

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(5):状態遷移表からの...

状態遷移表による設計手法(5):

状態遷移表からの実装(3/3)

2012年11月07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

3

Share

7

ステートドリブン型(S型)

ステートドリブン型は、先に状態を解析するため、プログラムの1箇所ですべての状態を解析します。そして、状態を解析した後に、状態ごとに必要なイベントのみを解析します。例えば、アクションセルが“無視”や“不可”となるイベントのアクションについては解析を行いません。

主に、メニュー画面などのある画像処理分野では一般的な構造であり、多く使用されています。例えば、メニューA表示状態で、イベントXが入力された場合と、メニューB表示状態でイベントXが入力された場合などは、表示しているメニュー画面(状態)が確定していないと、イベントの意味が明確となりません。従って、最初に現在表示されているメニュー画面などの状態を解析した後に、メニューに応じた入力イベントを解析することが多く、そのようなシステムにはステートドリブン型の実装が適しています。

以下に、ステートドリブン型の特徴を記します。

- 常に全ての状態を解析する
- 必要なイベントのみを解析する
- アクションセルが“無視”や“不可”となるアクションセルのイベントの解析は行わない

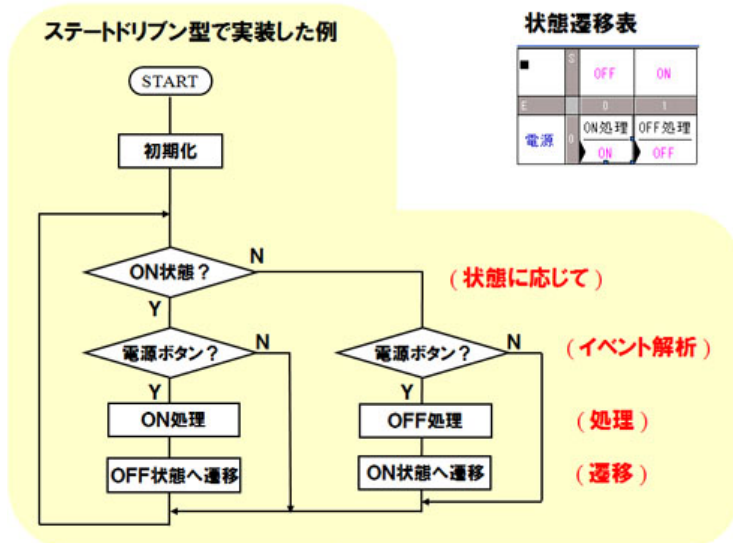


図10 ステートドリブン型(S型)で実装した例

カスタム検索



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

- 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 -



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



コネクター、センサーが実現する次世代モビリティ社会、「ホロレンス」で体験



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



スマートファクトリー化でCC-Link IE TSNが果たすべき役割



日本の製造業が第4次産業革命の波に乗るために必要な「3つのIoT活用」とは



デンソーが次世代車載コックピットへの採用を決めたフラッシュメモリとは? **New!**



IoTのPoC段階はもう終わり、工場のスマート化が本格的に加速へ

キッチンタイマーをステートドリブン型(S型)で実装

それでは、状態遷移表からステートドリブン型で実装したフローチャートを作成してみましょう。

まずメインループで、「時間設定状態」「カウントダウン状態」「アラーム音出力状態」の3つの状態を全て解析します。この時、有効となる状態は1つのみです。つまり、メインループで状態の切り替えを行い、状態が遷移した際に、有効となるループを切り替える構造になります。そして、以下のよう、状態ごとに必要なイベントを解析し処理を行います。

(1)「時間設定状態」のループにおいて、「時間設定ボタン」イベントがヒットした際には「カウントダウン時間設定」処理を行い、「スタートストップボタン」イベントがヒットした際には「カウントダウン開始」処理を行い、「カウントダウン状態」へ遷移します。

(2)「カウントダウン状態」のループにおいて、「スタートストップボタン」イベントがヒットした際には「カウントダウン停止」処理を行い、「時間設定状態」へ遷移します。「カウントダウン終了」イベントがヒットした際には「カウントダウン停止&アラーム音出力」処理を行い、「アラーム音出力状態」へ遷移します。

(3)「アラーム音出力状態」のループにおいて、「スタートストップボタン」イベントがヒットした際には「アラーム音停止」処理を行い、「時間設定状態」へ遷移します。

状態については、イベントドリブン型と同様に、状態変数を用意しておき、遷移する際は状態番号(時間設定状態(0)、カウントダウン状態(1)、アラーム出力状態(2))を設定します。そして、現在の状態を参照する際は、状態変数の内容を参照します。つまり、複数のマイナーループで状態を表現するのではなく、1つのメインループで表現し、状態を状態変数で切り替える構造になります。

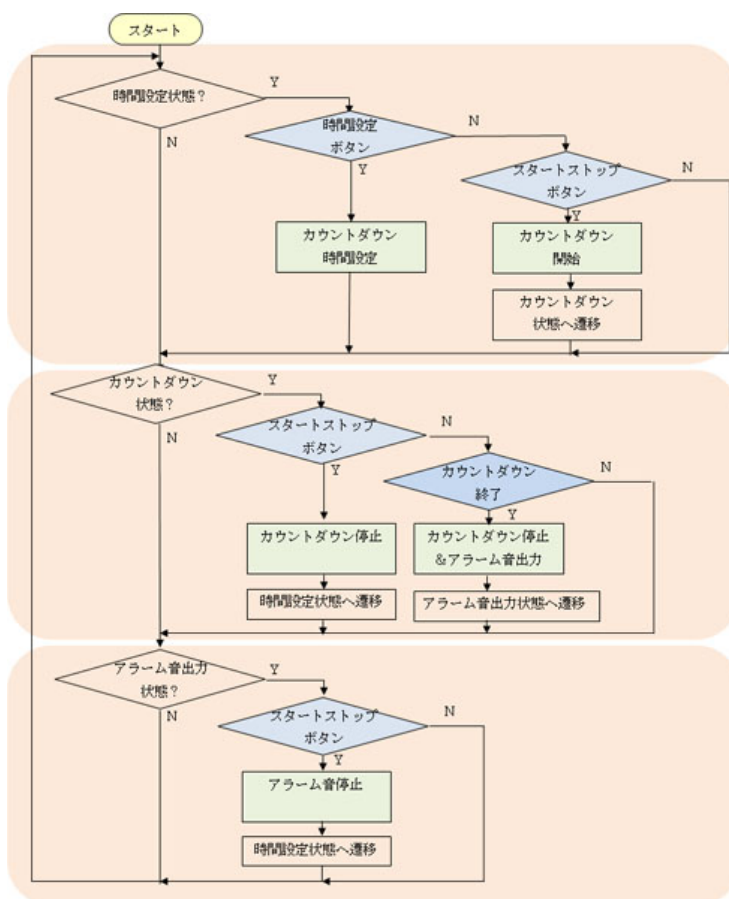


図11 キッチンタイマーをステートドリブン型(S型)で実装



IoT時代の安全と安心を確保する
ワンストップの組み込みプラットフォームとは **New!**



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

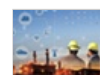


自動車業界の大変革CASEに向けて、マイクロソフトが持つ“部品” **New!**

» 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™

Digi-Key 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

深層学習初心者向けの無料オンライン
学習資料を公開

いまさら聞けないLPWAの選び方
【2019年春版】

低速自動運転を“商品レベル”で実現
へ、ヤマハ発動機とDMPがAIで資本提携

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

CANプロトコルを理解するための基礎
知識

状態遷移表からステートドリブン型 (S型) で実装した場合と、基本仕様から直接実装した場合の比較

それでは、状態遷移表を作成し、その後、ステートドリブン型で実装した場合と、基本仕様から直接実装した場合を比較してみましょう。

状態遷移表からステートドリブン型で実装した場合の特徴は、以下の通りです。

- “マイナーループ”が存在せず、1つのメインループ構造になっている
- 全ての状態を1箇所で参照している
- 状態変数の切り替えにより状態遷移を行っているため、順序性のない状態遷移にも適している

状態遷移表からステートドリブン型で実装した場合、状態は全て1箇所で管理されるため、状態の解析モレはなくなり、品質が向上します。また、状態を追加するなどの変更の際も、1つのメインループに状態を追加すればよいため、メンテナンス性も向上します。そして、状態遷移表から実装したフローチャートは、もともと状態を意識した構造となっており、下部から上部へ移動するなど順序性のない状態遷移部分についても容易に実装することが可能です。

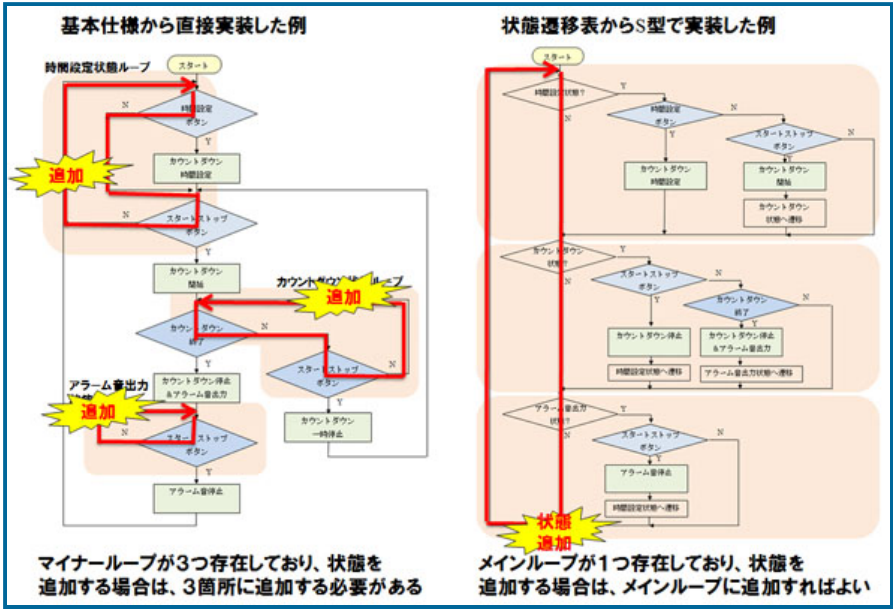


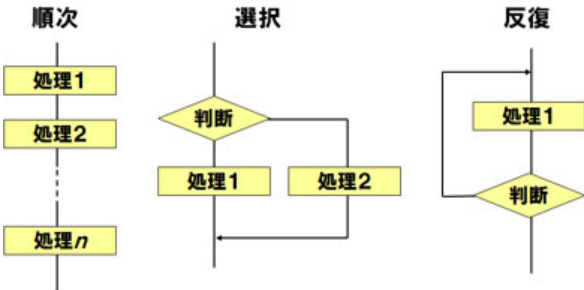
図12 状態遷移表からステートドリブン型 (S型) で実装した場合と基本仕様から直接実装した場合の比較

コラム: 構造化プログラミングとは

構造化プログラミングは、階層的に抽象化されたプログラムの組み合わせとして、プログラムを記述する手法であり、「順次」「選択」「反復」の3種類の制御構造のみを用いてプログラミングを行います。

構造化プログラミングの目的は、品質の良いプログラミング構造を実現することであり、無条件に、指定されたラベルや行番号にジャンプ (跳ぶ) するGOTO文を使用しません。GOTO文を使用すると、動作するシーケンスが複雑になるため、動作品質の保証が困難となります。

基本仕様から直接実装した場合のフローチャートにおいて、下部のループから上部のループにジャンプ (跳ぶ) するGOTO文を使用すれば、容易に実装できますが、上記の理由から、構造化プログラミングではGOTO文を使用せずに、フラグや変数でループを切り替えます。



<input type="checkbox"/>	HPCとAI性能を両立したポスト「京」のCPU、ウエハーが初公開
<input type="checkbox"/>	CAN通信におけるデータ送信の仕組みとは？
<input type="checkbox"/>	ポスト「京」のプロセッサ「A64FX」はArmベースながら異彩放つ重厚系
<input type="checkbox"/>	スマートグラス活用ソリューションを保守点検業務に採用
<input type="checkbox"/>	PythonだけでIoTのPoCを組める／グラフエンで世界最高感度の赤外線センサー

よく読まれている編集記者コラム

- 


「LOVOT」のプロダクトデザインから学んだ“仕事の流儀”
- 


製造業も人ごとではない、もう1つの2025年問題
- 

期待高まる金属3Dプリンタの本格導入、意識の壁を超えられるか
- » 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




【ホンダ】HEV/PHEV用で熱効率40%超を達成したエンジン戦略



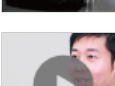
男性がつけていたら恥ずかしい「図柄入りご当地ナンバー」ランキング



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



【日系自動車メーカー・ADAS比較】自動運転・ADAS技術開発の最新動向



【動画で解説】研究開発の採用動向 CASE関連の中途採用が進む

» 他の記事を見る

図13 構造化プログラミングとは

今回のまとめ

以下に、「基本仕様から直接実装」「状態遷移表からイベントドリブン型で実装」「状態遷移表からステートドリブン型で実装」を比較した表を記します(表1)。

比較項目	基本仕様から直接実装	状態遷移表からE型実装	状態遷移表からS型実装
ループ構造	マイナーループ3つを実行順序で管理	メインループ1つで管理	状態ごとのループをメインループで管理
状態遷移構造	上位ループから下位ループへ切り替え	状態変数で切り替え	状態変数で切り替え
イベント参照部分数	5箇所	3箇所	5箇所
イベント変更/追加	検討する場所がさまざまで困難	1箇所検討すればよい	3箇所検討する必要がある
状態変更/追加	検討する場所がさまざまで困難	3箇所検討する必要がある	1箇所検討すればよい
順序性のない状態への遷移	あまり向いていない	可能 ※容易にできる	可能 ※容易にできる

表1 「基本仕様から直接実装」「状態遷移表からイベントドリブン型で実装」「状態遷移表からステートドリブン型で実装」を比較

「基本仕様から直接実装」を行った場合は、イベント参照、状態参照など同じ処理があちこちに点在しています。それに比べ、状態遷移表は、イベントと状態を整理して考えられ、その後実装を行えば、イベントや状態などの処理をまとめて実装することが可能となります。また、状態遷移表から実装を行う手法として、イベントドリブン型とステートドリブン型が存在します。システムの性質上イベントによって処理を変えることが多く、イベント主体の処理であれば、イベントドリブン型で実装を行い、状態によって処理を変えることが多く、状態主体の処理であれば、ステートドリブン型で実装を行う方が適しています。

□

さて、今回は“状態遷移表からの実装”について解説しました。状態遷移表を用いると「モレ」「ヌケ」のない設計が行えるため、要求分析のフェーズで、検討すべき仕様が明確となります。また、状態とイベントの組み合わせが整理された状態遷移表(設計書)を作成し、それを基に、イベントドリブン型あるいはステートドリブン型で実装を行うことにより、変更が容易なソフトウェアを実現できるため、メンテナンス性が向上します。今回の内容から、状態遷移表設計手法が、設計レベルだけではなく、実装レベルにおいても、品質を向上させるために有効な設計手法であることが理解できたのではないのでしょうか。

さて、次回は「状態遷移表を使用したテスト手法」をテーマに、状態遷移表から試験項目、試験シナリオを作成する手法を紹介したいと思います。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

関連記事

連載記事「状態遷移表による設計手法」

>>「組み込みモデリング」コーナー



「品質100%保証」を実現するCASEツールへ

「組み込みマルチコア進化論」最新記事一覧

前のページへ 1 2 3

組み込み開発

MONOist トップ

Copyright © ITmedia, Inc. All Rights Reserved.

Special Contents

- PR -



日本の製造業が第4次産業革命の波に乗るために必要な“3つのIoT活用”とは



IoTのPoC段階はもう終わり、工場のスマート化が本格的に加速へ



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



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



プラットフォームにならないマイクロソフトの「CASE戦略」**New!**



スマートファクトリー化でCC-Link IE TSNが果たすべき役割



IoT時代の安全と安心を確保するワンストップの組み込みプラットフォームとは **New!**



「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

