

# 定番&最新FPGAの研究 ~ Xilinx 編 ~ ピン配置や動作クロック周波数などの 制約(Constraint)の指定方法

丹下 昌彦 Masahiko Tange

HDLを使ってFPGAを開発する場合、HDLにはその回路を何MHzで動作させるかを指定する方法がありません。そのため何らかの方法で希望する動作周波数を指定する必要があります。またFPGAは自由にI/Oピンの割り当てができます。GUI上からピンの割り当てを指定することもできますが、慣れてくるとここで紹介するucfファイルを使うのが一般的になります。今回はこのucfファイルの記述方法について解説します。

## 1 FPGA設計と制約

これまで(FPGA マガジンNo.1 ~ No.7), Xilinx社のFPGAを使用した回路の論理設計や、FPGAに組み込まれている専用回路(メモリ・コントローラ、RAM、クロック関連など)の使い方を中心に説明を行ってきました。

しかし、実際の基板にデバイスを実装して回路を組み込み、正しく動作するFPGAを設計するには、今回説明する「制約」が非常に重要です。

これをきちんと指定しておかないと、論理は正しいはずなのに動作しない、論理合成をやり直すたびに動作が異なる…などの厄介なトラブルに悩まされることとなります。ここでは制約の記述と、論理合成ツールの処理について触れてみたいと思います。

### ● FPGAの論理合成の過程

Verilog HDLやVHDLで論理回路を記述し、合成・配置配線を行う場合、次の順序で処理されます。

#### (1) 合成 (Synthesis)

HDLコードが解析され、文法チェック、RAMや乗算器などのインプリメント、リソースの共有、最適化などが行われます。結果はネットリストと呼ばれるファイルに書き出されます。

#### (2) 変換 (Translate)

ネットリストを読み込んで、Xilinx社デバイスの論理構成要素に変換を行います。

#### (3) マップ (Map)

Translateで書き出されたロジックを、FPGAエレメントに割り当てます。この段階で論理を構成するための部品(LUT、乗算器、IOBなど)が選択され、どのように接続されるかが決まります。

#### (4) 配置配線 (Place and Route)

FPGAエレメントを実際に配線します。配線が完了すると、FPGAの設計の大半の作業が完了します。

この後、FPGAデバイスに書き込むためのファイルに変換するなどの作業が行われ、一通りの作業が終了

し、実際に書き込んで動作させることができるようになります。

### ● 性能を出すためには制約が必要

FPGAの中身はマトリクス状に配置されたエレメントを自由に配線で接続することができる構造なので、非常に自由度は高いのですが、配線の方法(どの部分のエレメントを使い、どのように配線するか)によって性能が決まってしまうます。プリント基板に電子部品を配置してプリント基板を作成する場合でも、パターン設計が上手か下手かで性能が決まるのと同じです。

FPGA内部の配線は手で行うことも可能ですが(開発ツールISEでも手で配線を行うことは可能)、普通はツールお任せにしてしまいます。そうするとツールはFPGAやツールのやりやすいように勝手に配線をしてしまいます。ほとんどの場合、これでは目的の性能を発揮できるものにはなりません。

FPGAの配線は、プリント基板と同じように全ての信号を理想的な距離で配線することはできません。そこで、信号に一定の条件を付けてその条件を満足するようにするために、論理合成時に“制約”(Constraint)と呼ばれるものを設定します。

### ● 制約の種類

実はISEの制約には非常にたくさんの種類があります。

ここでは主に使われるもの限定して説明しますが、次のようなものがあります。

#### ● 入出力に関する制約

ISEでは、制約を全く書かなくても論理合成・配置配線を完了することは可能です。しかしその場合、FPGAのピンにどの信号が割り当てられるかもISEが勝手に決めてしまいます。

一般的に、FPGAを使ったハードウェアを設計する場合、どの信号をどのピンに割り当てるかは回路設計/プリント基板設計の段階で先に決められるので、勝手に決められたピン配置では困ります。

そこで、ピンにどの信号を割り当てるかを決めます。また、ピンに接続されるドライバの属性(ドライ