

第3章

無償で使えるVivado HL WebPACKで
高位合成にチャレンジしよう初めての高位合成ツール
Vivado HLSの使い方

小野 雅晃 Masaaki Ono

本特集では高位合成ツールVivado HLSの使い方を手順を追って詳しく解説します。本章で取り上げる例題としては、Cソースコードで記述すれば1行で済んでしまう乗算(掛け算)回路とします。ここでは、この最も単純な演算回路を高位合成ツールで合成し、IPコアとして使えるようにパッケージ化するまでを解説します。



図1 乗算回路

リスト1 乗算回路のC++ソースコード(multi_apuint.cpp)

```
// multi_apuint.cpp
#include <ap_int.h>

void multi_apuint(ap_uint<8> multi_in0,
                 ap_uint<8> multi_in1,
                 ap_uint<16> *multi_out){
    *multi_out = multi_in0 * multi_in1;
}
```

1. 高位合成ツールVivado HLSを
使ったFPGA開発の流れ

● 本章では何をするのか？

それではVivado HLS 2016.1で高位合成を始めましょう。初めての高位合成を始めるに当たって、Cソースコードは可能な限りやさしい記述を使用することにします。これは、記述の意味を考えずにVivado HLSの使い方に慣れるという目的のために、あえてやさしい記述を使用しています。

とても簡単なC言語の記述を例にして、Vivado HLSの機能について解説していこうと思います。使用するのは乗算1行のみのソフトウェアです。8ビット幅のmulti_in0とmulti_in1を乗算して16ビット幅のmulti_outとして出力します。ブロック図を図1に示します。そのソフトウェアをリスト1に示します。リスト1で使用しているap_uint<8>やap_uint<16>は任意精度の符号なし整数のデータ型です。ここでは、8ビット精度と16ビット精度を使用していますが、任意のビット精度を使用することができます。例えば、ap_uint<5>とap_uint<10>にすれば、5ビット精度と10ビット精度の符号なし整数のデータ型になります。詳しくは後述のCとC++の任意精度型について(p.18)をご覧ください。

● 評価ボードZYBOで動作確認

実装するターゲットFPGAボードとしては、ARMコア内蔵FPGA“Zynq”を搭載したZYBO (Digilent社)を使います(写真1)。

本章と次章では、Vivado HLSでIP化した乗算回路IPをVivadoのブロック・デザインでIPとして使用します。そしてVivadoを使って論理合成および配置配線を行い、ビットストリームを生成します。最後に実機ZYBOボードにダウンロードし、乗算回路の動作を実機で確認します。

なお、使用するCソースコードや制約ファイル(XDC)の内容は、本書サポート・ページや筆者のWebサイト「FPGAの部屋」(<http://marsee101.blog19.fc2.com/blog-entry-3329.html>)にアップロードするので、それをコピーして使用してください。

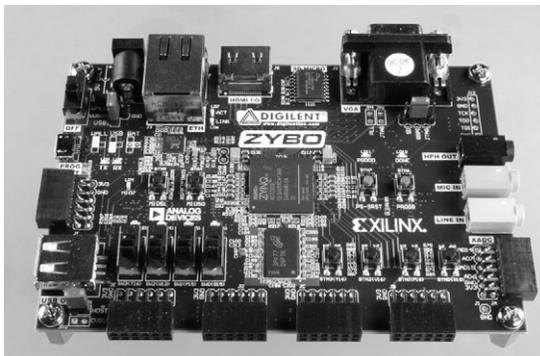


写真1 Zynq搭載評価ボードZYBO