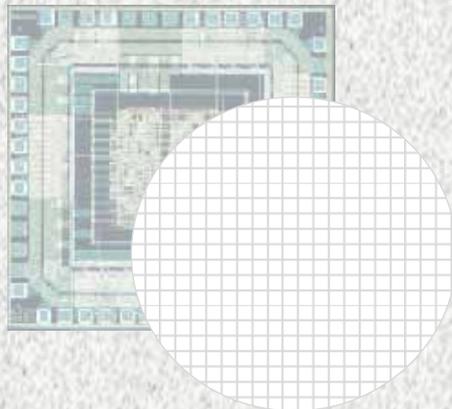


第3章 デジタル設計



第3章 デジタル設計

1 HDL について

1 - 1 HDL の概要

1.1 ハードウェア記述言語とは

ハードウェア記述言語 (HDL: Hardware Description Language) とは、その名のとおりハードウェア自体の動作を記述するための言語である。C 言語等のプログラミング言語 (ソフトウェア記述言語) がハードウェア (CPU 等) の動作手順を記述する (プログラムする) ための言語であるのとは根本的に異なる。しかし、「何かの動作をテキストで記述する言語」である点は同じなので、プログラミング言語と同様の考え方・手法が通用する場合が多い。

1.2 記述レベル

プログラミング言語が抽象的な記述をするもの (高級言語と呼ぶ) から、具体的な記述をするもの (低級言語と呼ぶ) まで存在するように、HDL も言語の抽象度合い (レベルと呼ぶ) により分類できる。HDL は以降に示す 4 段階の記述レベルに分類できる。

1.2.1 ビヘイビアレベル

ビヘイビアレベルは、回路の入力と出力の関係を記述するレベルである。入出力のサイクル数が定義されたものを BCA (Bus Cycle Accurate) レベルまたは動作レベルと呼び、ビヘイビア合成により RTL へ変換が可能である。例としては、System C がある。また、Verilog のシミュレーション用記述もビヘイビア記述だが、BCA ではないのでビヘイビア合成はできない。

1.2.2 レジスタ転送レベル (RTL)

RTL (レジスタ転送レベル: Register Transfer Level) はレジスタ (フリップフロップ) を明確に定

義し、その間の関係を記述するレベルである。RTLは論理合成によりゲートレベルに変換が可能である。RTLのツールとしては、Verilog (SystemVerilog)、VHDLが代表的である。

1.2.3 ゲートレベル

ゲートレベルはプリミティブセルの接続情報のレベルである。プリミティブセルとは基本的な回路部品のことである。セルベースICではプリミティブセルは様々な論理素子 (AND、OR、FF等) を自由に選択できる。一方、FPGA等のプログラマブルデバイスでは、ロジックスライスというような単位がプリミティブセルに該当する。このように、テクノロジーやデバイスの種類によりゲートレベルのデータは異なる (テクノロジー依存という)。ゲートレベルの回路データはネットリストとも呼ぶ。ゲートレベルでもVerilog/VHDLで扱うことが多い。Verilog/VHDLではゲートレベルはプリミティブセルのインスタンスと接続で表現される。

1.2.4 トランジスタレベル/アナログ記述

ゲートレベルを構成するプリミティブセルの中身はトランジスタ (NMOS、PMOS等) の接続情報で記述される。トランジスタレベルではSPICEによりアナログ動作のレベルで設計を行うのが一般的である。HDLでアナログを扱うものも登場している (Verilog-AMS)。

以上、ゲートレベルの方が記述レベルの抽象度は低くなっている。記述レベルの抽象度が高いほど記述量が少なくシミュレーションは軽く高速に実行可能で、抽象度が低いほど実際の回路に近づくがシミュレーションは重く実行速度は遅くなる。

複数のレベルが混在することを Mixed レベルと言い、デジタル/アナログが混在することを Mixed シグナルと言う。

1.3 代表的な HDL

代表的な HDL について説明する。

Verilog HDL

米国 Gateway Design Automation 社 (現 Cadence 社) により論理シミュレータ Verilog-XL のための言語として開発された。その後、1995 年に IEEE-1364-1995 として規格化されている。言語仕様としては C 言語に似ている。

第3章 デジタル設計

Verilog HDL 2001

オリジナルの Verilog に対する不満を解消するために 2001 年に大規模設計に対応するために仕様拡張が行われて、IEEE-1364-2001 となった。現在、Verilog 対応の主要なツールはこの規格に対応している。ただし、実行コマンドにオプションが必要な場合もある。

System Verilog

Verilog 言語をベースにモデリング機能と検証言語機能を導入している。

<設計向け記述>

- ・新しい変数（多次元配列・列挙型・構造体・共用体）
- ・合成結果を明確にできる always 文（always_comb, always_ff, always_latch）
- ・モジュール間接続を抽象化し記述を簡素にするインターフェースの概念。

<検証向け記述>

- ・アサーション記述
- ・制約付ランダム値生成
- ・機能カバレッジ

VHDL

米国防総省により機器の動作記述のためにドキュメント用言語として開発された。早くから（1987年）IEEE1067-1989として規格化された。Verilogに比較して言語仕様が厳格に定義されている。言語仕様としてはPascalやALGOLに似ている。

SystemC

主にアーキテクチャ設計用言語として使用される。記述が少なく、そのため高速にシミュレーションできる。C++のクラスライブラリという形で提供されている。そのため、C++のオブジェクト指向等の豊富な機能を使用することができる。ピヘイピアレベル合成も実現しており、BCAレベルで記述したものはRTLへ変換が可能である。設計環境はOSCI（Open SystemC Initiative）という組織のページから無料で入手できる。

Verilog-AMS

Verilog-AMS（Analog & Mixed Signal）は従来の Verilog のデジタル回路記述に加え、アナログ回路の記述を追加し、アナログ-デジタルの接続と混在が可能になっている。電圧・電流・抵抗・容量・インダクタンス等が変数として扱われ、時間微分・時間積分等の演算子を使用してキャパシタ・インダクタ等をモデル化できる。

シミュレータ/合成等の設計ツールが複数の言語に対応できることをミックスド言語と言う。システムレベルでの開発では、複数のレベル・言語を扱う場合が多くなっている。EDA ツールもそれに対応し、統合化が進んでいる。