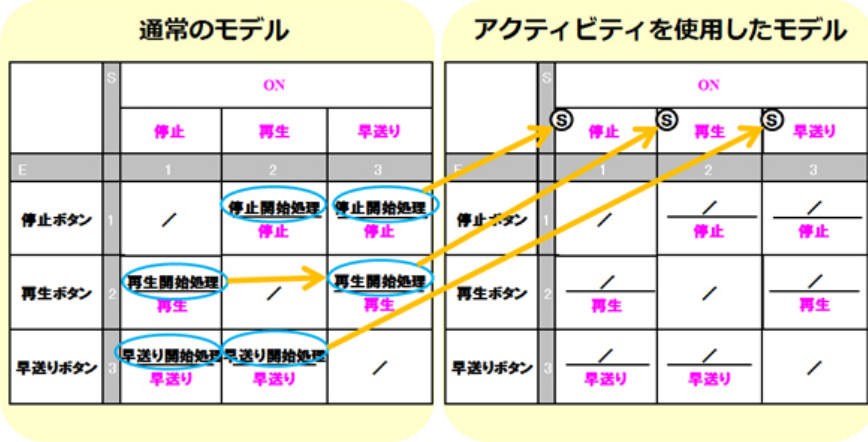


図14 状態スタートアクティビティ

状態エンドアクティビティの例

状態エンドアクティビティは、状態遷移が行われた際に、最後に実行する処理や機能を表現したものです。

図15のように、状態に遷移した際、最後に実行すべき処理は、状態エンドアクティビティによりまとめて表現できます。

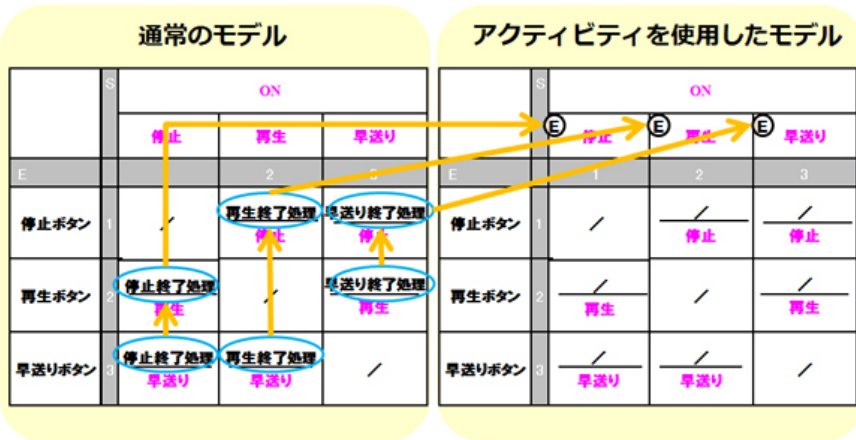


図15 状態エンドアクティビティ

アクティビティの効果

図16のように、通常のモデルを、状態アクティビティを使用して表現することにより、すっきりとしたモデルに変更できます。

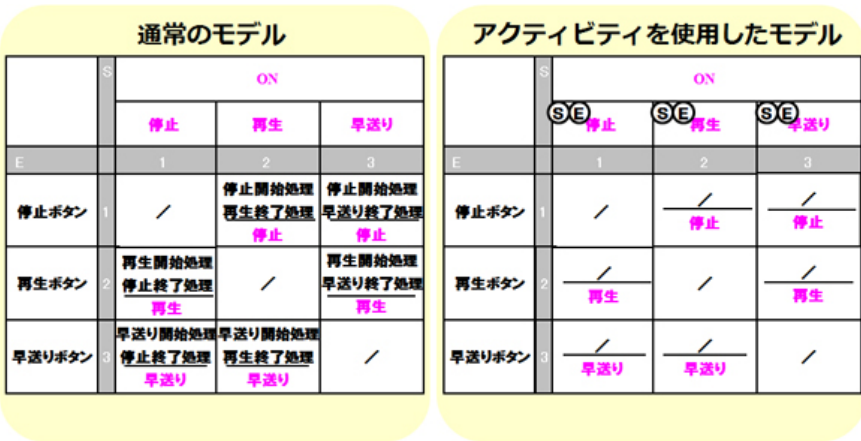


図16 アクティビティの効果



リアルタイムOS上でROSが動く、産業用機器へのOPC UAサーバ機能搭載も

» Special 一覧

Special Site

- PR -



第4次産業革命をチャンスに
日本の製造業が直面する課題とその解決、マイクロソフトが描く変革のシナリオ



【Embedded Innovations】
マイコン/アナログ/メモリ最新情報を配信中。
組み込みの最新情報をチェック

LTC6560/LTC6561 TIA
アンプは、LIDARおよび
産業用画像処理向け

出力多重化機能付き
シングルおよび4チャンネル
トランスインピーダンスアンプ


ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™
 メーカー公認
ディストリビュータ
詳細はこちら

コーナーリンク

Windows7サポート終了 対策ナビ

Windows 7 サポート終了 対策ナビ

Windows 10 IoT

FPGA

車載ソフトウェア

組み込み開発の記事ランキング

パナソニックがソフト開発体制強化へ
「製品を常にアップデート可能にする」

いまさら聞けないLPWAの選び方
【2019年春版】

HPCとAI性能を両立したポスト「京」の
CPU、ウエハーが初公開

人体通信で医療IoT、加速する医療機器の
モバイル化——MEDTEC Japan 2019レポート

CAN通信におけるデータ送信の仕組み
とは？

CANプロトコルを理解するための基礎知識

ROSロボット開発者向け開発管理環境
ユーティリティを公開

質量分析計のピークピッキングをAIで自動

キッチンタイマーモデルをアクティビティで機能拡張してみよう

それでは、キッチンタイマーのモデルをアクティビティで機能拡張してみましょう。

実は、今回の状態遷移表はカウントダウン状態で、「カウントダウン停止」処理が2つ存在していますが、これらを状態アクティビティでまとめて表現すると有効なモデルとなります。ここでは、あくまで状態アクティビティで表現した際の例として紹介します。

図17のモデルで、カウントダウン状態へ遷移する際、「カウントダウン開始」処理を行っています。また、アラーム音出力状態へ遷移する際、「アラーム音出力」処理を行っています。これらを状態スタートアクティビティへ変更します。また、カウントダウン状態から遷移する際、「カウントダウン停止」処理を行っています。そして、アラーム音出力状態から遷移する際、「アラーム音停止」処理を行っています。これらの処理は、状態エンドアクティビティに変更します。変更後の状態遷移表を図18に示します。

キッチンタイマー		時間設定	時間計測
		0	1
時間設定ボタン	分ボタン	設定値を1分カウントアップ	カウントダウン
	10秒ボタン	設定値が50秒の場合 設定値を1分カウントアップ 設定値を0秒に設定	アラーム音出力
	クリアボタン	設定値を0分0秒にクリア	
スタートストップボタン		設定値が0分0秒の場合 カウントダウン開始	カウントダウン停止
カウントダウン終了			アラーム音出力

状態スタートアクティビティへ変更

状態エンドアクティビティへ変更

図17 変更前

キッチンタイマー		時間設定	時間計測
		0	1
時間設定ボタン	分ボタン	設定値を1分カウントアップ	／
	10秒ボタン	設定値が50秒の場合 設定値を1分カウントアップ 設定値を0秒に設定	／
	クリアボタン	設定値を0分0秒にクリア	／
スタートストップボタン		設定値が0分0秒の場合 時間計測	／
カウントダウン終了		／	×

図18 変更した状態遷移表(アクティビティ)

階層化とは

階層化とは、1つの状態遷移表を複数の状態遷移表で分割表現したものです。階層化を行うことにより、状態遷移表の巨大化を抑制できます。

例えば、図19の状態遷移表は、状態が9個、イベントが12個、アクション(処理セル数)が108個も存在しています。この状態遷移表を階層化表現することにより、アクション(処理セル数)を約半分の50個まで減らすことができます。

- 化、島津製作所と富士通が共同開発
- ネクスティが販売パートナーになった
QNX、マイコンレベルのプロセッサもカバー
- 日本初のAIプロダクト品質保証ガイドライン、QA4AIコンソーシアムが発行へ

よく読まれている編集記者コラム

- モノづくり魂集の どうして地下鉄のホームにLiDARがあるのか、東京メトロに聞いてきた
- 自分にとっての「ぴったり」とは？ 個人のモノづくりの価値をもう一度問う
- 「LOVOT」のプロダクトデザインから学んだ“仕事の流儀”

» 編集後記一覧

人気記事ランキング

- PR -

- 提供 オートモーティブ・ジョブズ AUTOMOTIVE JOBS
- 【動画で解説】2019年度の採用開始!今年の転職トレンドは?
- <高齢者事故対策>ペダル踏み間違い防止や逆走防止など様々なアプローチが登場
- 45%が「勝手にブレーキをかけてくれる」と認識 自動ブレーキにまつわる誤解とリスク
- 【ホンダ】HEV/PHEV用で熱効率40%超を達成したエンジン戦略
- 男性がつけていたら恥ずかしい「図柄入りご当地ナンバー」ランキング

» 他の記事を見る

■ 階層化とは



図19 階層化とは

階層化の種類は、イベント(事象)に着目して階層化を行う「**事象階層化**」と、状態に着目して階層化を行う「**状態階層化**」が存在します。

キッチンタイマーモデルを状態階層化してみよう

それでは、キッチンタイマーモデルを状態階層化してみましょう。

状態階層化の候補は、状態を親子関係で表現している部分です。実は、今回の状態遷移表は、1つの状態で表現した方が分かりやすいモデルとなります。ここではあくまで、状態階層化を行った際の例として紹介します。

階層化の対象となる部分は、時間計測状態の子状態で「カウントダウン」「アラーム音出力」が存在している箇所です(図20)。

※注: 下記は、アクティビティに変更する前の状態遷移表をベースに、状態階層化を行った際の例です。



図20 変更前

階層化を行うと、図21のようになります。

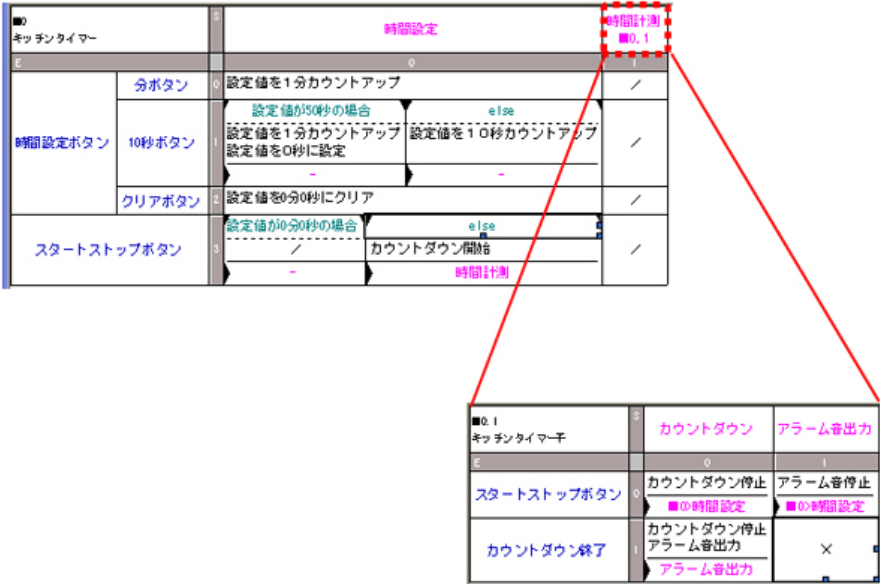


図21 変更した状態遷移表(階層化)

今回のまとめ

今回は“状態遷移表を使用した設計モデル”をテーマに、[前回](#)作成した要求分析モデルを基に、拡張階層化状態遷移表設計手法を使用して、設計モデルを作成しました。

拡張階層化状態遷移表は、組み込みソフトウェアの開発を行う際に必要な表現を追加し、巨大化する状態遷移表を階層化により解決したものです。今回はその一部を紹介しました。

□

さて、次回は「[状態遷移表からの実装](#)」をテーマに、今回作成した設計モデルを基にして、ソースコード(実装言語)を作成する手法を紹介します。お楽しみに！(次回に続く)

組み込みモデリング コーナー

>>コーナーTOPはこちらから

「状態遷移表による設計手法」バックナンバー
状態遷移表を使用したテスト手法【後編】
状態遷移表を使用したテスト手法【前編】
状態遷移表からの実装
状態遷移表を使用した設計モデル(拡張階層化状態遷移表)
状態遷移表を使用した要求分析モデル
なぜ状態遷移表を使うと、品質の良い開発ができるのか
状態遷移表設計手法の概要

この連載を「連載記事アラート」に登録する

関連キーワード

ソフトウェア | モデルベース開発 | 設計 | 組み込み | 組み込みソフトウェア | UML | 組み込みシステム | モデリング