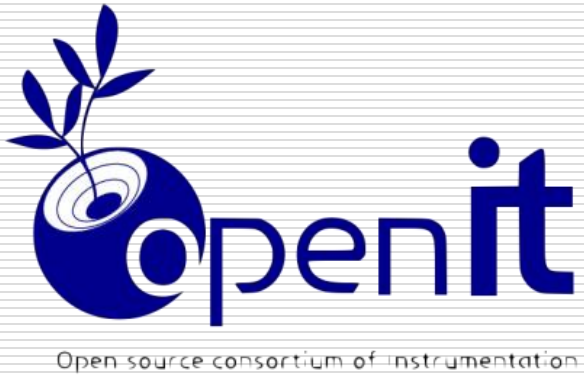
Open-It FPGAトレーニングコース（入門編）

2.1 Verilog-HDL記述（組み合わせ回路）



第3.4版

2016年02月22日

3.4.1. HDLを使用する理由

何故回路図ではいけないのか？

HDL:
Hardware Description Language

テキストファイルで回路を表現する為の言語

ソフトウエア・プログラムの様に書ける

代表的な言語はVerilog-HDLとVHDL

最近はHDLが主流です

ちなみに、ISEは回路図入力もサポートしています

回路表現方法

部品とその配線情報が回路情報

4要素の接続で決まるので、その接続を表現できれば良い

主な表現方法

- 回路図
- Hardware Description Language (HDL)
  - Verilog-HDL, VHDL, SystemCなど

FPGA回路はHDLで書く事が多い

何故HDLを使うのか？

回路図入力には幾つかの問題がある

- 作業効率が悪い
  - 基本的に1信号を1本の線で表現
  - 入力ミスが多くなる
- 大規模回路では読みにくい
  - 回路図が複数枚になると信号の接続先を調べるだけでも大変

HDLの魅力

- 設計作業を効率よく行うために誕生
- 効率良く入力できる事で
  - 入力ミスを少なく
  - 大規模回路が扱えるようになった
  - 読みやすい
- 他には、
  - コードを仕様書に近づける

回路動作の理解は回路図の方が優れています。
ブロック図、回路図なども組み合わせて設計を進めてください

HDLの注意

- 魔法の言語ではありません
  - HDLを使えばハードを知らなくてもハード設計ができると思っている人がいますが・・・
- 作業効率を上げる為の手法です
  - それ以上の期待をしてはいけません
  - 回路図の接続情報をテキストファイルで表現しただけです
- 回路を理解しなければ設計はできません

Verilog HDL

- テキストファイルとして書きます
  - テキストエディタを使って書きます
    - 各自好きなエディタを使ってください
  - 通常拡張子は.v
- 文法がC言語に似ている
  - しかし、動作はまったく異なるので注意！！
  - 回路図からHDLへ書き直すと理解しやすい

最初に衝撃的な事実！

HDL文法的に正しくても
動かない回路が生成される事がある！

この様な事が起きるのでハードウエア設計が難解だと思う人が少なくないようです

回路を考えずにコーディングしてはいけません。

回路を考え、その回路をHDLで記述して下さい。

何故、その様な事が起きるのか？

- シミュレーション用の記述も含まれている
  - 抽象度が高い記述
  - シミュレーションで使用する時はPC上の動作に限定されるので回路化する必要が無い
- 回路を言語で記述するために設計されている
  - 言語で回路を記述、生成するためではない

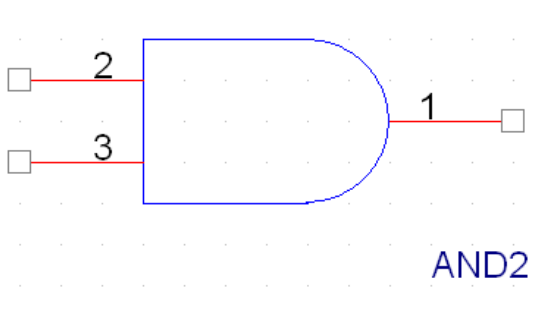
どの様に対応するのか？

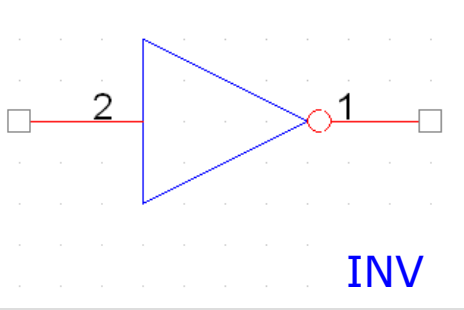
- 先ず回路を設計する
- 設計した回路を言語を用いて記述する
- 代表的な記述スタイルを真似する
  - ツールのHDL記述ガイドなどが参考になる

3.4.2. Verilog-HDL

デジタル回路の構成要素

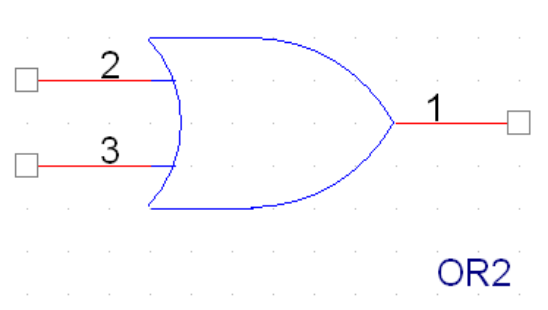
O=A&B;

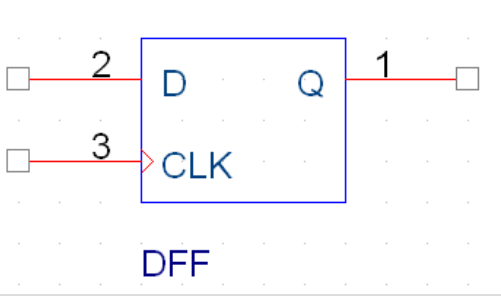




O=~I;

O=A|B;





全てのデジタル回路は
この4つの要素（回路）のみで構成されている

4要素のHDL記述を知っていれば最低限の知識としては十分

トレーニングコースで使用する文法のみ解説します

ネットワークに接続できる方は「Verilog-HDL入門」も参考にしてください

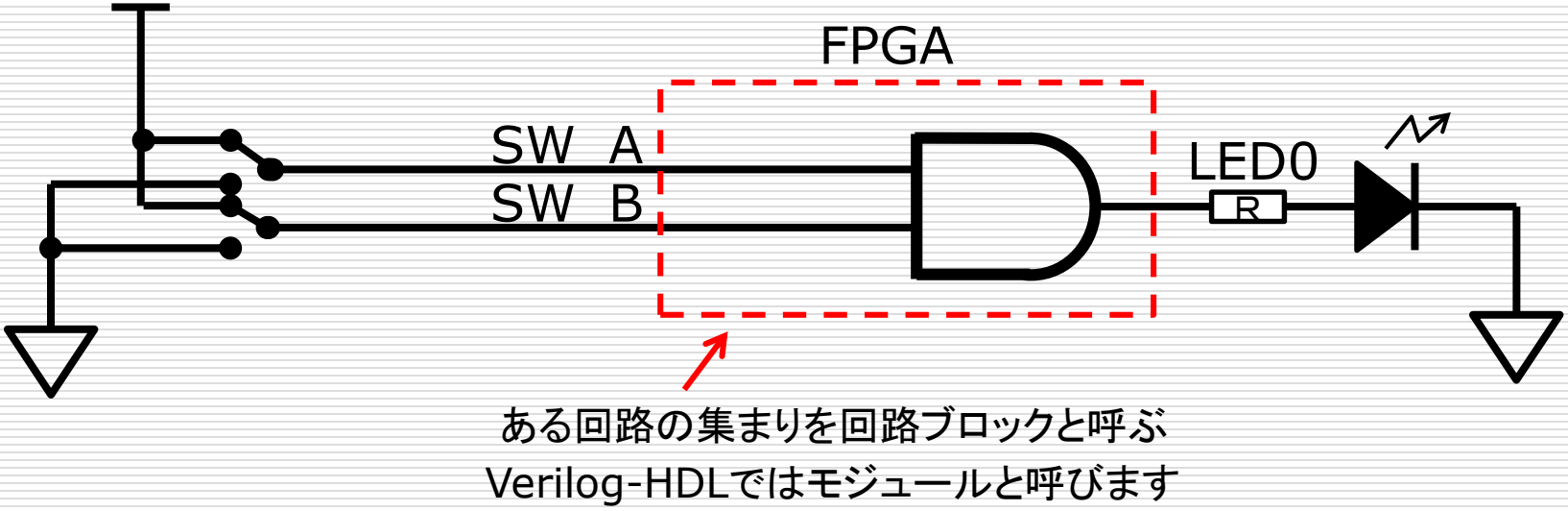
書籍に色々な書き方が紹介されていますので
気に入った、凝った書き方や美しい書き方を見つけてください。

Verilog-HDL概要

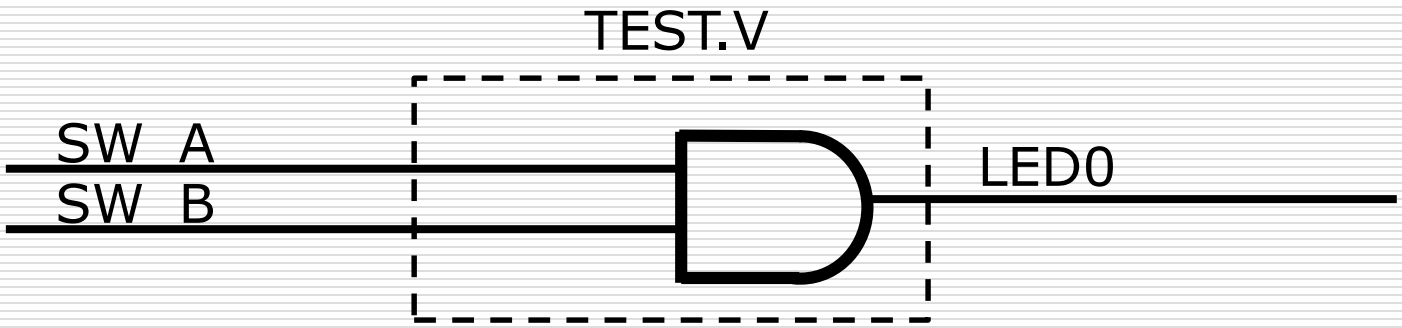
- Verilog-HDLコードはテキストファイルとして作成する
- 拡張子は".V"
- 通常1ファイルで1回路(モジュール)
  - 数字から始まるモジュール名は使用しない
  - 文法では決まっていないが問題を起きにくくするために奨励
- 大文字、小文字は区別される
- コメントは/*から*/、または//から行末まで
- 基本的に1文は;で終わる
- 全角文字はコメント以外では使用できない
  - エラーが出るが、理由が表示されない！！
  - 特に全角スペースに注意！！エディタ上での発見が困難！！

目標

ANDをVerilog HDLで記述してみましょう



ANDゲート回路



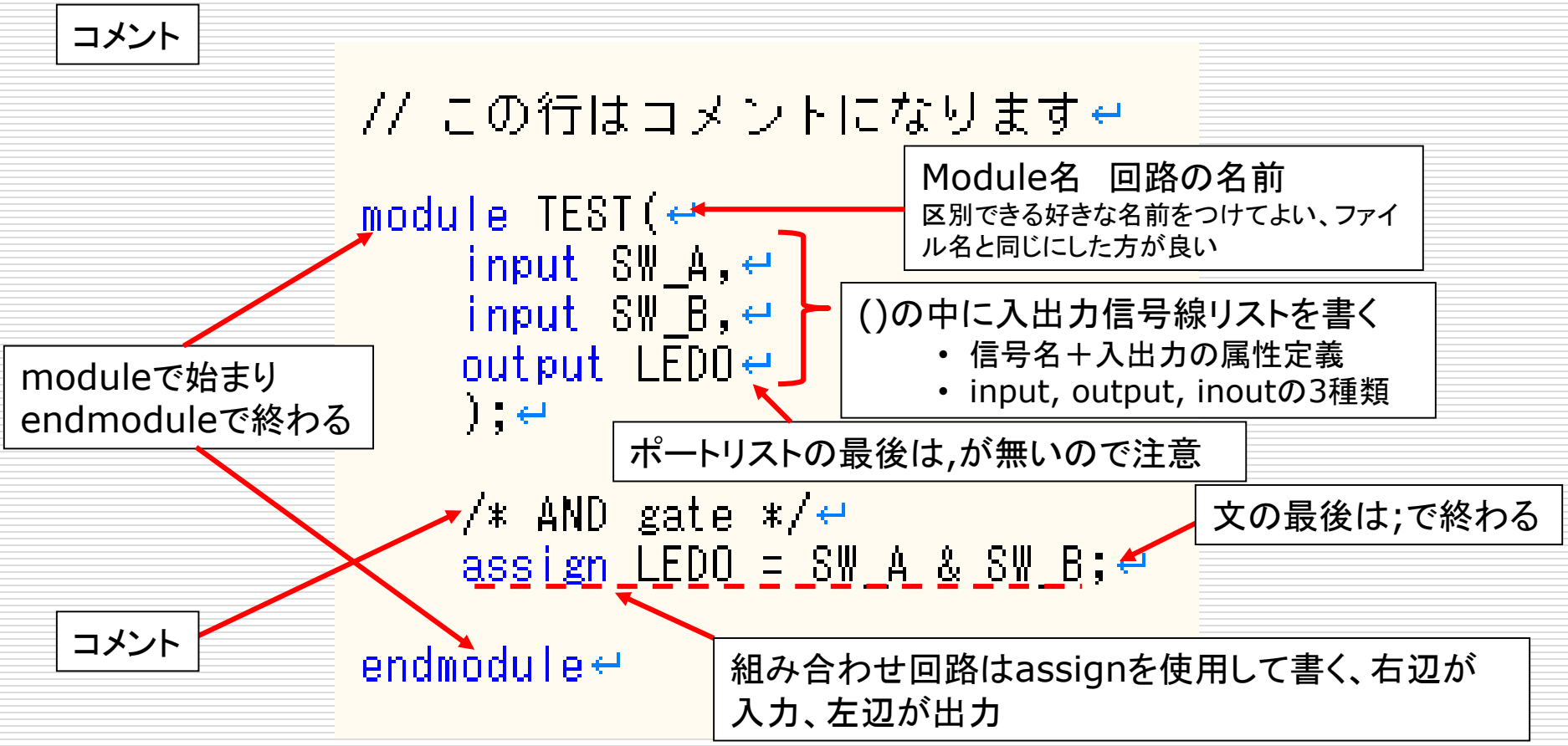
入力信号：
SW_A, SW_B

モジュール名：TEST
ファイル名：TEST.V

出力信号：
LED0

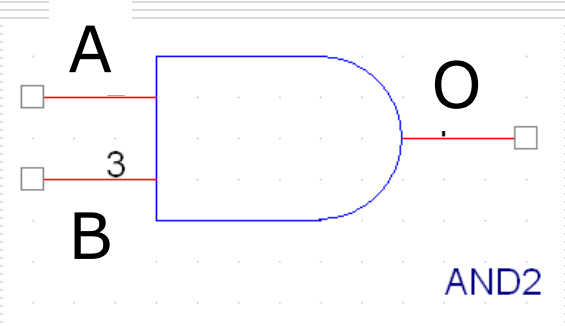
このモジュールをVerilog-HDLで書くと・・・・

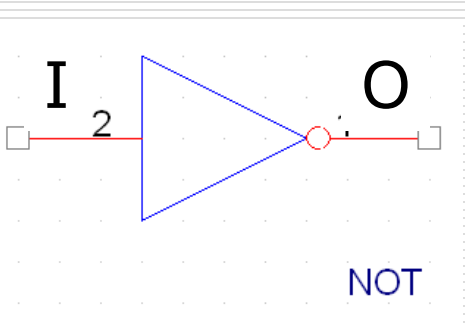
ANDゲートのHDLコード



演算子

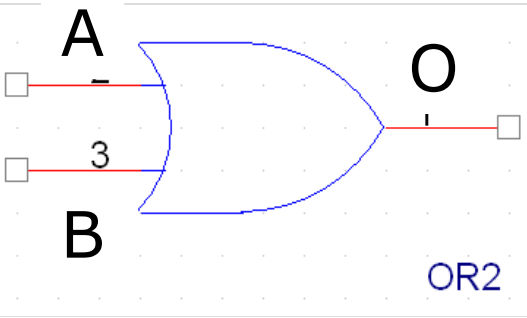
O=A&B;

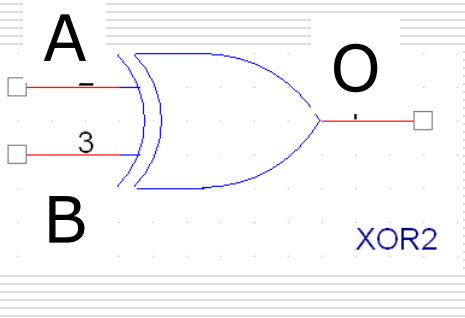




O=~I;

O=A|B;





O=A^B;

Assign文

- 組み合わせ回路出力はassign文をつかう
  - 例）AND回路、 assign O = A &B;
- wire宣言
  - 明示的に組み合わせ回路が出力する信号である事を示す
  - 例）wire O;
  - 使用されているassign文よりも上の行で宣言する必要あり
  - output信号はwire宣言してはいけない
    - Output記述はwire宣言と同じ

3.4.3. Verilog-HDL文法補足

マルチビット信号の書き方

- マルチビット信号
  - 複数本で一つの意味をなす信号
  - 本数をビット幅と言います。
- 例えば、”data”と名付けた信号が8本で構成されている時は”data[7:0] ”と書きます
  - 慣習的にLSBを0とすることが多いです

定数の書き方1

- 例、4'd3
  - 4はビット幅を意味します。
  - dは10進表現を意味します
    - 他に、hで16進、bで2進を表すことができます。
  - 3が値です。ここではd3なので十進数で3であることを表しています。
- 例2
  - 5'h1Fは16進数で1F,すなわち10進表現で31を表しています。

定数の書き方2

- 省略
  - ビット幅を省略するとビット幅32
  - 単に数字を書くと10進数表現
- 読みやすく為にスペースの代わりに_を使用することができます、32'h0123_4567など

履歴

- 2012/5/17 第1.0版 ISE13.4対応　内田智久(Esys, KEK/総研大), 林達也（大阪大学）
- 2014/8/7　第2.0版 章構成変更　内田智久(Esys, KEK/総研大)
- 2015/7/31 第3.0版 章構成変更　内田智久(Esys, KEK/総研大)
- 2015/8/6 第3.1版 誤字修正　内田智久(Esys, KEK/総研大)
- 2015/12/03 第3.2版 誤字修正　内田智久(Esys, KEK/総研大)
- 2016/01/27 第3.3版 誤字修正　内田智久(Esys, KEK/総研大)
- 2016/02/22 第3.4版 誤字修正　内田智久(Esys, KEK/総研大)