ASICデザインコンテスト
規定課題
「MP3デコーダのIMDCT処理のハードウェア化」

パルテノン研究会

平成13年12月11日 Version 1.0

目 次

1 はじめに **1**

2 課題の概要 **1**

3 課題の詳細 **2**

3.1 IMDCT 処理 2

3.2 設計仕様 4

3.3 設計環境 5

4 評価項目 **5**

5 提出物について **6**

6 HW/SW 協調シミュレーションについて **6**

7 付録 (参照ソフト、WEB サイト、書籍など) **6**

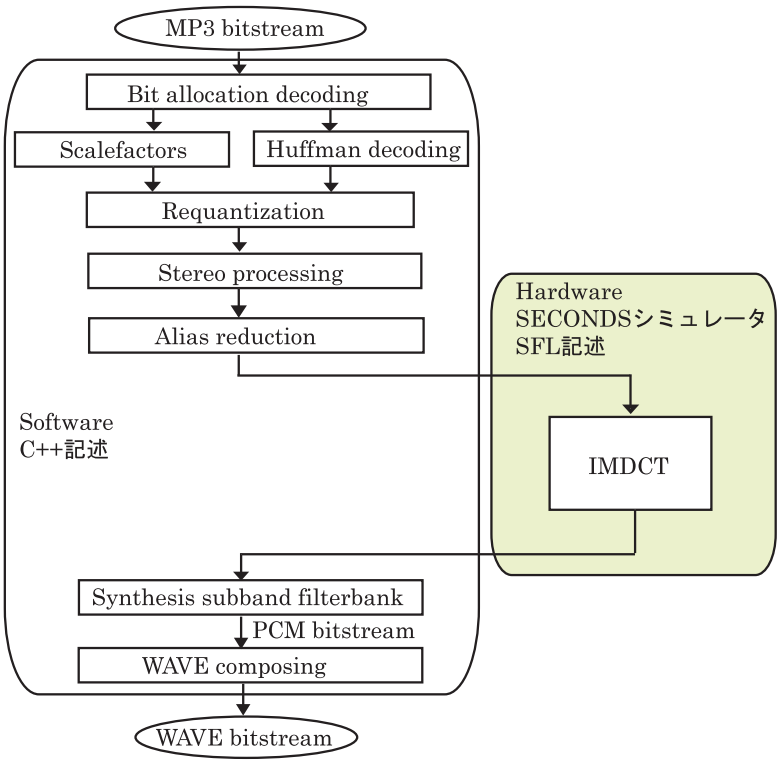


図 1: MP3 デコーダ HW/SW 協調シミュレーション

1 はじめに

本稿はパルテノン研究会が主催する「ASIC デザインコンテスト」の規定課題である MP3 デコーダの IMDCT (Inverse Modified Discrete Cosine Transform) 処理部分のハードウェア設計仕様を定めるものです。MP3 とは国際音声伸張規格である MP3 (MPEG-1 Audio Layer3) の略称です。本稿では MP3 デコーダの詳細に関しては述べません。MP3 規格の詳細は「付録」ページにある WEB サイトや書籍リストを参照してください。

2 課題の概要

MP3 デコーダのブロック構成を図 1 に表します。本課題では、MP3 デコーダの IMDCT 処理回路の設計を行ないます。設計した IMDCT 処理回路に対して、HW/SW 協調シミュレーションにより実際の MP3 フォーマットのビットストリームをデコードし、音質などの検証を行ないます。

ソフトウェアによる処理の解析では IMDCT 処理は MP3 デコード全体の約 50%を占めています [2]。その処理を専用ハードウェアで実行することで CPU の負荷を軽減することができます。IMDCT は積和演算中心の処理であり、演算器、演算順序、データパス、係数テーブルの取り扱い、メモリアクセス、などについての工夫が必要です。また、データのビット幅や演算精度の音質への影響を考慮する必要があります。

本課題の設計では、標本化周波数 48kHz、 16bit 量子化、ステレオ音声のリアルタイム処理を目標とします。これは、Digital Audio Tape や Digital Video の音質に相当するものです。ライブラリ

は NEC CMOS9 (0.35μm、2 層、CMOS Gate Array) を用い、PARTHENON システムでの論理合成により、動作周波数 (処理レート) 、回路規模、消費電力を評価します。また、使用するメモリ (係数のテーブル、入出力ならびに中間バッファ) の規模も評価します。

設計した IMDCT 処理回路に対して、課題用に提供される MP3 デコーダ HW/SW 協調シミュレーション環境を用いて検証を行ないます。この環境では、MP3 デコーダソフトウェアの一部を seconds による回路シミュレーションに置き換えて実行することができます。

MP3 の規格では、音質 (精度) に関して規格に準拠しているか否かのテスト (compliance test) が規定されています。上述のシミュレーション環境を用いて、実際の MP3 のビットストリームをデコードし出力される音声を確認するとともに、compliance test を行ないます。compliance test では、評価用のサンプル音声データならびに評価式が規定され、結果は “fully compliance”, “limited accuracy”, “not compliance” の 3 段階で与えられます。本課題の設計では、“limited accuracy” を満たすことを目標とします。

3 課題の詳細

3.1 IMDCT 処理

MP3 デコード処理の各ブロックでは、576 サンプルデータ分からなるグラニュール (granule) を単位として処理を行います。576 サンプルデータは周波数領域に変換されており、32 個の等幅サブバンド (サブバンド当り 18 サンプル) に分割されています。図 2 に入力データの構成を表します。各サブバンドは、エンコードの際にロングブロックあるいはショートブロックに分類され、それぞれに対応する変換処理が行われています。ロングブロックは、さらにスタートブロック、ノーマルブロック、ストップブロックに分類されます。ショートブロックの場合、サブバンドはさらに 3 つの窓関数によって処理されます。各グラニュールにおけるブロックの種類は MP3 ビットストリームのヘッダに格納され、デコードの際に、各グラニュールはそれに従って処理されます。

IMDCT 処理は IDCT 処理部と逆窓関数処理部からなります。

一つのサブバンド sb に対する IDCT の処理は次の計算式で与えられます。ロングブロックの場合、入力を X_k^{sb}、出力を x_i^{sb}、$n = 36$ として

$$x_i^{sb} = \sum_{k=0}^{\frac{n}{2}-1} X_k^{sb} \cos\left(\frac{\pi}{2n}\left(2i+1+\frac{n}{2}\right)(2k+1)\right) \qquad (i = 0, \ldots, n-1)$$

となります。ショートブロックの場合、入力を $X_k^{sb,j}$、出力を $x_i^{sb,j}$、$n = 12$ として

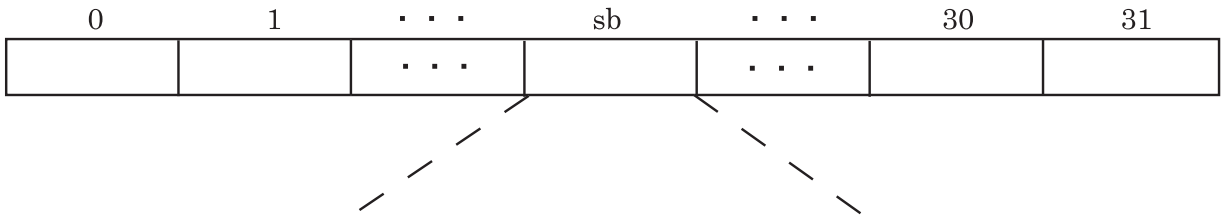
$$x_i^{sb,j} = \sum_{k=0}^{\frac{n}{2}-1} X_k^{sb,j} \cos\left(\frac{\pi}{2n}\left(2i+1+\frac{n}{2}\right)(2k+1)\right) \quad (i = 0, \ldots, n-1,\ j = 0, 1, 2)$$

となります。

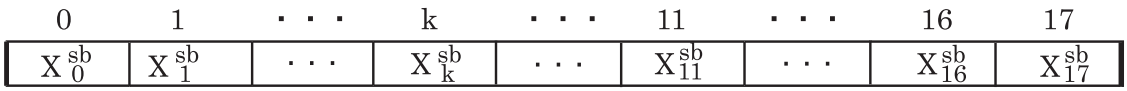
一つのサブバンド sb に対する逆窓関数の処理は、次の計算式で与えられます。ロングブロックの場合、入力を x_i^{sb}、出力を z_i^{sb} とし、ノーマルブロックの場合

$$z_i^{sb} = x_i^{sb} \sin\left(\frac{\pi}{36}\left(i+\frac{1}{2}\right)\right) \qquad (i = 0, \ldots, 35)$$

1 グラニュールデータは32サブバンドデータからなる。(サブバンドを表すン添字をsbとする。)



ロングブロックの場口合(サンプルを表すン添字をkとする。)



ショートブロックの場合。サブバンドはさらに３つの窓関数に分けて処理される。
(窓関数の部分を表す添字を j として、それぞれの中でサンプルを表すン添字をkとする。)

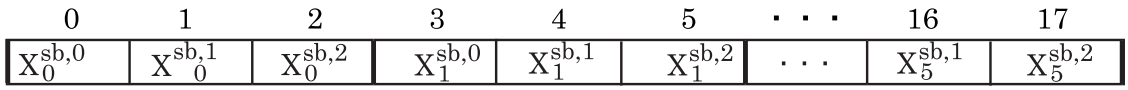


図 2: IMDCT への 1 グラニュール分の入力データの構成

ロングブロックのスタートブロックの場合

$$z_i^{sb} = \begin{cases} x_i^{sb} \sin\left(\frac{\pi}{36}\left(i+\frac{1}{2}\right)\right) & (i = 0, \ldots, 17) \\ x_i^{sb} & (i = 18, \ldots, 23) \\ x_i^{sb} \sin\left(\frac{\pi}{12}\left(i-18+\frac{1}{2}\right)\right) & (i = 24, \ldots, 29) \\ 0 & (i = 30, \ldots, 35) \end{cases}$$

ロングブロックのストップブロックの場合

$$z_i^{sb} = \begin{cases} 0 & (i = 0, \ldots, 5) \\ x_i^{sb} \sin\left(\frac{\pi}{12}\left(i-6+\frac{1}{2}\right)\right) & (i = 6, \ldots, 11) \\ x_i^{sb} & (i = 12, \ldots, 17) \\ x_i^{sb} \sin\left(\frac{\pi}{36}\left(i+\frac{1}{2}\right)\right) & (i = 18, \ldots, 35) \end{cases}$$

となります。ショートブロックの場合、入力を $x_i^{sb,j}$、出力を z_i^{sb} とし、

$$y_i^{sb,j} = x_i^{sb,j} \sin\left(\frac{\pi}{12}\left(i+\frac{1}{2}\right)\right) \quad (i = 0, \ldots, 11,\ \ j = 0, 1, 2)$$

として、

$$z_i^{sb} = \begin{cases} 0 & (i = 0, \ldots, 5) \\ y_{i-6}^{sb,0} & (i = 6, \ldots, 11) \\ y_{i-6}^{sb,0} + y_{i-12}^{sb,1} & (i = 12, \ldots, 17) \\ y_{i-12}^{sb,1} + y_{i-18}^{sb,2} & (i = 18, \ldots, 23) \\ y_{i-18}^{sb,2} & (i = 24, \ldots, 29) \\ 0 & (i = 30, \ldots, 35) \end{cases}$$

となります。

最後に、各サブバンド前半のデータと、直前のグラニュールの対応するサブバンド後半のデータの和をとり、IMDCT の出力が計算されます。$z_i^{'sb}$ を直前のグラニュールの sb 番目のサブバンドの i 番目のデータとし、$result_i^{sb}$ を IMDCT 処理の最後出力として、

$$result_i^{sb} = \begin{cases} z_{i+18}^{'sb} + z_i^{sb} & (i = 0, \ldots, 17, \quad i \text{ または } sb \text{ が偶数の場合}) \\ -(z_{i+18}^{'sb} + z_i^{sb}) & (i = 0, \ldots, 17, \quad i, sb \text{ が共に奇数の場合}) \end{cases}$$

となります。また、次のグラニュールの処理のために z_{18}^{sb} ~ z_{35}^{sb} を保存しておく必要があります。なお、この値は最初のグラニュールでは 0 に初期化されています。

3.2 設計仕様

- Alias reduction 処理からの出力 (c/c++の倍精度実数に相当) を入力とし、IMDCT 処理を行ない、Subband synthesis filterbank 処理への入力を出力します。
- 入出力はグラニュール (576 サンプル分) 単位でメモリバッファに格納されます。
- 入出力バッファの構成 (ダブルバッファ化してもよい)、データ構造 (固定小数点/浮動小数点、ビット割当て) バス幅などは自由に設計してください。
- 処理実行中の状態を表す`imdct_busy` 信号を出力 (`instrout`) します。アイドル状態 (`imdct_busy=0`) のとき、外部からの `imdct_start` 信号 (`instrin`) により処理を開始 (`imdct_busy=1`) し、処理が終了したらアイドル状態 (`imdct_busy=0`) に戻ります。

- ブロックタイプは`block_type`信号 (2bit) で次のとおりに与えられます。
  00 ロングブロックのノーマルブロック
  01 ロングブロックのスタートブロック
  10 ショートブロック
  11 ロングブロックのストップブロック
- 回路記述には SFL を用います。
- 合成ライブラリは NEC CMOS9 (0.35 μ m、2 層、CMOS Gate Array) を用います。
- 標本化周波数 48kHz、16bit 量子化、ステレオ音声のリアルタイムデコードが可能であること。処理レートは、1 グラニュール分の処理に要するクロックサイクル数と最大動作周波数から計算します。その際、入出力バッファへのアクセスに要するサイクル数も考慮する必要があります。(例えば、Alias reduction からの入力バッファのデータ書き込みの間 IMDCT 処理が停止してしまうような設計では、そのサイクル数も見積もって加えます。)
- 音質 (精度) は compliance test の limited accuracy を満たすこと。

3.3 設計環境

本コンテスト課題では以下の設計環境で行うことにします。

- OS:
  Debian GNU/Linux, Version 2.2.17 以降、
  Red Hat Linux, Version 6.1 以降。
- 論理合成システム:
  PARTHENON, Version 2.4.1 以降。
- C++ compiler:
  GNU C++, Version 2.95.2 (Linux 版) 以降。
- CPU:
  Intel 系 CPU。

4 評価項目

- 処理レート (動作周波数、処理に要するクロックサイクル数)、回路規模、消費電力、使用するメモリ (係数のテーブル、入出力ならびに中間バッファ) の量。
- 音質 (精度)
- アーキテクチャの独創性
- ドキュメンテーション

5 提出物について

- ドキュメント
  - 設計した回路のアーキテクチャの解説。
  - 処理レート (動作周波数、処理に要するクロックサイクル数) 、回路規模、消費電力、使用するメモリ (係数のテーブル、入出力ならびに中間バッファ) の量。
  - compliance test の結果。
  - ソースコード類の解説、シミュレーション手順、合成手順、シミュレーション環境コンパイル手順など。(入出力バッファの仕様と HW/SW 協調シミュレーション用、データ受け渡し関数の解説を含む)
  - 応募者の使用した OS, PARTHENON, C++ コンパイラなどの処理系の名称とバージョン。
- 設計した回路の全ての SFL 記述及びシミュレーションならびに合成スクリプトなど。
- 応募者が作成した全ての C++ソースコード、スクリプトなど。

6 HW/SW 協調シミュレーションについて

本課題に応募する方のための HW/SW 協調シミュレーション環境パッケージは追って Web Site 上に公開されます。

7 付録 (参照ソフト、WEB サイト、書籍など)

参考文献

[1] パルテノン研究会 ASIC デザインコンテスト WEB サイト
http://www.kecl.ntt.co.jp/parthenon/html/contest.htm

[2] 福島徹也、李星日、境和久、泉知論、尾上孝雄、中村行宏: PARTHENON を用いた MP3 デコーダの HW/SW 設計, 第 18 回パルテノン研究会資料集, PP.7-13,(2001).

[3] ISO の MP3 デコーダサンプルソースコード (C 言語) の配布サイト:
ftp://ftp.tnt.uni-hannover.de/pub/MPEG/audio/mpeg2/software/technical_report/dist10.tar.gz

[4] 「MP3 詳細マニュアル」
http://pub.maruzen.co.jp/book_magazine/mp3/index.html

[5] http://www.cselt.it/mpeg/standards/mpeg-1/mpeg-1.htm
ISO/IEC 11172-3: 1993 Information technology
– Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit /s
– Part 3: Audio

[6] Moving Picture Experts Group : MPEG Home page
http://www.cselt.it/mpeg/

[7] International Standards Organization : ISO Web Page
http://www.iso.ch/

[8] Heiko Purnhagen : MPEG Audio Web Page
http://www.tnt.uni-hannover.de/project/mpeg/audio/

[9] Robert Leslie : MPEG Audio Decoder Compliance,
http://www.mars.org/home/rob/proj/mpeg/compliance/