



HDL 記述による設計法をマスターする 実験で学ぶロジック回路設計

木村 真也
Shinya Kimura

第1回 0と1を使った計算をマスターする

前号までは、ロジック回路設計に欠かせないツール、「HDL シミュレータ」のしくみを解説してきました。今月号からは心機一転、仕切り直してHDLを使った実際のロジック回路設計の基礎を解説します。

ゴールは2006年4月号付録CPLD基板を使ったテレビ・ゲームを完成させることです。連載で解説される事柄を身につけていけば、自分の欲しいロジック回路が作れるようになるでしょう。 <編集部>

近年、ロジック回路(論理回路)の設計は、回路図を描く方法からハードウェア記述言語(HDL)を利用する方法へと移り変わってきました。

FPGAやCPLDなどのプログラマブル・ロジック・デバイスが急速に普及し、アマチュア・レベルでも大規模で複雑なロジック回路を簡単に作成できるようになってきています。

▶ロジック回路の基礎知識をもっていないと困る

HDLは、一見するとC言語のようなプログラム言語に似ています。しかし、HDLはあくまでも「回路」を記述するための言語です。

HDLの文法を知っていれば、ロジック回路を実装することはできます。しかし、ロジック回路の基本的な知識がないと回路規模や動作速度に問題が発生した場合、対応が難しくなります。また、高品質な回路を設計するためには、ロジック回路そのものの基本的な

知識は必要不可欠といえます。

このような観点から、この連載では、ロジック回路の基礎とHDL記述の方法を関連付けながら解説していきます。

▶Verilog HDLを使って実験をしながら解説

使用するHDLはVerilog HDLです。文法が比較的簡単であることと、これまでの連載でVerilogシミュレータを取り扱ってきたことから、引き続きVerilog HDLをベースに解説を進めます。

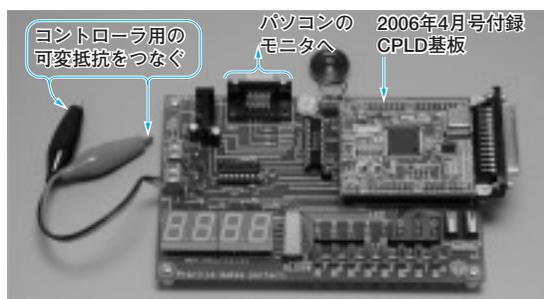


写真1-1 この連載で実験に使うロジック回路実験ボード
2006年4月号付録CPLD基板を利用する。詳細は連載第3回以降に解説する予定。動作確認用のLEDや入力用スイッチ、モータ出力用のバッファ回路などを集めたもの

Keyword 1

HDL

Hardware Description Language(ハードウェア記述言語)の略語で、ロジック回路の機能、動作、構成をプログラミング言語に類似したスタイルの文字列で表現したものをいいます。

ロジック回路を言語で記述する試みはかなり古くからありましたが、最近になって二つの重要なツール「シミュレータ」と「論理合成ツール」の実用化とともに、本格的に普及してきました。

集積回路の大規模化が著しい一方、一人のエンジニアが扱える回路図ベースの設計規模には限界があります。それ

を打破する方法としてそれらツールの開発があったことが、HDL普及の背景にあります。

HDLではロジック回路を直接表現するのではなく、抽象化して「どのような動作をするか」を規定することで、設計工数の削減や設計対象の拡大を可能にしています。

その代わりに、抽象化して記述した設計対象の動作確認にはシミュレータが必要となり、集積回路として製造するためには具体的なロジック回路へ変換するツールとして論理合成ツールが不可欠です。

HDL 記述と実験を結び付けて解説する

● 技術を身に付けるには実験を行うことが大切

多くのことに言えますが、解説文を読んだだけではなかなか身に付くものではありません。実際に試しながら話を進めていきます。

まずはシミュレータで動作確認をして、次に実際のロジック回路で動作を確認してみることで、理解が深まり、内容をしっかりと身に付けることができます。

● 2006年4月号付録 CPLD 基板を利用する

ロジック回路の実験には、本誌2006年4月号の付録 CPLD 基板 (Altera 社の MAX II を搭載) を使います。付録基板には書き込み回路以外ほとんど搭載されていないので、実験には不足です。実験に必要な機能を搭載した「ロジック回路実験ボード」(写真1-1)を追加して使用します。ロジック回路実験ボードについては第3回以降に解説します。

● テレビ・ゲームの製作を目標にする

連載のゴールとして、テレビ・ゲームの製作を目標にします。パソコンのモニタ上でボールを移動させ、ラケットではねかえす単純なものです。

ラケットのコントロールは可変抵抗器(ボリューム)を使います。また、得点表示や効果音の発生も行います。これらの機能の一つずつ、シミュレータやロジック回路実験ボードで動作確認したあと、ロジック回路実験ボードで実際に動作させて、設計を進めていきます。

● 連載の最初はロジック回路設計のための基礎知識

これまで説明したように、HDL 記述と実験を組み合わせることで連載を進めていきます。

ただし、今回と次回の2回では、ロジック回路を設計する際に必要となる基礎知識をまとめておきます。これらの基礎知識を知らないと、単純なロジック回路

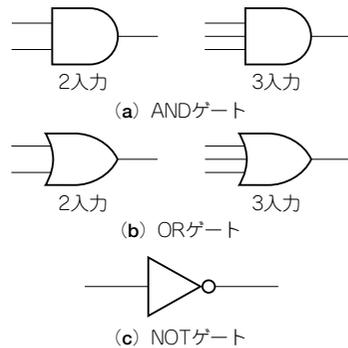


図1-1 ロジック回路の構成単位である基本ゲートの回路記号
一番多く使われている MIL スタンドアードの記号。これ以外の記号もある

を考えるのにも困る場合が出てくるからです。

ロジック回路を構成する基本部品は三つ

● ロジック回路を構成する三つの部品が基本ゲート

「ロジック回路」は、「デジタル回路」あるいは「論理回路」など、いろいろな呼び方があります。

そのロジック回路は0と1で動作している、という話をどこかで聞いたことがあると思います。

ロジック回路の応用範囲は広く、デジタル時計からコンピュータまで、いろいろなものがあります。

しかし、その実体は基本的な三つの部品でできています。それらはANDゲート、ORゲート、NOTゲートと呼ばれるもので、基本ゲートと呼ばれています。

回路記号としては、図1-1に示すものが広く使用されています。この記号はMILスタンダードと呼ばれるものです。

ロジック回路はこの三つの基本ゲートが基本部品で、どのような高度なロジック回路もこの三つのゲートで構成できると言っても過言ではありません。

実際のロジック回路はいろいろな回路構成で作られていて、基本ゲートそのものではないことが多いのですが、ロジック回路を扱ううえでは、この三つのゲート

Keyword 2

ゲート

ブール代数の演算を電子回路で実現したものを論理ゲート(あるいは単にゲート)と呼んでいます。

ブール演算機能を電子回路化する場合、ANDゲートと

表1-A NANDゲート/NORゲート/ExORゲートの真理値

入力		NAND 出力	NOR 出力	ExOR 出力
A	B	$\bar{A} \cdot \bar{B}$	$\bar{A} + \bar{B}$	$A \oplus B$
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	0	0	0

ORゲートはあまり使われません。

実際の回路には、ANDゲートとNOTゲートを結合した

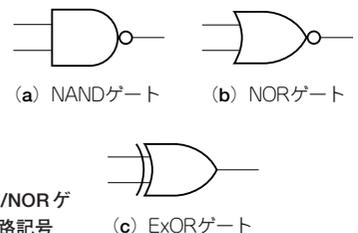


図1-A NANDゲート/NORゲート/ExORゲートの回路記号