

第2章

回路を作ってみよう

～回路図を描き, FPGA/PLD にインプリメント～

田中良平/八木憲司

第1章ではXilinx社とAltera社の開発環境のセットアップについて解説しました。本章では、開発環境を使って実際にサンプル回路を作成します。本章を通じて開発環境を使うことに慣れてください。「自分の力でできた」という成功体験が自信

につながるものと思います。前半のXilinx編では着メロ・コード入力/演奏回路を、後半のAltera編ではナイトライダー回路を作成します。
(編集部)

..... Xilinx 編

Xilinx社製FPGAを使って着メロ・コード入力/演奏回路を作成しましょう



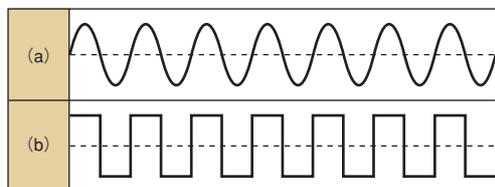
本章の前半で作成する着メロ・コード入力/演奏回路(以降、着メロ回路)は、ハードウェア面での手間を省くため、生成した波形の加工(フィルタリングなど)はしません。しかし、意外と聞ける音質になっていると思います。音階は32、楽譜は224音符まで入力可能です。もう少し音階の幅や、入力できる音符量を大きくしたかったのですが、今回使う評価ボードに搭載されているFPGA(XCS10)では、これが限界です。興味があれば、さらに大容量のFPGAを使って、より高度な着メロ回路を作れます。もちろん、Xilinx社以外のFPGA/PLDを使うこともできます。記事を読んでぜひ挑戦してみてください。

2.1 着メロ回路の方針

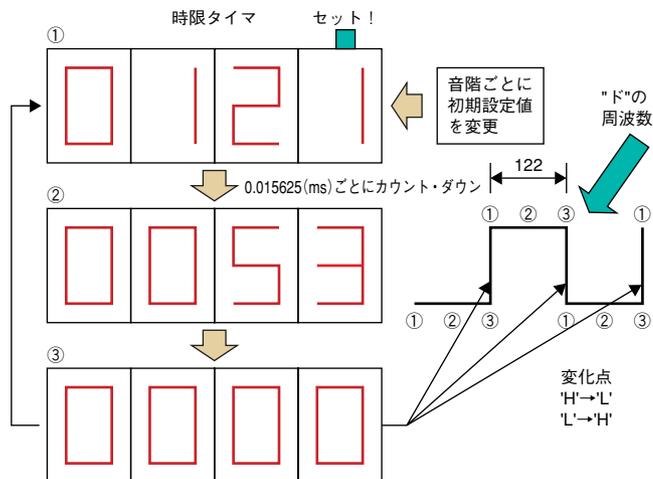
● 音声の発生原理は電子楽器のシンセサイザと同じ

回路図を入力する前に、どうしてFPGAで音が作成できるのか、その原理を説明します。とくに興味のない方は、2.2節から読み進めてもかまいません。

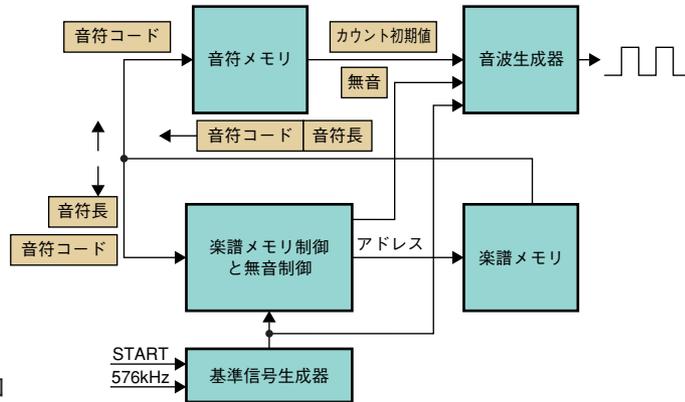
音の基本形は、図2-1(a)に示すように正弦波です。その周波数を変化させると、人の耳には、いろいろな音階の音として聞こえます。今回のサンプル回路では、図2-1(b)に示す方形波を作成して、周波数だけを原音とほぼ同じにして出力し



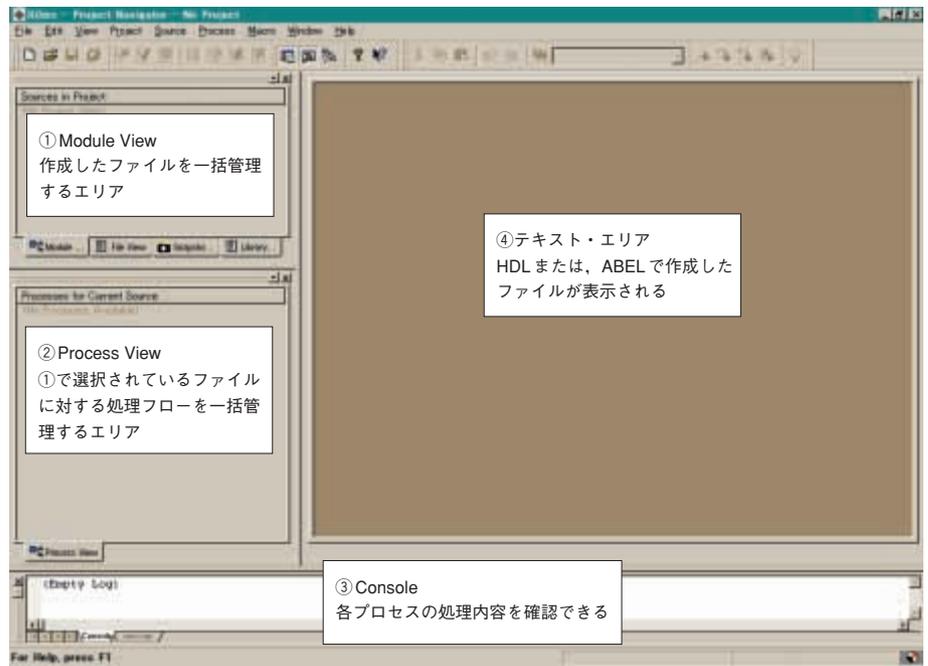
〔図2-1〕 音波と方形波



〔図2-2〕 音階生成の原理



〔図2-3〕着メロ回路の機能ブロック図



〔図2-5〕Project Navigatorの初期画面



〔図2-4〕Project Navigatorの起動
ダブル・クリックして起動する。

ます。方形波ではノイズ(高調波)を含み、音叉を打ち鳴らしたときのような音は聞こえませんが、今回は手軽さを追求しているので、図(b)から図(a)への波形加工は行いません。

● 希望の周波数を作るにはカウンタのカウント値を利用する

FPGAには576kHzのクロック信号を入力するものとします。まず、これを9分周して64kHzの音階生成用基準信号(方形波)を作成します。各音階は、この64kHz周期のパルスを元に生成されます。つまり、 $1/64 \text{ (ms)} = 0.015625 \text{ (ms)}$ ごとにカウント・ダウンするダウン・カウンタを作成し、カウンタのプリセット値を替えることで希望する音階を生成しようというものです。

たとえば、“ド”の音は261.626Hzです。261.626Hzの周期

は64kHzの約244倍です。したがって、図2-2のように、64kHzの周波数でダウン・カウンタを122回まわします。カウンタ値が0になるごとにFPGAからの出力信号を‘1’、‘0’と交互に切り替えれば、立派な“ド”の音になります。

● 着メロ回路全体のブロック図

着メロ回路は、図2-3に示す構成となります。

- (1) 基準信号生成器：64kHzのパルスを発生する
- (2) 音波生成器：音階データに対する波形を発生する
- (3) 楽譜メモリ制御と無音制御：楽譜データの読み出しと、無音時の制御を行う
- (4) 音符メモリ：各音階ごとのカウント初期値を書き込んだROM
- (5) 楽譜メモリ：楽譜データを保存しているROM