

## 組込みマイコンシステムにおける単体テストツールの開発\*

1 L-9

柴友理子 植木克彦 安田剛†

株式会社東芝 研究開発センター システム・ソフトウェア生産技術研究所‡

### 1 はじめに

組込みマイコンシステムの開発では、ハードウェアとソフトウェアを並行して開発する場合が多い。このとき、ハードウェアの信号値を読みとるような、ハードウェアに依存しているプログラムモジュール（以下モジュール）は、対象となるハードウェアが存在しないと機能しない。そのため、これらのモジュールの上位モジュールを単体テストする場合は、ハードウェアに依存しているモジュールの代替モジュール（スタブ）が必要[1]となるが、これらの作成には多くの工数がかかっていた。

また、現状の単体テストツールでは、スタブとなる下位モジュールの引数や戻り値に構造体・共用体・ポインタ変数が含まれている場合、構造体や共用体のメンバやポインタ変数の領域確保されているか把握することが困難なため、構造体・共用体・ポインタ変数にテストデータを与えることができない。そのため、ユーザ自身がスタブをプログラミングによって修正しなくてはならない。

これに対して我々は、組込みマイコンシステムの開発で使用するために、C言語に対応し、呼び出し時にユーザが指定した値を返すことができる単体テスト支援ツール「ドライバ/スタブジェネレータ」を開発した。

本稿では、この単体テスト支援ツール「ドライバ/スタブジェネレータ」で解決した問題点と特徴について報告する。

### 2 アプローチ

テスト対象となるモジュールを単体テストする場合、

- ドライバやスタブの自作した場合、自作したドライバやスタブもテストしなくてはならないため、単体テストに多くの工数がかかってしまう。
- 従来のスタブジェネレータを利用した場合、一種類のスタブを自動生成するため、ユーザが要求するスタブを作成することができないので、スタブを修正しなくてはならない。
- さらに、構造体・共用体のデータ構造やポインタ変数のメモリ領域の抽出が困難なため、ジェネレータにより自動生成されたスタブを修正しなくてはならない。
- ドライバやスタブに関わるデータ（モジュール名、モジュールの引数や戻り値、ドライバやス

タブの種類、テストデータ）をファイルに保存していないため、テスト対象となるモジュールの再テストが困難となる。

などの問題がある。

そこで、本ジェネレータでは、これらの問題点を以下のように解決し、ドライバ/スタブジェネレータの開発を行なった。

- ドライバやスタブのコーディングやテストからユーザを解放するために、ドライバやスタブを自動生成させる。ドライバは、テスト対象となるモジュールがメインモジュールである可能性が低いため、上位モジュールの有無にかかわらず自動生成する。また、スタブは、作成されていない下位モジュールの代わりに自動生成する（図1）。
- ユーザが必要とする機能のスタブを生成するために、モジュール毎あるいはモジュールの引数（スタブを呼び出すモジュールに値を返す引数）毎に、スタブの種類を選定し、各引数にテストデータを与える機能を用意する。
- ユーザが指定したソースファイル中から抽出した構造体や共用体のメンバにテストデータを与える機能を用意する。ポインタ変数の場合は、メモリ領域が確保されているかユーザに確認を要求し、その変数が配列の場合は要素数の入力をユーザに要求し、また領域を確保し、テストデータを設定する機能を用意する。
- テスト対象となるモジュールのドライバやスタブに関わるデータをファイルに保存する。再テスト時には、このファイルのスタブの種類やテストデータを一部修正することで、単体テストを容易にできるようにする。

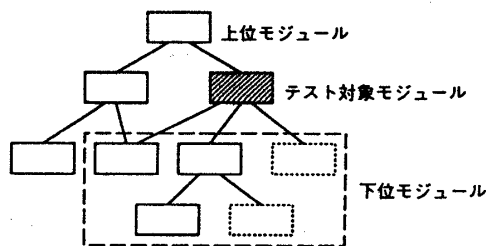


図1: ドライバ/スタブの位置づけ

### 3 特徴

本ジェネレータの機能の特徴について述べる。

\*The Development of a Unit Test Tool for Embedded Microcomputer Systems

†Yuriko SHIBA, Katsuhiko UEKI, Takeshi YASUDA

‡R & D Center, Systems & Software Engineering lab., Toshiba Corp.

### 3.1 ドライバ/スタブの自動作成

本ドライバ/スタブジェネレータは、テスト対象となるモジュール名や単体テストをする際に使用したいソースファイル名を指定することにより、テスト対象となるモジュールのドライバやスタブを自動作成する。作成時には、ドライバやスタブ機能の種類とテストデータを選定、入力することも可能である。これにより、ユーザが必要とする機能のドライバやスタブの自動生成が可能となり、テスト工数の削減がはかれると期待できる。

### 3.2 ドライバ/スタブの機能

#### スタブ機能の選定

本ジェネレータでは、モジュール単位で機能を選択できるだけでなく、モジュールの引数や戻り値毎に下の4つの機能を選択できる。これにより、テスト対象となるモジュールのスタブをユーザの要求に合わせ、作成できる。

本ジェネレータにより生成されるスタブは、大きく以下の4つに分類できる [2]。

- nop スタブ  
呼び出し時に何も行なわず、呼び出されたことを表示して、呼び出したジュールに戻る。
- 固定値スタブ  
あらかじめ用意していた値 (固定値) を引数に代入して、呼び出したモジュールに値を返す。
- テストデータ保持型スタブ  
あらかじめ用意していたテストデータ列を順次引数に代入して、呼び出したモジュールに値を返す。
- 問い合わせスタブ  
スタブがユーザと呼び出したモジュールのインタフェースとなり、対話的にテストデータを設定して、呼び出したモジュールに値を返す。

#### テストデータの入力

あらかじめテストデータを用意しておかなくてはならない固定値スタブ、テストデータ保持型スタブのを選定した場合には、モジュールの引数毎にテストデータの入力をユーザに供給する。また、入力されたテストデータは、そのテストデータを代入する引数の型に合致するかチェックする。

### 3.3 サポートするデータ型

本ジェネレータでは、基本的なデータ型 (文字型、文字列型、整数型、実数型)、構造体や共用体の抽出したメンバにテストデータを入力したり、またポインタ変数に領域を確保しテストデータを入力する。

- 構造体/共用体  
引数や戻り値が構造体や共用体の場合、構造体や共用体のメンバ名などのデータをユーザが指定したソースファイルから抽出し、ドライバやスタブの種類指定により、メンバ毎にテスト

データを入力する。このとき、テストデータの入力では、ユーザがテストデータを必要とするメンバのみに入力することにより、データ入力にかかる工数の削減をはかれるようにしている。

- ポインタ変数

引数や戻り値がポインタ変数である場合、メモリ領域が確保されているかユーザに確認を要求する。配列の場合は要素数の入力をユーザに要求し、テストデータを入力することができる。配列の要素数の入力では、定数だけでなく要素数を示す引数の変数を指定することもできる。ただし、実行時にポインタ変数の値が NULL の場合には、テストデータの代入は行なわない。

### 3.4 データファイルの再利用

本ツールは、テスト対象となるモジュールやソースファイルから抽出されたモジュール名、モジュールの引数や戻り値、また選定されたドライバ機能やスタブ機能や入力されたテストデータをデータファイルに保存している。これにより、この保存したデータファイルを再利用し、同じテストを何度でも繰り返し行なうことができ、また一部のドライバやスタブ機能の種類、テストデータを選定、指定しなおすことで違う条件下の単体テストができるようになる。

## 4 効果

本ドライバ/スタブジェネレータを開発することにより、以下のような効果が期待される。

- 単体テスト工程にかかる工数の削減  
構造体や共用体のメンバや必要に応じてポインタ変数にテストデータを与える機能により、スタブを修正する工数の削減ができる。
- 再テストが容易  
ドライバやスタブを作成する時に必要となるデータをファイルに保存することで、再テストをする場合も、そのファイルの一部のデータを変えるだけで単体テストを容易にできる。

## 5 おわりに

現在、本ジェネレータの評価を行なっているが、単体テストにかかる工数の1/4を削減できると予測している。今後は、東芝製 CPU、TLCS-900 上での単体テストを支援するため、ドライバ/スタブジェネレータの適応を行なっていく。

## 参考文献

- [1] 松田 茂広, "ソフトウェアのテスト技法 テスティングツール (3)", bit Vol.18, No.11.
- [2] 飛山 哲幸, 渡辺 俊雄, 明智 憲三郎, "ソフトウェア・テスト技法入門", インタフェース, Feb. 1988.