第1章 Altium社の統合開発環境「Altium Designer」

C言語を使用したFPGA設計を体験する

剣持裕治

オーストラリアAltium社の「Altium Designer」は，FPGAの開発からプリント基板の設計まで，電子機器開発の一連の工程を網羅する統合開発環境である．Summer 08版からC言語によるFPGA開発の機能が追加された．ここではC言語によるカスタム回路の設計からFPGA評価ボードによる動作までの流れを体験する．（編集部）

組み込み機器で使われるマイコンのプログラミングには，アセンブリ言語が主流であった時代がありました．限られたハードウェア・リソースを効率良く制御するには，ハードウェアを直接制御可能なアセンブリ言語による記述が最適だったからです．しかしプロセッサのアーキテクチャに依存した記述であるため，ほかの機器への移植性や後のデバッグ時の視認性に問題がありました．さらに，使用したいプロセッサごとに固有のアセンブリ言語を学ぶ必要がありました．

最近ではI/Oを直接制御したい場合や最適化を重視するなどの状況以外では，C言語のみでプロセッサの実行コードを記述することが多くなっています．この場合，使用するコンパイラの品質・性能やハードウェアを意識したコードの記述がより問題になります．組み込みソフトウェアの世界では，C言語に対するスキルが最も重要視されます．では組み込みハードウェアの世界ではどうでしょうか．

最近では組み込み機器でも，既存のディスクリート・デバイスに変わってプラグラマブル・デバイスであるFPGA（Field Programmable Gate Array）やCPLD（Complex Programmable Logic Device）の使用が飛躍的に増えてきました．このプログラマブル・デバイスは内部の論理回路を自由に書き換えることが可能で，まさに「柔らかいハードウェア」です．論理の記述も回路図，HDL（Hardware Description Language）などを適所で使い分けることができます．さらにFPGA内部にソフト・マクロのプロセッサを実装することで，「さらに柔らかいハードウェア」化も実現できています．この柔らかさを利用し，組み込みソフトウェア技術者が活躍できる環境が整ってきています．

最近では，プロセッサが実行していた関数や呼び出される変数をハードウェアに固定しプロセス全体の実行時間を短くすることや，ハードウェア記述言語であるHDLを使用せずに，標準のANSI C（ISO C 99）でハードウェア論理ブロックを記述することが可能になっています．これにより，ますますハードウェアとソフトウェアの境界線はあいまいになり，ハードウェアに直結したアセンブリ言語がC言語に置き換わってきたのと同じことが起きようとしているのです．とはいえ，今すぐにHDLをすべてC言語に置き換えることは時期尚早かもしれません．しかし，いま注目のC言語によるこれらの設計を，付属のDVD-ROMに収録されている「Altium Designer」を使用して体験してみましょう．

1．C-to-Hardware コンパイラ（CHC）

C-to-Hardwareコンパイラ（CHC）とは，C言語で記述した関数をRTL（Register Transfer Level），具体的には

Keyword Altium Designer，FPGA，ANSI C，C-to-Hardwareコンパイラ，ASP，JTAG，電卓，浮動小数点演算

図1　C-to-Hardware コンパイラ（CHC）

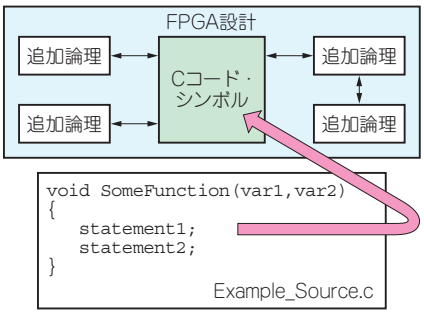

図2
FPGA のハードウェアをC コード・シンボルで表現

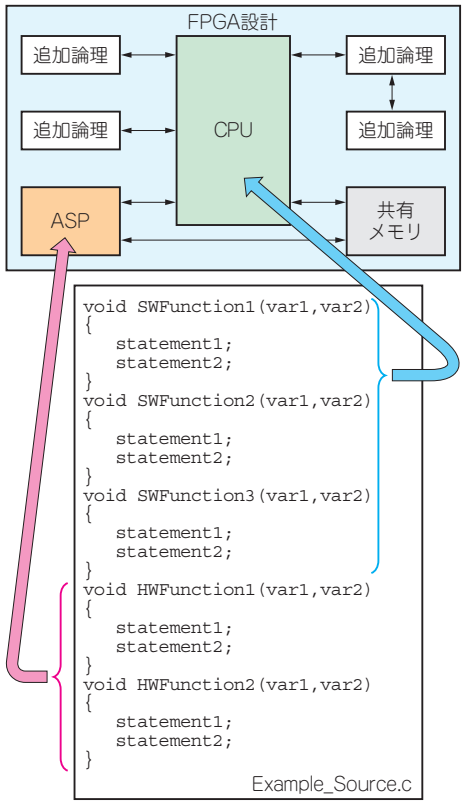

図3
プロセッサが実行している関数の一部をASP で実行

FPGA 内のハードウェア論理ブロックへ変換できるコンパイラです（図1）．

このコンパイラでは，ハードウェアの機能ブロックとして直接記述する方法（図2）と，プロセッサを動作させるために記述したC 言語の関数の一部を選んでASP（詳細は後述）で実行させる方法（図3）の2通りがあります．ASP とはアプリケーション実行速度最適化コンポーネントです．

前者はハードウェア側からの，後者はソフトウェア側からのアプローチと言い換えることができるでしょう．当然ながら，ハードウェア化した分はFPGA のリソースを消費しますので，アプリケーション最適化とリソース増大のバランスに関する見極めを注意深く評価する必要があります．

● ハードウェア側からのアプローチ

FPGA 内部の論理を表現する手法として，回路図やHDL が一般的に多く使われています．これに加えてC 言語を混在させた記述も可能です．

Altium Designer では，ポートとFPGA ピン番号の関連付けのため最上位階層は回路図であることが求められます．図4 に示すように回路図シートに配置された「C コード・シンボル」の下階層がC 言語による関数が置かれる場所となります．C 言語をC コード・シンボルへ変換（逆も可能）した際に，そのプロセスがクロックに依存した処理（Multi-cycle）なのか，逐次出力する処理（Combinatorial）なのかが選べます．

Combinatorial の場合は，ゲートのように入力値に対応した出力値が得られますが，時間軸を考慮した処理ではあ

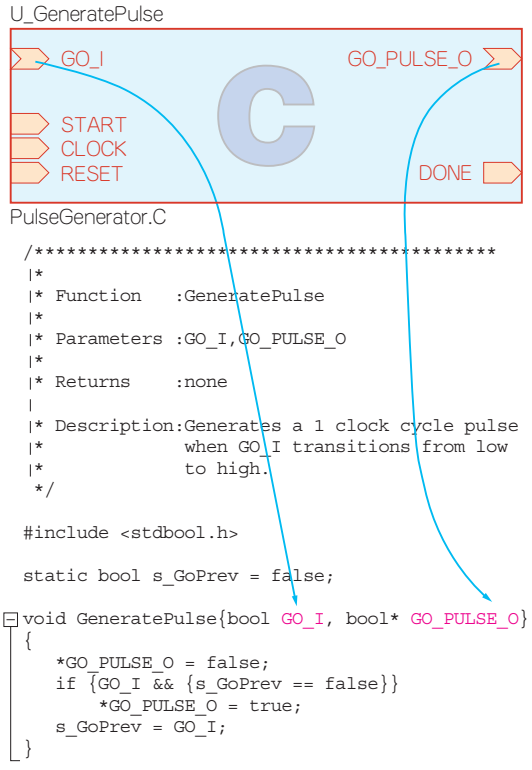

図4　C コード・シンボルと関数の関連
ポートが関数の入出力に対応している．