設計ツール

第6章

SpecC&EDAツール活用チュートリアル

枝 隆司

ここでは，SpecC言語を利用したLSIの仕様決め，および設計のフローを体験していただきます．使用するツールは本特集第4章でも紹介したVisualSpecとHDLシミュレータ「ModelSim」です．また，SpecC記述をVHDLに変換するモデル・コンバータも利用します．SpecC言語で仕様を記述した後，テストベンチになる部分はSpecC記述のまま残し，ハードウェアになる部分をVHDLに変換します．サンプルとなるシステムは，簡易電卓です．VisualSpecとModelSimのトライアル版，およびサンプル・データは本誌付属のCD-ROMに収録されています．（編集部）

1 はじめに

設計作業に忙殺されているエンジニアにとって，新しい設計言語を学ぶことは勇気のいることです．新しい言語を使って，まともに設計できるようになるまでには何ヵ月もかかるでしょうし，最初は試行錯誤の連続になるかもしれません．

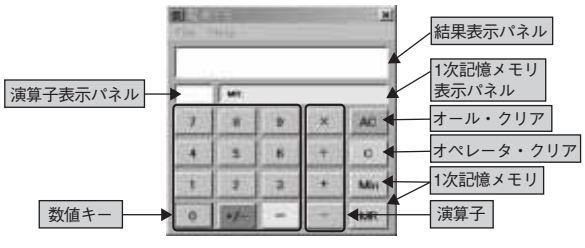
〔図1〕VisualSpec

VisualSpecの基本画面．左側のウィンドウは，ビヘイビア階層を表している．各ビヘイビアのアイコンをダブル・クリックすると，右側にビヘイビア編集ウィンドウが現れる．ポートやステートマシン（FSM）など，SpecC言語に特有の構文を，グラフィカルに表現できる．

SpecC言語はC言語のスーパセットなので，C言語を知っている人にとっては，比較的早く習得できると思います．それでも，新しい拡張部分を学ぶのはめんどうなものです．

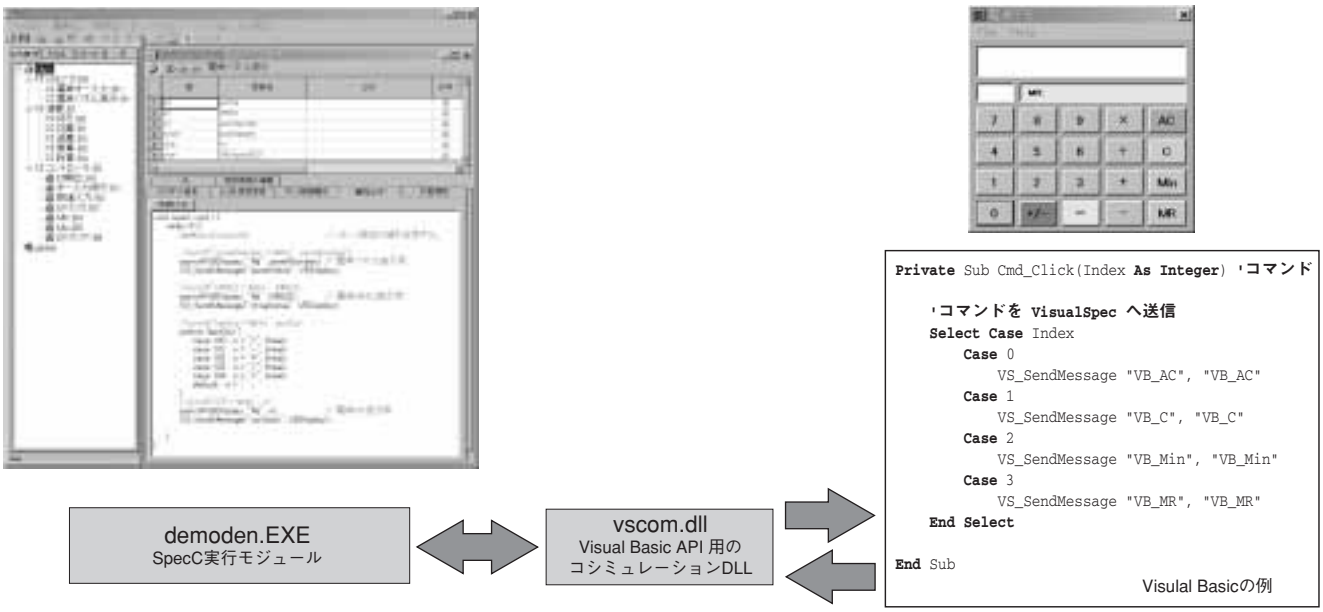
第4章および本章のチュートリアルで利用する「VisualSpec」は，SpecC言語のフロントエンドになり，SpecCの拡張部分をグラフィカルに入力できるツールです．たとえば，C言語にはない階層構造やコンカレント動作（並行実行）をテキストで記述するかわりに，**図1**のように，ツリーで表します．ビヘイビアの編集画面では，ポートやローカル変数を表形式で入力します．ビヘイビアの動作は，標準的なC言語を使用して記述します．C言語の文法を知っていれば，言語拡張部分の構文を意識しなくても，SpecC言語を利用できます．

本チュートリアルでは，SpecC言語をテキストで記述するのではなく，VisualSpecを使用してその概要を学んでいきます．題材としては，簡易電卓を選択しました．簡易電卓の仮想的なモックアップ（バーチャル・プロトタイプ）はVisual Basicで作りました（**図2**）．電卓のボタンを押すと，どのキーを押したかの情報が，Visual Basicインターフェースを通して，SpecC記述側へ転送されます．逆に，SpecC記述側のデータを仮想モックアップへ転送することも可能です．今



〔図2〕電卓のモックアップ

チュートリアルの題材として使用する電卓のモックアップ．Visual Basicを使用して作成した．各ボタンにはコマンドが割り付けられている．押されたボタンのIDコードが，メッセージとしてSpecC言語で記述された実行モジュールに送信される．実行モジュールからのデータを受け取り，表示パネルに表示することもできる．



〔図3〕VisualSpec とVisual Basic のテスト環境

SpecC 言語で記述した実行モジュールは，Visual Basic インターフェース（vscom.dll）を介して，電卓モックアップと通信する．demoden.exe はターゲットの回路で，電卓モックアップはテストベンチと考えることもできる．電卓モックアップのメニューから特別なコマンドを送信すると，実行モジュール内のセルフ・テスト機能が動作するように作成した．

回のチュートリアルで使用するテスト環境を図3に示します．

2 サンプル・デザインと利用ツール

チュートリアルで必要となる環境やツール，サンプル・デザインについて説明します．

2.1 サンプル・デザインの仕様

今回のサンプルである簡易電卓は，一般に販売されている電卓をまねて作りました．ここでは，処理を簡潔に行いたいので，整数演算のみを扱うことにします．たとえば，除算の結果に剰余があっても無視しますし，小数点以下は切り捨てます．

0から9までの数値キーは，数値入力に使用します．内部ではC言語の整数型（int），符号付き32ビットを利用しているので，扱う数値の範囲は－2147483648～＋2147483647になります．演算の結果，オーバフローやアンダフローが起こっても無視します．

演算子は，加減乗除の4種類です．それぞれ＋，－，×，÷に対応しています．内部では，C言語の演算子を使って処理を行っています．加算部分は，別階層にアダー（加算器）本体をおきました．いつでも，別のアーキテクチャをもつアダーと交換できるようにしました．特別な演算子として，＝があります．それを押すことにより，演算を開始します．内部では，演算子の優先順位を見ていますので，＝がなくても演算結果を出力することがあります．演算子の優先順位はC言語と同じで，＋，－より×，÷のほうが高くなります．

ACキーは，オール・クリアを行うキーです．すべてのオペレーションを終了します．1次記憶メモリも0に初期化されます．Cキー（オペレータ・クリア・キー）は，1回の操作をキャンセルします．たとえば，7＋5と打った後に5ではなく6にしたい場合，Cキーを押し，続けて6を入力します．Minキー（1次記憶メモリ・キー）を押すと，入力した数値や計算結果を保存します．MRキーにより，その値をいつでも呼び出すことができます．

結果表示パネルは，演算結果を表示します．演算子表示パネルはその時点で有効な演算子を，1次記憶メモリ表示用パネルはMinキーで記憶している数値を表示します．

＊　　＊　　＊

どうですか．上記の仕様を読んで納得がいきますか．まったくわかりにくい仕様ですよね．たとえば，連続した数値入力のすぐ後にCキー（オペレータ・クリア）が押されたらどうしますか．仕様には，そのことが書かれていません．このことに気がつかずに設計を進めて，後で仕様の変更を求められるかもしれません．

仕様の内容を誤解してしまうこともありますよね．仕様段階のエラーは，取り返しがきかないことが多いものです．こうしたことから，紙でもらう仕様書よりSpecC言語で書かれた実行可能な仕様書のほうが優れている，と筆者は考えています．

2.2 利用ツール

使用するマシンはWindowsが動作するパソコンです．OSは