

地上波デジタル放送受信機の音声処理

1C-5

白須賀 恵一 竹村 貴子 和田 哲朗
 大川 雄敬 菅 隆志 厚井 裕司
 三菱電機株式会社 情報技術総合研究所

1. はじめに

AAC(Advanced Audio Coding)方式は 1997 年 4 月に ISO/IEC13818-7 として国際標準になったオーディオ符号化方式である。本方式は、2000 年本放送開始予定の BS デジタル放送やその後の地上波デジタル放送のオーディオ符号化方式として採用される予定である。我々はこれに先立ち地上波デジタル実験放送のための受信機を開発した。この受信機における AAC オーディオのデコード処理について報告する。

2. オーディオデコード仕様

開発した受信機のオーディオデコード仕様を表 1 に示す。

表 1 オーディオデコード仕様

項目	仕様
対応符号化方式	MPEG-2 AAC MPEG-1 Layer1,2
プロファイル	LC
チャンネルモード	ステレオ
ビットレート	144kbps/2ch
パケット長	384byte 固定長
量子化ビット	24bit
サンプリング周波数	48kHz
出力レベル	250mVrms±3dB (FS-18dB 入力時)

3. AAC オーディオデコード制約条件

AAC オーディオデコードにおいて次の 2 つの制約条件がある。

- (1) デコード遅延時間は、デマルチプレクサ(TS Demux)でのバッファアンダーフローをさけるためなるべく小さくする。
- (2) デコード遅延時間のゆらぎにより AV 同期の

ずれが判別されないように 20ms 以内にする。

4. 受信機全体構成

受信機は図 1 のように OFDM 復調部とビデオデコーダ部 (Main BOX)、AAC オーディオデコーダ部(AAC BOX)から構成される。AAC BOX の AAC デコーダ部は DSP により構成した。AAC BOX へのストリームは ES(Elementary Stream) 渡しとした。これは PES(Packetized Elementary Stream)パケット入力とした場合に BOX 間で必要な STC(System Time Clock)伝送用配線数の増大を避けるためである。アンテナからの OFDM 変調 RF 信号は OFDM 復調部で復調され、MPEG-2 TS(Transport Stream)信号として Main BOX 内の TS Demux に入力される。TS Demux で TS 信号からビデオ、オーディオとその他の情報に切り分けビデオ、オーディオデコーダ部それぞれにおいてデコードしモニタとス

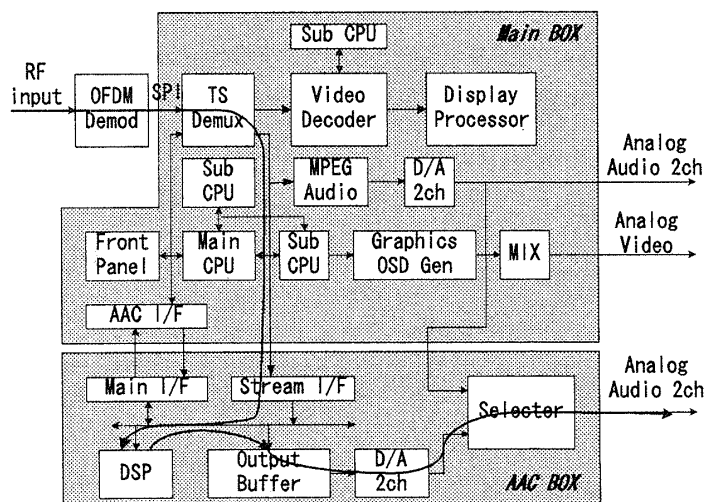


図 1 受信機全体構成

ピーカに出力する。AAC ES データは AAC BOX に入力され Stream I/F 部を介して DSP に転送されデコードされる。デコードされた PCM データは Output buffer 部に転送される。この Output Buffer は DSP の PCM データ出力レートと PCM レートのレート差を吸収する。PCM 出力バッファの PCM データは、D/A 変換され 2 チャンネルステレオ出力が得られる。

地上波デジタル放送ではオーディオ符号化方式は AAC であるが、本受信機は MPEG-1 Layer1,2 符号化方式にも対応している。MPEG-1 ES データの場合は Main BOX 内でデコード、D/A 変換される。そのアナログステレオ 2ch 出力が AAC BOX 内のセクタに入力される。セクタでは AAC か MPEG-1 オーディオか選択され出力される。

5. DSP デコード処理動作

制約条件(1)のデコード低遅延を満足するため DSP がデコードを開始するまでに DSP 管理の DSP buffer に蓄積する ES データ量を 2 フレーム分に抑えた。(ここで AAC 符号化 1 フレームとは入力 PCM データ 1024 サンプル分を表す。1 フレーム時間は、48kHz サンプリグでは 21.3ms 時間に相当する。) また制約条件(2)のデコード遅延時間のゆらぎを抑えるために 1/4 frame plus 信号を発生させ DSP のデコードを 1/4 フレーム周期を基準として動作させた。

DSP デコード処理の様子を図 2 に示す。図 2 において各段の意味を以下に示す。

- ・"1/4 frame plus" ; 1 フレーム時間である 21.3ms の 1/4 周期のパルス
- ・"ES data to DSP" ; Stream I/F から DSP 管理の DSP buffer に ES データが転送される様子
- ・"Decode" ; DSP が AAC ES データをデコードする様子
- ・"PCM(DSP=>Output buf)" ; DSP から Output buffer へ PCM データを出力する様子
- ・"PCM(Output buf=>DAC)" ; Output buffer から DAC へ PCM データが転送される様子

ES データはまず TS Demux から AAC BOX に入力され Stream I/F を介して DSP に転送される。DSP は Stream I/F からの割り込み毎に ES データをリードし、DSP buffer にデータを蓄積していく。図 2 では"ES data to DSP buffer" の"ES frame1"が最初に DSP buffer に入力される ES データ 1 フレーム分を表している。DSP buffer に ES データ 2 フレーム分が蓄積された後、次の 1/4 frame plus 割り込みまで待ち、この割り込みをトリガとして 1 フレーム期間の計測を DSP が開始するとともにデコード処理を開始する。1 フレーム期間は 1/4 frame plus を 4 回カウントして 1 フレーム期間とするので 1/4 frame plus が 4 回発生する毎に 1 フレーム期間

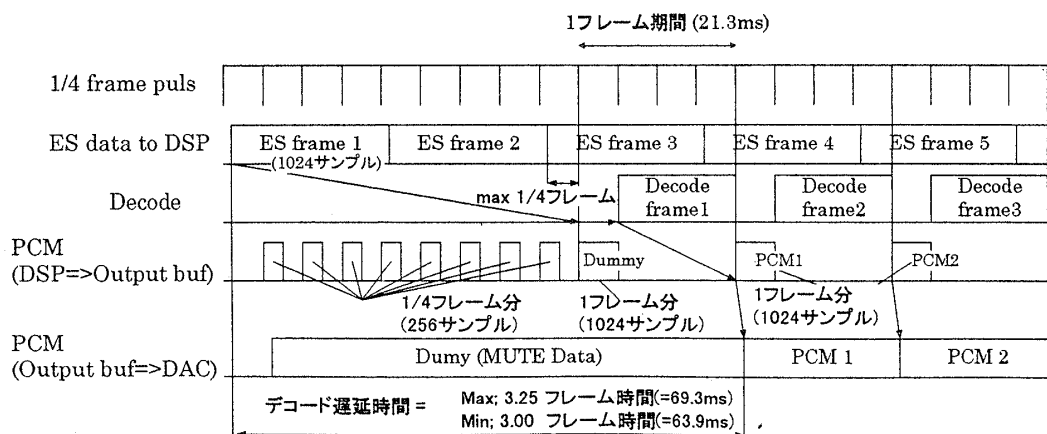


図 2 DSP デコード処理

が始まる。この後 DSP は1フレーム期間を単位として1パケットづつデコード処理をする。

1フレーム期間内の DSP の動作について説明する。1フレーム開始タイミングが来ると、まず前フレーム期間にデコードした1フレーム分の PCM データを Output buffer に出力する。ただし一番はじめの PCM 出力データについてはダミーデータ（ミュートデータ）を1フレーム分(1024 サンプル分)出力する。(図 2 の 1 フレーム分の Dummy)。この出力の後 DSP バッファにある ES データ 1 フレーム分をデコードする。このデータは次のフレームの開始タイミングの直後に出力される。

6. デコードディレイ時間見積もり

DSP のデコード処理時間は、図 2 より次のようになる。

- (1) DSP buffer への蓄積時間
; 2 フレーム時間
- (2) 1/4 frame plus が来るまでの待ち時間
; 1/4 フレーム時間(Max.)
0 フレーム時間 (Min.)
- (3) デコードから Output buffer への PCM 出力までの時間
; 1 フレーム時間

Output buffer から DAC、Analog 出力遅延時間は、他のブロックの遅延時間と比較して十分無視できるのでデコード遅延の見積もり時間は、下式となる。

$$\begin{aligned} \text{最大デコード遅延時間} &= 3.25 \text{ (フレーム)} \\ &= 3.25 * 21.3(\text{ms}) = 69.3(\text{ms}) \text{ (max)} \end{aligned}$$

$$\begin{aligned} \text{最小デコード遅延時間} &= 3.0 \text{ (フレーム)} \\ &= 3 * 21.3(\text{ms}) = 64.0(\text{ms}) \text{ (min)} \end{aligned}$$

ES データ入力タイミングと 1/4frame Plus のタイミングが非同期であり、(2)の項に最大 1/4 フレーム時間(5ms)のゆらぎが存在する。しかし 5ms 程度の AV 同期のずれは主観的に判別できず問題ないと考ええる。

7. デコード遅延時間評価

表 2 にデコード遅延時間計測結果を示す。

表 2 デコード遅延時間計測結果

	Decode delay(ms)
1	69.42
2	69.38
3	67.74
4	67.62
5	68.50
6	65.62
7	68.70
8	64.78
9	64.62
10	67.18
Average	67.36

デコード遅延時間は、64.78ms から 69.42ms の範囲に分布しており、制約条件(1),(2)を満たすほぼ理論通りの結果を得た。

8. AV 同期

上記のように AAC デコード遅延時間は固定ビットレートではほぼ一定遅延時間となる。TS Demux が出力する時刻は、PTS の示す時刻から既知のオーディオデコード遅延時間分(69.3ms)だけ差し引いた時刻にすればよい。

9. 今後の課題

今回開発した受信機の音声処理は、144kbps 固定長パケットの ES に対応している。今後は可変レート、可変長パケットに対応することが課題である。