

cokatielシステム

SmartAudio16

サンプルデータガイド
シューティングゲーム版

目次

1.はじめに	3
2.サンプルデータについて.....	4
3.サンプルデータを用いたトレーニング	6
3.1.準備	6
3.2.SmartAudio16の初期化	6
3.3.グループのロード	6
3.4.シーケンスの演奏	7
3.5.グループ入れ替えの実践	7
4.補足	9
4.1.グループ入れ替えを想定したメモリ設計	9
4.2.グループロードによるサウンドヒープの変化	9
4.3.グループ間でのデータ重複時の挙動	9
改訂履歴	10

1.はじめに

この文書では、『cockatielシステム SmartAudio16(以下SmartAudio16)』サンプルデータの詳細と、そのサンプルデータを用いて、実際にSmartAudio16を使用する流れを説明します。

2. サンプルデータについて

本STG(シューティング)パッケージに含まれるサンプルデータの構成を記載します。

ロードグループ名	コールラベル名	イメージ	サイズ	有
GROUP_BGM01	BGM01	ステージBGM	2,213,084	
GROUP_BGM02	BGM02	ステージBGM	1,612,518	●
GROUP_BGM03	BGM03	ステージBGM	1,374,723	●
GROUP_BGM04	BGM04	ステージBGM	2,865,881	●
GROUP_BGM05	BGM05	ステージBGM	2,928,417	●
GROUP_BGMJIN	BGMJIN	ジングル	3,021,000	
GROUP_BGMOP	BGMOP	オープニングBGM	3,577,815	●
GROUP_SE_SYS	SE_SYS01_SELECT	セレクト移動音	612,648	
	SE_SYS02_DECIED1	決定音1		
	SE_SYS03_CANSEL	キャンセル音		
	SE_SYS04_WINDOW1	メッセージウィンドウ開く音1		
	SE_SYS05_WINDOW2	メッセージウィンドウ開く音2		●
	SE_SYS06_DECIED2	決定音2		●
	SE_SYS07_PAUSE	ポーズ音		●
	SE_SYS08_INFOMATION	インフォメーション		●
	SE_SYS09_NG	NG(NoGood)		●
	SE_SYS10_COUNT	カウント音		●
	SE_SYS11_DECIED3	決定音3		●
	SE_SYS12_SPECIAL	スペシャル音		●
GROUP_SE_PLY	SE_PLY01_SHOT1	ショット音1	612,648	
	SE_PLY02_SHOT2	ショット音2		
	SE_PLY03_SHOT3	ショット音3		
	SE_PLY04_SHOT4	ショット音4		
	SE_PLY05_SHOT5	ショット音5		●
	SE_PLY06_POWERUP	パワーアップ		●
	SE_PLY07_POWERDOWN	パワーダウン		●
	SE_PLY08_ITEMGET	アイテム取得		●
	SE_PLY09_SHIELD	シールド		●
	SE_PLY10_DOWN1	ダウン1		●
	SE_PLY11_DOWN2	ダウン2		●
	SE_PLY12_SHOT6	ショット音5		●

SmartAudio16: サンプルガイド

ロードグループ名	コールラベル名	イメージ	サイズ	有
GROUP_SE_ENM	SE_ENM01_BREAK1	敵破壊1	614,350	
	SE_ENM02_BREAK2	敵破壊2		
	SE_ENM03_BREAK3	敵破壊3		
	SE_ENM04_BREAK4	敵破壊4		
	SE_ENM05_BEAM1	ビーム1		
	SE_ENM06_BEAM2	ビーム2		
	SE_ENM07_BEAM3	ビーム3		●
	SE_ENM08_BEAM4	ビーム4		●
	SE_ENM09_BOM1	爆発1		●
	SE_ENM10_BOM2	爆発2		●
	SE_ENM11_BOM3	爆発3		●
	SE_ENM12_BOM4	爆発4		●
	SE_ENM13_LIGHT1	光音1		●
	SE_ENM14_SPARK1	スパーク1		●
	SE_ENM15_SPARK2	スパーク2		●
	SE_ENM16_LASER1	レーザー1		●
	SE_ENM17_FIRE	炎		●
	SE_ENM18_LASER2	レーザー2		●
	SE_ENM19_BOM5	爆発5		●
	SE_ENM20_SHIELD	シールド		●
	SE_ENM21_LASER3	レーザー3		●
	SE_ENM22_BOM5	爆発5		●
GROUP_SE_ACT	SE_ACT01	動作アクション1	612,652	
	SE_ACT02	動作アクション2		●
	SE_ACT03	動作アクション3		●
	SE_ACT04	動作アクション4		●
GROUP_BGMJINGAMEOVER	BGM_JINGAMEOVER	ゲームオーバージングル	847,862	●

※『有』欄に●印のついているものは、有料版のみに含まれるサウンドです。

※『サイズ』欄は、有料版におけるグループサイズで、単位はバイト(byte)です。含まれるサウンド数が少ない無料サンプル版では、表示よりデータサイズが少なくなる場合があります。

※一部グループはオーディオデータが重複しており、表記されているグループサイズの合計値より実際にメモリにロードされるサイズが小さくなる場合があります。

理由と詳細については、“4.3.グループ間でのデータ重複時の挙動”の項をお読みください。

3. サンプルデータを用いたトレーニング

サンプルデータを使用して、グループのロードや入れ替え、そしてグループに含まれるシーケンスデータの演奏を行ってみましょう。

3.1. 準備

“SmartAudio16: プログラマーズガイド”ドキュメント内の“2. 準備”の項を参考に、SmartAudio16ライブラリの登録を行います。

また、サウンドパッケージファイル(**.spkg)および、サンプルデータに含まれるグループ番号とラベル番号がシンボル化されたヘッダファイル(**.h)もプロジェクトに登録しておきます。

3.2. SmartAudio16の初期化

初期化用メソッド『cktlInit』を用いて、SmartAudio16システムの初期化を行います。

『cktlInit』メソッドの詳細は、“SmartAudio16: プログラマーズガイド”ドキュメント内の“3. SmartAudio16の初期化”の項をお読みください。

引数『heapSize』に指定するサウンドヒープサイズですが、BGMグループのサイズ + SEグループのサイズ + 常駐するSmartAudio16システム情報のサイズを最低限確保する必要があります。後述するリソースパスの長さやシーケンスプレイヤーの数によってシステム情報のサイズは変動するので、少し余裕を持った値に設定するのが良いでしょう。

メモリサイズに関する詳細は、“SmartAudio16: プログラマーズガイド”ドキュメント内の“3.1. 初期化時メモリサイズの目安”の項をお読みください。

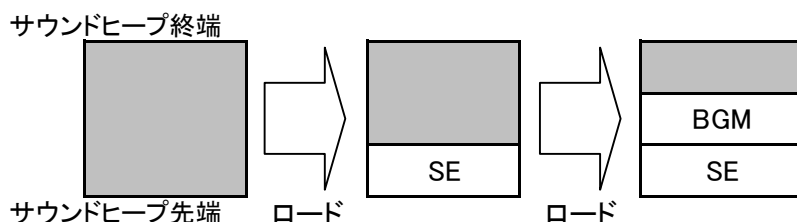
引数『filePath』には、サウンドパッケージファイルのパスを指定します。リソースパス取得の方法については、各開発環境のマニュアルを参照してください。

3.3. グループのロード

SmartAudio16の初期化に成功したら、『cktlLoadGroup』メソッドを用いてグループをロードしましょう。

『cktlLoadGroup』メソッドの詳細は、“SmartAudio16: プログラマーズガイド”ドキュメント内の“4. 演奏に必要なデータのロード”の項をお読みください。

今回は、まずSEのグループをひとつロードし、次にBGMのグループもひとつロードしておきましょう。



なおロードの際には、ヘッダファイルに記述されたシンボル名を使用することができます。(ただし、Javaの場合はstatic変数などに書き換えておく必要があります)

3.4.シーケンスの演奏

グループのロードに成功したら、グループに含まれるシーケンスデータの演奏をしてみましょう。

シーケンスデータにはそれぞれラベル番号が設定されており、演奏開始用のメソッドでは演奏したいシーケンスデータに対応するラベル番号を指定します。

たとえばラベル番号0番のBGM演奏し、さらにラベル番号2番のSEを鳴らす場合は以下のような記述になります。

```
[cktISys cktIStartSequence:0 volume:127 pan:0];
[cktISys cktIStartSequence:2 volume:127 pan:0];
```

iOS

```
cktISys. cktIStartSequence(0, 127, 0);
cktISys. cktIStartSequence(2, 127, 0);
```

AndroidOS

※“cktISys”は、CktIDriverクラスのインスタンスです。

正しく演奏されることが確認できたら、ラベル番号を変更して他のサウンドも演奏してみましょう。

なおグループ番号と同様に、ヘッダファイルがインクルードされていればシンボルによるラベル番号の指定が可能です。

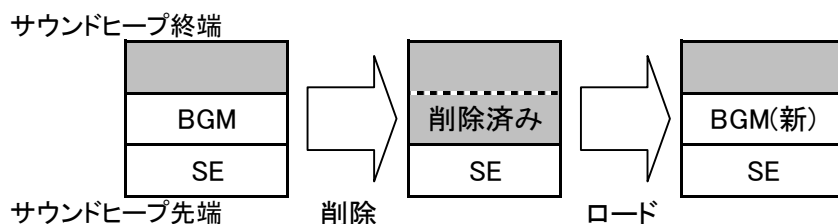
演奏開始用メソッドの詳細および演奏の種類については、“SmartAudio16:プログラマーズガイド”ドキュメント内の“5.シーケンスの演奏”の項をお読みください。

3.5.グループ入れ替えの実践

サンプルデータでは、BGMはひとつのグループに一曲だけ設定されています。そこで、BGMのグループを入れ替えて、別のBGMを演奏してみましょう。

まず、BGMを『cktIStopSequence』メソッドで停止した後、BGMグループをサウンドヒープから削除します。グループ削除用メソッドの詳細および注意点は、“SmartAudio16:プログラマーズガイド”ドキュメント内の“7.グループの削除”の項をお読みください。

その後、新たなグループをロードすれば、グループの入れ替えは完了です。



SmartAudio16: サンプルガイド

入れ替えたBGMが正しく演奏されるか確かめてみましょう。

以上が、SmartAudio16の基本となる初期化からシーケンスデータ演奏までの流れです。

4.補足

4.1.グループ入れ替えを想定したメモリ設計

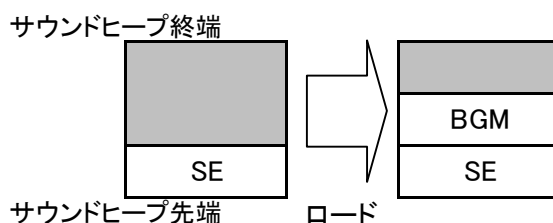
システム初期化時にはグループ入れ替えに最低限必要なメモリサイズを算出する必要があります。

たとえば『GROUP_SE_SYS』グループを常駐の効果音として、『GROUP_BGM01～03』の三つのグループを入れ替えのBGMとして運用する場合は、最低限必要なサウンドヒープサイズは“常駐グループサイズ (GROUP_SE_SYS) + 入れ替えグループでもっともサイズが大きいもの”と考えられます。

4.2.グループロードによるサウンドヒープの変化

サウンドヒープは、シンプルな積み上げ式のメモリ領域です。ロードされたグループデータは、層(レイヤー)としてサウンドヒープ内に積み上げられていきます。

たとえば『SE』グループを先に、『BGM』グループを後にロードした場合、サウンドヒープ内部は下図のような状態となります。



グループデータの削除はもっとも新しくロードされたグループに対してのみ有効です。したがって、上記のケースでは『BGM』グループを削除してからでなければ『SE』グループの削除を行うことができませんので、注意してください。

つまりSmartAudio16においては、『入れ替え頻度の高いものほど後にロードを行う』ことが重要となります。

4.3.グループ間でのデータ重複時の挙動

グループロードの際、すでにサウンドヒープ内に存在しているオーディオデータ・シーケンスデータが重複してロードされることはありません。

したがって、グループデータサイズより、実際にサウンドヒープにロードされるデータサイズは小さくなる場合があります。

改訂履歴

日付	内容
2014.04.16	サウンドパッケージデータ更新に伴い、『2. サンプルデータについて』リスト内容更新。
2014.04.08	『2. サンプルデータについて』リスト内容更新。
2014.04.02	各サンプル毎に文書を分割。
2014.03.27	SmartAudio16 バージョン1.4.1に沿った内容に改訂。
2013.02.07	新規ドキュメント作成。

※ 「iPhone」「iPad」「Objective-C」はApple Inc.の米国およびその他の国における登録商標です。

また「IOS」の商標は、Ciscoの米国およびその他の国のライセンスに基づき使用されています。

※ 「Android」はGoogle Inc.の商標または登録商標です。

※ 「Java」は、Oracle America, Inc. の商標です。

※ 「ARM」はARM 社の登録商標です。

※ その他記載されている会社名、製品名は各社の登録商標または商標です。