

プロローグ

サンプリング周波数を48kHzから44.1kHzにする
レート変換器をどのように実現するか？

デジタル信号処理を実現する五つの方法

岩田 利王
Toshio Iwata

● 48kHzを44.1kHzに…サンプリング・レート変換器を題材に

「デジタル信号処理」とは、従来アナログで信号処理(フィルタリング, ノイズ・キャンセル, 周波数解析, 特徴抽出など)していたものをデジタルで行う手法をいいます。

デジタル信号処理の代表格はなんといっても「デジタル・フィルタ」です。また、そのデジタル・フィルタを駆使した応用例に「サンプリング・レート変換器」があります(図1)。

今回の特集では、そのサンプリング・レート変換器をどのようにFPGAに実装するか、その手法・手順・勘所を紹介します。

1. 一口に「デジタル信号処理」といってもいろいろなやり方がある

デバイスの集積化が進んだおかげで、近年の信号処理は、FPGAやマイコン上でデジタルで行うのが中心になってきました。またパソコンやスマートフォンが身近になったことから、Windowsアプリケーションなどのソフトウェア上でデジタル信号処理を行う例もよく見かけます。

● ソフトで実現/ハードで実現, C言語/HDL, FPGA/マイコン…どれも立派なデジタル信号処理

図2に「デジタル信号処理を実現する五つの方法」を示します。本特集ではこれら五つについてそれぞれ説明したいと思います。

● ソフトで実現かハードで実現か

まず、大きくくくりとして「ソフトウェア/ハードウェア」があります。前者はWindowsやAndroidなどの「アプリケーション」、後者はFPGAやマイコンなどを用いた「組み込み機器」上でデジタル信号処

理を行うものです。

● FPGAに実装かマイコンに実装か

さらにハードウェアで実現する中でも「FPGA/マイコン」という選択肢があります。FPGAの場合はHDL(Hardware Description Language)での開発が基本ですが、マイコンの場合はC言語が基本になります。達成物はどちらも組み込み機器であり、見た目は似ているのですが、何しろ言語が違うのでその開発スタイルもだいぶ異なります。

● C言語で開発かHDLで開発か

それでは「C言語/HDL」というくくりで考えてみましょう。マイコンはC言語で開発するので、Windowsアプリケーションと同じグループになります。組み込み機器とアプリケーションですから達成物的には全然違うのですが、ソース・コード類の相性はよさそうです。

● 浮動小数点演算か固定小数点演算か

Windowsアプリケーションの場合、加減乗除は浮動小数点で行うことが多いと思います。それに対し、FPGAやマイコンなどのハードウェアの場合は、固定小数点(整数)による演算が中心になります^{脚注}。

● HDLを自力で書くか高位合成ツールに生成させるか

C言語で書かれたアルゴリズムが巷にあふれているためか、近年はそれらをFPGAに実装したいというニーズも多いようです。その際はC言語をHDLに変換するための何らかの手段が必要になります。

その変換手段のくくりとして「手作業でHDL化/高位合成ツールでHDL化」があると思います。

脚注：FPU(Floating Point Unit)があれば高速な浮動小数点演算が可能だが、それを有するデバイスは少数派である。

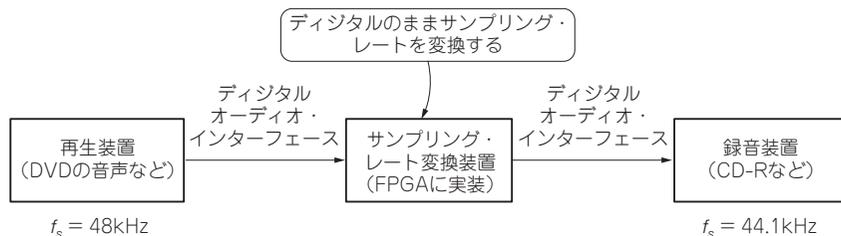


図1 サンプリング・レート変換器とは