

マイコン用CASEツール：testCASE(4)

7D-4

～評価テスト実行ツール～

別所 雄三†、中島 毅†、石川 洋†、小林 剛†、広田 和洋‡

三菱電機(株)情報技術総合研究所†、三菱電機セミコンダクタソフトウェア株式会社‡

1. はじめに

我々は、テストの自動化技術により、評価テスト工程の手間と時間の削減を目指した、評価テスト実行ツールの開発を進めている。

本ツールは、マイコン用CASEツール testCASEの最下流に位置付けられ、上流のCASEツールであるテスト項目生成ツールによって、生成されたテストケースをもとに、ソフトウェア開発の最終工程である評価テストを行う。

本ツールにおける評価テストは、モジュール試験ではなく、要求仕様レベルの機能テストであるブラックボックステストが対象である。

本稿では、評価テスト実行ツールの概要を中心に説明する。

2. 現状の評価テスト支援ツールの問題点

現在、評価テスト工程を支援するシステムとして、いくつかのツールが市販されている。それらのツールの実現方式と問題点について述べる。

(1) 入出力ポートの制御によるテスト支援ツール

マイコンの入出力ポートとターゲットシステムとの間に、テスト支援システムを挿入し、入出力ポートの信号を操作してテストを実行する。

入力データは手動で入力し、そのオペレーションを記憶することにより、2回目以降のテストの自動実行や繰り返し実行を可能とする。

このシステムは、マイコンの入出力ポートへの接続が容易ではない。また、評価テストの実行に、特別なハードウェアが必要であるという問題点がある。

(2) ソフトウェアシミュレーション

マイコンや周辺デバイスなどターゲットシステムをすべてソフトウェアでモデル化することにより、ソフトウェアのシミュレーションに

よって評価テストを実行する。

このシステムでは、周辺デバイスのモデル化やビジュアル表示の設定に手間を要するため、小規模の開発には不向きである。また、全てがシミュレーションモデルによるテストであるため、最終的には実機によるテストが必要であるという問題点がある。

さらに、両システムともテストケース生成機能を有していないため、人手によって入力しなければならない。

我々は、このような現状のツールの問題点を踏まえ、要求分析工程と評価テスト工程の統合化を testCASEにより実現する。testCASEを構成する各ツールによって、要求分析、テスト項目生成、評価試験の各工程を連携し、各工程における人手による手間を削減することにより、ソフトウェアの開発効率向上を図る。

3. 評価テスト実行ツール

3.1 構成

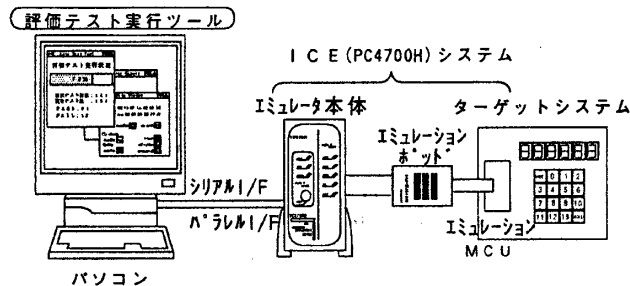


図1. 評価テスト実行ツールのシステム構成

本ツールのシステムは、図1に示すように、パソコン、ICEシステム、ターゲットシステムから構成される。

このように、本ツールのシステム構成は、現状

testCASE : CASE Tools for Microcomputer Software(4) - Evaluation Test Tool -

† Yuzo BESSHO, † Tsuyoshi NAKAJIMA, † Hiroshi ISHIKAWA, † Tsuyoshi KOBAYASHI

‡ Kazumi HIROTA

† Mitsubishi Electric Corporation

‡ Mitsubishi Electric Semiconductor Software Corporation

のマイコン開発に使用されている既存のシステムによって、容易に構築することが可能である。

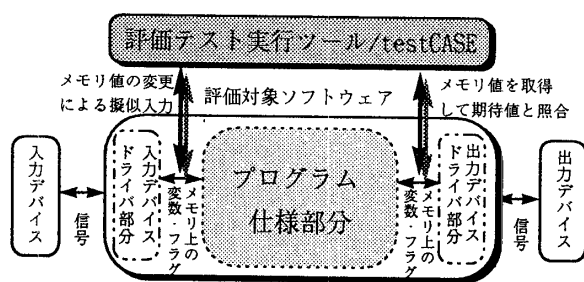


図2. 評価テスト実行ツールの実現方式

本ツールの実現方式を図2に示す。

本ツールでは、評価対象ソフトウェアを入出力デバイスとのデバイスドライバ部分と、ソフトウェアの本質であるプログラム仕様(本体)部分に分けてとらえている。このように分けることにより、デバイスドライバ部分とプログラム仕様部分のインタフェースとなる、メモリ値を操作し、評価テストを実行する。

本方式により、入出力ポートの信号ではなく、その直前のメモリ値を操作し、擬似入力を与えたり、実行結果を取得することにより、特別なハードウェアを必要とせず、評価テストが実行できる。

デバイスドライバとは、入出力ポートに接続されている入出力デバイスを制御するためのソフトウェアである。

3.2 機能

本ツールの主な機能を以下に述べる。

(1) テスト項目選択機能

テスト項目生成ツールによって、生成されたテストケースを入力し、テストケース中の複数のテスト項目から、ユーザが任意のテスト項目を選択することを可能にする。

(2) テスト実行機能

入力したテストケースを解析し、テストケース中の各テスト項目にしたがって、テストを実行する。

(3) データ入力機能

ICEのメモリ変更機能を利用して、マイコン内に組み込まれるターゲットプログラムの変数やフラグに、テストケースに記述されている入力データを書き込み、マイコンに対して擬似的な入力を与える。

(4) 各種実行モードの設定機能

エラーが発生した時点で以後のテストを行わずに終了する。または、エラーが発生しても最後までテストを行うといった実行モードが、ユーザから選択できることを可能にする。

(5) データ照合機能

ICEのメモリ参照機能を利用して、ターゲットプログラムの変数やフラグから出力データを取り出す。また、テストケースに記述されている期待値と取得データを照合し、テスト合否を判定する。

(6) テスト実行状況表示機能

テストの進み具合、現在実行中のテスト項目、エラー情報などを実行中のテスト状況として、随時表示する。

(7) カバレッジ計測機能

ICEのカバレッジ機能(C0カバレッジ)を利用して、プログラムが実行(アクセス)したアドレスをプログラムのソースレベルで表示する。

(8) 各種レポート機能

評価テストの実行結果を表示するだけでなく、不具合情報や、テスト実行結果(照合結果)などをまとめた、試験成績レポートを作成する。

さらに、試験成績データは、客先向けの試験成績書として、自由に利用できるよう、ファイルとしても出力できる。

4. 効果

本ツールにより、以下の効果が期待できる。

- テストの自動化により、人手による手間と時間を削減し、ソフトウェア開発期間の短縮と品質向上を実現する。
- 繰り返しテストを行うことが可能であるため、不具合修正後の再テスト実行が容易になる。
- カバレッジ機能により、テスト実行後、未実行のプログラム領域が把握でき、テストデータが十分であるか確認することが可能になる。
- ドキュメント(検査成績書)作成の支援機能により、テスト実行だけでなく、レポート作成の手間についても削減することができる。

5. おわりに

本稿では、我々が現在開発を進めている、評価テスト実行ツールについて紹介した。

今後は、フィールドにおいて本ツールの有効性の評価を行い、操作性や機能、動作性能等を充実させていく予定である。