第3章 特集 組み込みソフトとSOCのシステム設計最前線

USBシステムをSpecC言語で設計する

―― ハード構成を思い浮かべながら仕様を記述する

浅利康二

設計データ

SpecC言語を使ってマイコンとUSB用LSIからなるシステムを記述した例を紹介する．Cベース言語はまだ使用実績が少ない．そのため，Cベース言語を使ったシステム設計についても，具体的なイメージがわきにくい．ここではハードウェア構成を思い浮かべながらシステム仕様を記述した．本記事で紹介したすべてのSpecC記述は，本誌のホームページ(http://www.cqpub.co.jp/dwm/)からダウンロードできる．（編集部）

システム・レベル言語の一つであるSpecCが注目を集めています．これまで，LSI（ハードウェア）は主にVerilog HDL，VHDLなどのハードウェア記述言語を用いて，組み込みソフトウェアは主にC言語などを用いて設計されてきました．SpecCなどのシステム・レベル言語を利用すると，システムの中のハードウェアとソフトウェアを単一の記述言語で設計できるようになります．これは大きな意味を持ちます．

システム・レベルの設計では，実装方法に関する情報は定義せず，そのシステムのふるまいだけを定義することが理想です．システムのある部分をハードウェアで実現しようと，ソフトウェアで実現しようと，要求を満たしてさえいればどちらでもかまわない，というのがシステム・レベル設計の立場です．ハードウェアで実現するか，それともソフトウェアで実現するか（ハードウェア・ソフトウェア分割）は，性能を見積もれるようになった段階で決定されるべきです．その意味で，実装について，ハードウェアとソフトウェアの両方へのパスが用意されているSpecC技術は有効です．

しかし頭ではわかっていても，実際にそれを用いてシステムを記述してみないと，どのようなものなのかがよくわからないのではないかと思います．そこで，ここでは，USBシステムを例題として，実際にSpecC言語を使ってシステム記述を行ってみます．本稿を読んでいただくことで，SpecC言語を用いてどのようにシステムを記述できるかという点について，理解を深めていただければ幸いです．

●SpecCリファレンス・コンパイラをインストール

まずSpecCのリファレンス・コンパイラをインストールしましょう．SpecCのリファレンス・コンパイラには，コンパイラだけでなく，シミュレータとテストベンチも含まれています．いずれもオープン・ソースで，無償で入手できます．

コンパイラは，入力したSpecC記述をC++記述に変換して出力します（その意味では，コンパイラというよりプリプロセッサ）．シミュレータは，C++記述をコンパイル・実行する際に必要となります．さらに，コンパイラやシミュレータの動作を確認するテストベンチが用意されています．このリファレンス・コンパイラを用いると，SpecC言語で書かれたシステムのシミュレーションを行えるだけでなく，言語マニュアルでは書ききれない詳細な仕様まで確認できます．

このリファレンス・コンパイラは，SpecCの生みの親であるDaniel D. Gajski教授の所属する米国University of California, IrvineのWebサイト（http://www.cecs.uci.edu/~specc/reference/）で公開されています．そのほかの詳しい情報については，SpecC言語の普及推進・標準化を目的に活動を行っているSTOC（SpecC Technology Open Consortium）のホームページ

(http://www.specc.gr.jp/)を参照してください.

では,まずhttp://www.cecs.uci.edu/~specc/reference/からDownloadを選択し,SCRC Release V1.1(August 6, 2001)をダウンロードしてください.このリファレンス・コンパイラは,Solaris(デフォルト),SunOS,NetBSD,LinuxといったUNIX系の環境で動作します.ちなみに筆者は,本記事のシミュレーションにRedHat Linux 7.1 英語版の環境を使っています.GNU C++ compiler g++のバージョンなどは,SCRC Downloadページを参照してください.

また,Windows上でSpecC言語のシミュレーションを行いたい場合,インターデザイン・テクノロジーが開発しているVisualSpec 2001(販売はキャッツ,ソリトンシステムズ,東芝が担当)を利用できます.これは,グラフィカル表現を使ってシステム仕様を入力するツールです.VisualBasicなどで作成したGUIと連携動作させるためのインターフェースも備えています.

リファレンス・コンパイラのインストール方法は簡単です.ダウンロードしたファイルを解凍した後,scrc-1.1というディレクトリの下にあるINSTALLファイルの指示に従ってインストールします.ちなみにRedHat Linux 7.1の場合,INSTALLファイルの下のほうにあるSPECIAL INSTRUCTIONを見てください.make testというコマンドが正常終了すれば,インストールは完了です(テストにはかなりの時間がかかる).これでSpecC言語のシミュレーションを行える環境が整いました.

●SpecC言語の構文

リファレンス・コンパイラを使用する前に,SpecC言語について簡単に見ておきましょう.ANSI Cから拡張された構文(概念)として,以下のものがあります.

1)ブーリアン型(bool)
真(true),偽(false)のいずれかの値をとる.

2)ビット・ベクタ型(bit)
任意の長さのビット・ベクタをサポートする.

3)イベント型(event)
同期と例外処理のためのイベントを定義する.

4)タイム型(waitfor, delta)
タイム付き動作(ハードウェア)とタイムなし動作(ソフトウェア)をサポートする.

5)ビヘイビアのクラス(behavior)
ビヘイビアのオブジェクトで,演算が内部に記述される.

6)チャネルのクラス(channel)
通信を表すオブジェクトで,プロトコルなどが記述される.

7)インターフェースのクラス(interface)
ビヘイビアとチャネルの間を結ぶもの.

8)ポート(in, out, inout)
ビヘイビアとチャネルのコネクタの定義.

9)クラスのインスタンス
ビヘイビア階層の仕様と,ビヘイビアとチャネルの接続.

10)逐次型実行
逐次型の制御フローの仕様.

11)並列実行(par)
並列動作の仕様.

12)パイプライン実行(piped, pipe)
パイプライン仕様を明示的にサポートする.

13)有限状態機械(fsm)
有限状態機械(ステート・マシン)と状態遷移を明示的にサポートする.

14)例外処理(try-trap-interrupt)
実行の放棄と割り込みの処理.

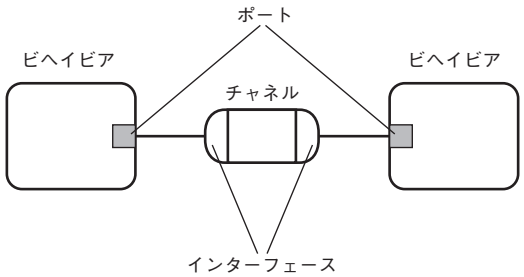
15)同期(wait, notify, notifyone)
並列動作の同期をサポートする.

16)タイミング仕様(waitfor, do-timing)
実行時間,遅れ,タイミング制約の明示的な仕様.

17)バイナリの導入(import)
ライブラリ・コンポーネントを再利用する.

18)永続的注釈(note)
永続的な設計注釈をサポートする.

上記のように,SpecC言語では,C言語に,主にハー



〔図1〕SpecC言語の基本構造

SpecC言語は,C言語にハードウェアを記述するための構文を追加した言語である.これらの構文を用いることにより,ハードウェアとソフトウェアの両方からなるシステムを記述できる.また,SpecC言語では,ビヘイビアとチャネルの二つを基本要素としてシステムを記述する.処理をビヘイビアの中に記述し,そのビヘイビアの間の通信をチャネルとして構成する.ビヘイビアはポートを介してチャネル内のインターフェースと接続する.