

第8章 回路を書くために必要な最低限の知識

VHDL の書きかた & 読みかた入門

岡崎 光隆
Mitsutaka Okazaki

第6章や第7章では、VHDLの内容にまで踏み込んで説明しませんでした。プログラミング言語のようなテキスト記述から回路が合成できる、という手軽さに興味をもった人もいるのではないのでしょうか。本章では、VHDLの読みかたと書きかたを解説します。

本章で説明した範囲のVHDLコードを組み合わせただけでも、通常の回路なら十分記述することができます。

VHDL のあらまし

VHDLは、70年代末から80年代前半にかけて米国防総省で行われた「超高速ICプロジェクト(VHSICプロジェクト)」の研究成果として誕生したハードウェア記述言語です(図1)。

ハードウェア記述言語とは、**回路の構造と動作を文章で表現するための言語**です。米国政府に納品する回路の仕様書を書くために発明されましたが、VHDLから回路を自動的に合成するツールの進歩により、現在では回路設計用の言語として世界中で幅広く利用されています。

回路図入力方式は、数値計算を扱ったり、入力クロックに同期させる回路の設計には不向きで手間がかかりますが、VHDLなら**プログラミング言語のような数式や構造を使って直感的にわかりやすく記述でき**

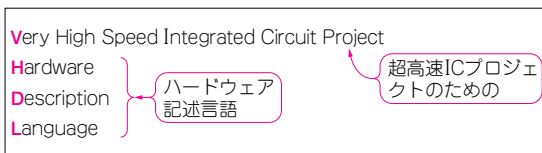


図1 VHDLって何の略?

す。回路図よりはむしろ、CやJavaなどのプログラミング言語を使うのと近い感覚で回路を設計できるのです。

エンティティ宣言部とアーキテクチャ宣言部の2部構成

● 回路とHDLの関係を見てみる

さっそく、ごく単純な回路をVHDLで記述して雰囲気慣れましょう。

▶ 例題回路

図2に示すのは、内部に回路が入っている箱です。箱の側面には、内部の回路と接続されている3本の入出力端子が付いています。

端子にはそれぞれ、a : IN, b : IN, c : OUTというラベルが貼ってあり、aとbが入力端子、cが出力端子になっています。箱の中にはANDゲートとNOTゲートが1個ずつ入っていて、箱の側面の端子aとbはANDゲートの入力に、端子cはNOTゲートの出力につながっています。

ANDゲートの出力とNOTゲートの入力は1本の信

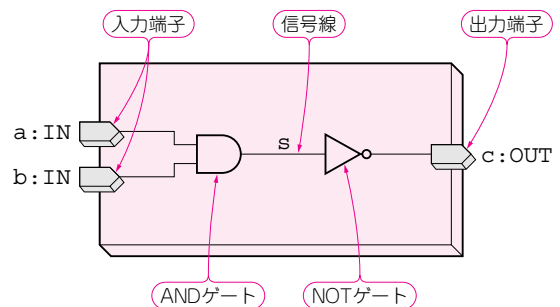


図2 例題…この回路をVHDLで書いてみると?

Keywords

VHDL, エンティティ, アーキテクチャ, ライブラリ, パッケージ, std_logic, 信号線, データ型, ベクタ信号, 論理演算, スライス, 接続, データ型変換, 条件付き信号代入, integer, Dフリップフロップ, プロセス, センシティブリティ・リスト, イベント属性, コンポーネント, ポート・マップ, 条件分岐

リスト1 図2の回路をVHDLで書くとこうなる

```

--
-- The first Example of VHDL code
--
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY nand_gate IS
    PORT (
        a : IN std_logic;
        b : IN std_logic;
        c : OUT std_logic
    );
END nand_gate;

ARCHITECTURE rtl OF nand_gate IS
    SIGNAL s : std_logic;
BEGIN
    s <= a AND b;
    c <= NOT s;
END rtl;

```

リスト2 メモの書きかた

```

-- これは普通のコメントです。
-----
-- 上下の行も最初が--で始まっているので、コメントです。
-----

ENTITY foo IS
    PORT ( x : IN std_logic );
END foo;

```

号線で繋がっていて、信号線にはsというラベルが貼ってあります。

▶ 例題回路をVHDLで書くと…

箱の内部回路をVHDLで表現すると、リスト1のようになります。以降、VHDLで書かれたテキスト文章のことをVHDLコードと呼びます。リスト1のVHDLコードは、大きく四つの部分に分かれています。

先頭のハイフン二つ(--)で始まっている3行はコメントです。コメントは回路としての意味をもちません。タイトルやメモを残す目的で使います。

次のライブラリとパッケージの使用宣言部は、VHDLのデータ型や演算子などの定義をVHDLコードから参照するために必要です。詳細は後述します。

この例では、このVHDLコードがライブラリIEEEのstd_logic_1164というパッケージを使うことを宣言しています。これは、VHDLの規格書で規定されているVHDLの標準パッケージで、VHDLコードを書く際にはほぼ必須です。詳細は後述します。

続くエンティティ宣言部は、回路の入出力端子の情報を書く部分です。

最後のアーキテクチャ宣言部は回路の具体的な内容を書くための部分です。

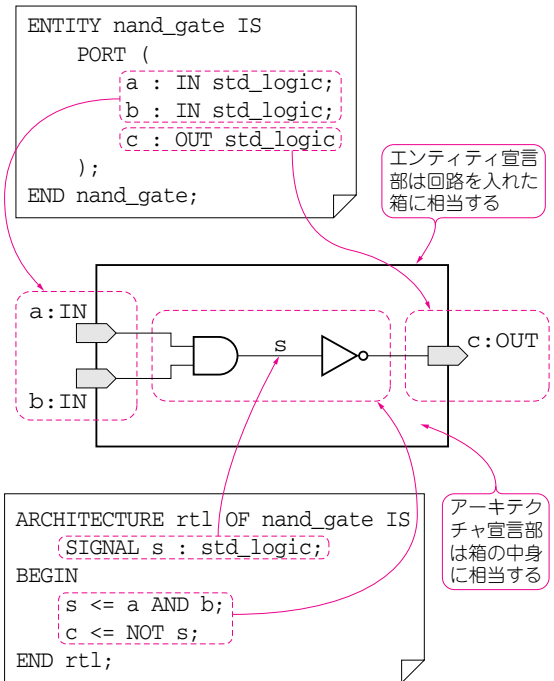


図3 VHDLコードと回路図の比較

● 外から見た宣言文と中身の宣言文で構成する

おおざっぱに言ってしまうと、図2の箱を外側から見たようすをエンティティ宣言部に、箱の中身に相当する部分をアーキテクチャ宣言部を書くわけです。

図3に示すのは、リスト1のVHDLコードと回路の対応です。端子のラベル名や信号線のラベル名に注目して比較すると、両者の関係がなんとなく見えてきたのではないのでしょうか？

1 エンティティ宣言部の読みかた

■ 1-1 メモを残す「コメント文」

リスト1の最初の3行のように、二つのハイフン"--"を書くと、それ以降から行の終わりまで、コメント文として扱われます。この部分は回路として扱われることはありません。コメント文は、VHDLソースのどの行に書いてもよく、リスト2に示すように、行の途中から始めてもかまいません。

■ 1-2 おまじない「ライブラリとパッケージの使用宣言」

● 使用する用語の種類を定義する

VHDLコードを書くときはまず、使用する用語の種類を宣言しなければなりません。

用語をまとめたものをパッケージと呼びます。また、