

ハードウェアを用いたリアルタイム雑音除去処理システムの検討

山田彰巧[†] 松崎隆哲[‡]近畿大学大学院 産業技術研究科[†] 近畿大学 産業理工学部[‡]

1. はじめに

近年、デジタル信号処理技術の発展により、信号処理はアナログ信号処理からデジタル信号処理が主流となってきている。

デジタル信号処理の一種である雑音除去処理は、我々が日常的に使用する携帯電話やオーディオ機器に広く用いられており、高性能化による雑音除去処理能力の向上に伴って、演算時間が増加する問題が生じてきている。

雑音除去処理における演算時間の増加問題は演算を行うコンピュータの高性能化やソフトウェアのアルゴリズム見直し、ハードウェアを用いた演算の並列処理化が解決策として考えられる。しかし、それぞれに長所と短所が存在するため、一概にどの方式が雑音除去処理に最適であるかを定めることは困難である。

そこで本研究では、再設計や書き換えが容易である理由から、ハードウェアである FPGA ボードを用いて演算を並列処理化することにより、リアルタイムでの雑音除去処理が可能であるか検討を行っている。

本研究では、これまで音声の入力インターフェースを設計し音声処理を行う環境を構築した。本稿では、雑音除去処理アルゴリズムを用いた雑音除去処理のために、時間領域信号を周波数領域信号へと変換を行う回路の設計について述べる。

2. 一般的な雑音除去処理

一般的な雑音除去処理では、まず入力されたアナログ信号をコンピュータ上で扱えるデジタル信号へ変換を行うために、ソフトウェア上で定めたサンプリングレートと量子化ビット数に従い標本化し、量子化の後に符号化する。

そして、符号化した信号に ICA (Independent Component Analysis) [1] などの雑音除去処理アルゴリズムを用いて元の入力信号から雑音成分を取り除き、復号化によってアナログ信号として出力する。

ソフトウェアを用いた雑音除去処理方式では、入力された音声データを入力順に処理する逐次処理方式が用いられている。そのため、ICA などの雑音除去処理アルゴリズムを用いて雑音除去を行う場合、同じ演算を幾度も繰り返すことから、入力データ数に比例して処理時間は増加する。したがって、ソフトウェアを用いた方式では、入力データ数が膨大であるほど雑音除去のリアルタイム処理は困難である。

3. FPGA ボードを用いた雑音除去処理

本研究では、FPGA ボード上に同じ演算を行う回路を複数実装し、繰り返し演算処理の並列化を行うことで、リアルタイムでの雑音除去処理システムが実現可能であるか検討を行っている。音声の入出力には、入力インターフェースをラインイン、出力インターフェースをラインアウトとし、入力装置にマイク、出力装置にスピーカーを用いる。

FPGA ボード上での雑音除去処理のおおまかな流れを図 1 に示し、以下に詳細を記す。

1. マイクからのアナログ入力信号をラインインに入力
2. 取り入れたアナログ信号をデジタル信号に変換
3. 変換された信号の一部を切り出し FPGA に搭載されているメモリに保存
4. FPGA 上に実装した ICA などの雑音除去アルゴリズムを用いて切り出した信号の雑音成分を除去処理
5. 雑音除去処理されたデジタル信号をアナログ信号に変換
6. アナログ信号をラインアウトからスピーカーへ出力

雑音除去処理システムを評価するためには、雑音除去処理の前後データの比較が行えなければならない。雑音除去処理前後のデータを比較するために、雑音除去処理前のデータを保存し、別途出力できる機構の構築が必要である。

本研究の雑音除去処理システムではリアルタイム性を重視している。演算処理を可能な限り高速化することで演算時間の短縮を図るため、雑音除去処理中の演算について並列処理化可能となる箇所についての検討が求められる。

Research of Real-time Noise reduction system using Hardware.

[†]Akiyoshi Yamada, Graduate school of advanced technology, Kinki University.

[‡]Takanori Matsuaki, Faculty of Humanity-Oriented Science and Engineering, Kinki University.

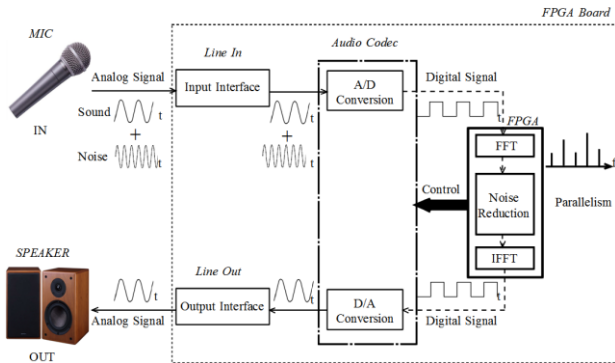


図 1: FPGA ボードを用いた雑音除去処理の流れ

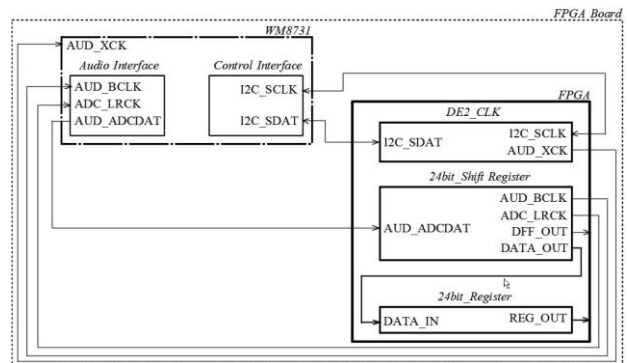


図 2: 多ビット音声データ取り扱いの構成回路

4. I²C を用いた音声インターフェースの設定

ハードウェアである FPGA は専用のライブラリやドライバは用意されていないため、音声を FPGA ボード上で取り扱うために音声インターフェースの設定を行わなければならない。

FPGA ボード上には A/D 変換や D/A 変換を行えるオーディオコーデックである WM8731 が搭載されている。入出力インターフェースは、I²C を用いて FPGA と WM8731 の通信を行い、FPGA から WM8731 を制御する回路を設計することで完成した。I²C を用いた音声インターフェースの設定により、FPGA ボード上で音声処理を行う環境が整った。

5. FFT 設計についての検討

本研究で用いる雑音除去処理アルゴリズムは、周波数領域データを利用するが、A/D 変換後の信号は時間領域信号であるため周波数領域信号へ変換を行う必要がある。そこで、本研究ではいくつかある周波数領域信号への変換方法の中から、高速フーリエ変換と呼ばれる FFT (Fast Fourier Transform) [2] を用いることにした。

FFT の設計にあたり、FFT アルゴリズム内部の演算について調査を行った。FFT アルゴリズムにおける演算はバタフライ演算が用いられており、入力信号にバタフライ演算を複数回行うことで周波数領域信号へと変換される。つまり、周波数領域への変換はバタフライ演算を行う回路を設計し、設計した回路を複数用意した後、FPGA 上へ実装を行うことで実現することができる。

FFT を用いて周波数領域へ変換を行う入力音声データは A/D 変換により符号化されたビット列であり、複数のビットをまとめたものが 1 つの音声データとなる。そのため、FFT への入力データは多ビットとなる。多ビットの音声データを取り扱うために設計した回路を図 2 に示す。

1 つの音声信号は WM8731 による A/D 変換によって、24 ビットのデータ列で表わされる。その

ため、FFT への入力データは 24 ビットとなる。なお、FFT に入力を行うデータは 24 ビットの実部データと虚部データが必要であるが、音声信号に虚部は存在しないため、虚部データの入りは 0 となる。

6. おわりに

本研究では、ハードウェアである FPGA ボード上に雑音除去処理システムを構築し、演算の並列処理化による雑音除去処理の高速化によってリアルタイムでの雑音除去処理が可能であるか検討を行っている。

FPGA ボード上で音声処理を行う環境を構築するために、FPGA とオーディオコーデックである WM8731 を通信する回路の設計を行い、FPGA ボード上で音声信号の取り扱いを可能にした。次に、ICA などの雑音除去処理アルゴリズムを用いて雑音除去処理を行うために、時間領域信号を周波数領域信号へ変換を行う手法である FFT を設計する方法について検討を行った。

FPGA ボード上への雑音除去処理システムの構築は、FFT アルゴリズム内の演算であるバタフライ演算回路の設計と、FPGA 上に雑音除去処理アルゴリズムの実装を行い、それぞれの回路を統合することで可能となる。その後、雑音除去処理の前後データを比較し、雑音除去処理時間や、雑音除去の精度についての評価を行う。

今後、FFT の設計と雑音除去処理アルゴリズムの実装を行い、FPGA ボード上に雑音除去処理システムの構築を行う。

参考文献

- [1] C.Charoensak, F.Satter, "Design of Low-Cost FPGA Hardware for Real-time ICA-Based Blind Source Separation Algorithm," EURASIP Journal on Applied signal Processing, pp.3076-3086, 2005.
- [2] Robert W.Stewart, Daniel.G.Alis "Concise DSP Tutorial," 2001.