

## マイコン組み込みS/Wの開発を支援するCASEツール：testCASE(3)

6 J-9

～testCASEを用いた開発の流れ～

別所 雄三\*、中島 毅\*\*、山中 弘\*、萩原 正敏\*、広田 和洋\*\*\*

三菱電機(株)情報技術総合研究所\*、三菱電機(株)鎌倉製作所\*\*、三菱電機セミコンダクタシステム(株)\*\*\*

## 1. はじめに

マイコン組み込み S/W(以下マイコン S/W)の開発を支援する CASE ツール testCASEは、要求分析ツールおよびテスト項目生成ツール<sup>[1]</sup>、評価テスト実行ツール<sup>[2]</sup>を備える。要求分析ツールは状態遷移表を用いて仕様を記述し、仕様を明確にする。テスト項目生成ツールと評価テスト実行ツールは、テストケースの生成と評価テストの実行を自動化する。

我々は、さらに要求分析段階または評価テスト段階における個々の段階の開発支援だけでなく、testCASEの各ツールを統合し、上流から下流までの段階を一貫して支援する統合環境の整備を進めている。本環境では、開発の各段階で定義したデータの一元的な管理と、開発プロセスに基づいたツールの統合を実現する。

本稿では、統合環境のねらいと統合環境を用いた開発の流れについて説明する。

## 2. 統合環境

testCASEの統合環境は、データの一元的な管理とツール統合により、上流から下流までの段階をシームレスに支援し、データの整合性を保証する。また、再試験の効率と操作性の向上を実現する。

本環境は、要求分析ツールで生成する状態遷移表データや、テスト項目生成ツールが生成する試験仕様データなどの複数の情報をデータベースとして一元的に管理する(図1)。一元的に管理することにより、状態遷移表データを変更した場合、即座に評価テスト実行ツールの入力となるテストケースに反映させることが可能となる。従って、度重なる変更が発生するような場合においても、個々の開発段階におけるデータの不整合を防ぐことができる。また、データベース中の試験仕様データを利用することにより、試験仕様書の作成も支援する。

さらに、統合環境では、testCASEの各ツールを呼び出すためのGUI(Graphical User Interface)を提供することにより、作業の流れに従ったツールの起動をガイドする。

## 3. testCASEを用いた開発プロセス

これまでの開発プロセスでは、ツールの統合化が実現されていなかったため、各ツールをどのような手順で使用するのか、開発者にとってわかりにくいところがあった。しかし、統合環境において各ツール起動用のGUIを提供することにより、開発プロセスに基づいた支援が可能となり、操作性を向上することができる。

testCASEの統合環境を用いたマイコン S/Wの開発手順について次に説明する(図1)。

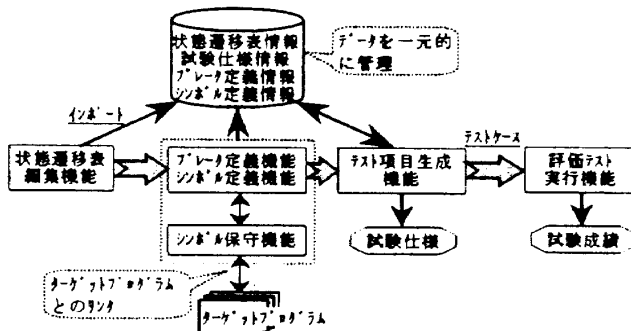


図1. 統合環境を利用した開発の流れ

## (1) 状態遷移表の作成

要求分析ツールを用いて状態遷移表を記述する<sup>[1]</sup>。

この開発ステップでは、要求分析ツールが提供する状態遷移表の作成と状態遷移表上でのシミュレーション機能によって、要求仕様を早期に明確化し、仕様の不備を排除することができる。

## (2) ターゲットプログラムとの関連付け

状態遷移表データをデータベースにインポートし、入力イベントと出力アクション情報をターゲットプログラムに関連付ける。ターゲットプログラムとの関連付けは、ターゲットプログラムで定義しているシンボルやラベルを使い、ブレーク情報を定義することにより行う。シンボルやラベルはシンボル保守機能により、ターゲットプログラムからシンボル情報を抽出し、データベースにインポートする。

ブレーク情報とは、評価テスト実行ツールにおいて実現した機能である、ICE(In-Circuit Emulator)制御によるメモリ値設定(入力ブレーク)とメモリ値

testCASE : CASE Tools for Microcomputer Software(3) - How to Use testCASE in Practice -

\*Yuzo Bessho, \*\*Tsuyoshi Nakajima, Hiroshi Yamanaka, \*Masatoshi Hagiwara, \*\*\*Kazumi Hirota

\*Information Technology R&D Center Mitsubishi Electric Corporation, \*\*Kamakura Works Mitsubishi Electric Corporation

\*\*\*Mitsubishi Electric Semiconductor System Corporation

参照(照合ブレイク)のための情報である。例えば図3のような、入力装置としてキーマトリクス、出力装置としてLEDがあるようなターゲットシステムを想定する。この場合、入力としてキーが押されたとき、LEDに対応する値が表示されることを確認するには、入力ブレイクおよび照合ブレイク情報を次のように定義する。

【入力ブレイク】

イベント: キーを押した  
 ブレイクポイント: キー入力判定処理通過直後  
 ブレイク時の処理: キー値格納メモリに試験データとなるキー値を設定

【照合ブレイク】

アクション: LEDの表示がそのキー値となる  
 ブレイクポイント: LED表示値が格納された直後  
 ブレイク時の処理: LED表示値の格納メモリと期待値を照合

この開発ステップでは、統合環境によってブレイク定義機能を提供することにより、要求仕様とターゲットプログラムとの関連付けを行うことができる。

統合環境では、状態遷移表データのインポートやブレイク情報の入力を支援するGUI(図2)を用意し、入力の手間を軽減する。

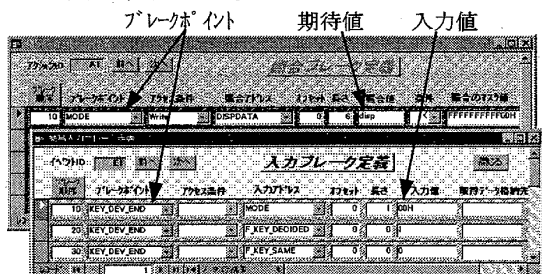


図2. ブレイク情報定義画面

(3) 要求仕様に基づくテストケースの生成

テスト項目生成ツールを用いて状態遷移表データに基づいたテスト手順を生成する<sup>[1]</sup>。さらに、テスト手順に記述された入力イベントと出力アクション情報を、入力および照合ブレイク情報と関連付けてテストケースを生成する。

この開発ステップでは、テスト項目生成ツールによって、テスト手順の生成とブレイク情報との関連付け機能を提供することにより、要求仕様に基づいたテストケースを生成することができる。

(4) 評価テストの実施

評価テスト実行ツールを用いて実機テストを行う。評価テスト実行ツールでは、ターゲットプログラムの入力後、テストケースに基づいてICEを制御することにより、評価テストを実行する。

この開発ステップでは、評価テスト実行ツールの

提供する評価テストの自動実行機能により、評価テスト段階の作業効率を向上することができる。

次に、評価テスト実行ツールの実現方式について簡単に説明する。

評価テスト実行ツールはテストの自動化を実現するためのツールである。このツールでは、マイコンS/Wを図3のように、入出力デバイスドライバとプログラム仕様部に分けて考える。このような構造では、入出力ポートの信号ではなく、その直前のメモリに対して擬似入力の設定や実行結果を取得し、期待値と照合することにより評価テストを行うことができる。また、この方式により、入力信号用の試験データを作成するのと比べ、試験データの作成も容易となる。

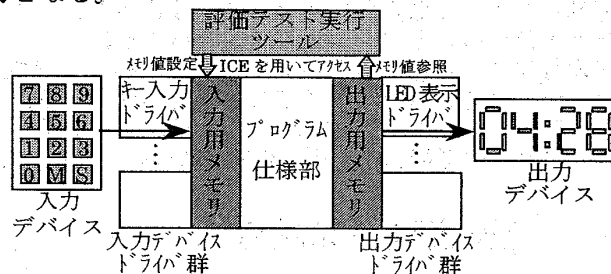


図3. マイコンS/Wの一般的な構造

4. おわりに

マイコンS/Wの開発効率を向上するCASEツールtestCASEの各ツールを統合する統合環境のねらいと統合環境を用いた開発の流れについて説明した。

統合環境では、データを一元的に管理することにより、要求仕様の変更が頻繁に発生するような場合においても、上流から下流までの段階でデータの整合性を保証することが可能になった。さらに、各ツールとの連携により操作性の向上を実現した。

我々が構築したtestCASEでは、仕様変更時に変更部分のみの再設定だけで再試験を行うことができる。従って、仕様変更から再試験までの作業を軽減でき、再試験において容易に試験項目の作成と評価テストが行える。これまでの試験運用により、再試験の効率を高め、製品の品質向上への大きな効果があることがわかった。

今後は、実プロジェクトへの適用を通じて、評価データを分析していく。

参考文献

[1]山中 弘 他:マイコン組み込みS/Wの開発を支援するCASEツール: testCASE(2)～状態遷移表とテスト手順生成～, 情報処理学会第57回全国大会, 6J-08, 1998.  
 [2]別所 雄三 他:マイコン用CASEツール: testCASE(4)～評価テスト実行ツール～, 情報処理学会第53回全国大会, 7D-4, 1996.  
 [3]中島 毅 他:シグナルチップマイコン用S/W開発における問題点と一解決法, 情処研報, Vol197, No. 74, pp. 17-24, 1997.