

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日  
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

# アプリケーション・ノート

## 78K0R/LH3

### サンプル・プログラム（音声出力）

### D/Aコンバータとオペアンプの機能を使用した音声データ再生編

この資料は、ADPCM形式の音声データの再生を12ビットD/Aコンバータより音声出力するアプリケーションノートです。また、ADPCMデータは32kbps（サンプリング周波数：8kHz、量子化ビット数：4bit）を使用しています。

#### 対象デバイス

μ PD78F1506 , 78F1507 , 78F1508

#### 目次

第1章	概要	...	3
第2章	回路イメージ	...	5
2.1	回路イメージ	...	5
2.2	マイコン以外の使用デバイス	...	6
2.3	端子機能一覧	...	6
第3章	ソフトウェアについて	...	7
3.1	ファイル構成	...	7
3.2	使用する内蔵周辺機能	...	8
3.3	初期設定と動作概要	...	9
3.4	フロー・チャート	...	10
第4章	設定方法について	...	13
4.1	周辺ハードウェア・マクロ使用の設定	...	13
4.2	タイマ・アレイ・ユニット0の設定	...	14
4.3	ボルテージ・リファレンスの設定	...	22
4.4	オペアンプの設定	...	24
4.5	D/Aコンバータの設定	...	28
4.6	ソフトウェア記述例	...	31
4.7	再生処理について	...	38
第5章	関連資料	...	45
付録A	プログラム・リスト	...	46
付録B	改版履歴	...	99

- ・本資料に記載されている内容は2009年09月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
  - ・文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
  - ・当社は、本資料に記載された当社製品の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、一切その責を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
  - ・本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責を負いません。
  - ・当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。また、当社製品は耐放射線設計については行っておりません。当社製品をお客様の機器にご使用の際には、当社製品の不具合の結果として、生命、身体および財産に対する損害や社会的損害を生じさせないように、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計を行ってください。
  - ・当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。  
「標準水準」：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット  
「特別水準」：輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器  
「特定水準」：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等  
当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。
- 注1. 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。
- 注2. 本事項において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいう。

(M8E0909J)

# 第1章 概 要

このサンプル・プログラムは、ADPCM形式の音声データを12bit D/Aコンバータにて再生出力する使用例を示しています。

再生するADPCMデータは32kbps（サンプリング周波数：8kHz、量子化ビット数：4ビット）です。ADPCMデータをADPCM-SPライブラリ関数より伸長して、D/Aコンバータで出力し、サンプリング周波数8kHzはタイマ・アレイ・ユニットを使用して制御しています。ローパスフィルタ回路には、内蔵オペアンプを使用しています。また、再生はキー入力により開始するようにしています。

## (1) 初期設定の主な内容

### < オプション・バイトでの設定 >

- ウォッチドッグ・タイマの動作禁止
- 高速内蔵発振回路の周波数を8MHzに設定
- LVIデフォルト・スタート機能停止
- オンチップ・デバッグを動作許可に設定

### < リセット解除後の初期化処理での設定 >

#### 入出力ポートの設定

- ・ P23, P25, P26, P150をオペアンプ入力用に設定
- ・ P24, P27をオペアンプ出力用に設定
- ・ KR0をキー入力用にキー割り込み信号検出に設定

低電圧検出回路<sup>注</sup>の機能を使用し、2.7V以上の電源電圧を確保

CPUクロックをX1発振回路動作に設定（20MHz）

高速内蔵発振回路の停止

タイマ・アレイ・ユニット0の設定

- ・ チャンネル0をボルテージ・リファレンスのセントリング・タイム用、およびキーのチャタリング対策用に約10msのインターバル・タイマ・モードに設定
- ・ チャンネル4をADPCMデータ再生のサンプリング周波数が8kHzとなるようにインターバル・タイマ・モードに設定

ボルテージ・リファレンスの設定

- ・ 基準電圧源としてボルテージ・リファレンスを選択
- ・ ボルテージ・リファレンスの出力電圧を2.0Vに設定
- ・ ボルテージ・リファレンスの動作安定待ちとして、タイマ・アレイ・ユニット0のチャンネル0を使用し約20msウェイト

オペアンプ1, 2の設定

#### D/Aコンバータの設定

- ・動作モードをリアルタイム出力モードに設定
- ・分解能を12ビットに設定
- ・D/Aコンバータの基準電圧源として $V_{REFOUT}/AV_{REFP}$ 端子を選択

注 低電圧検出回路についての詳細は、ユーザズ・マニュアルを参照してください。

#### (2) メイン・ループ以降の内容

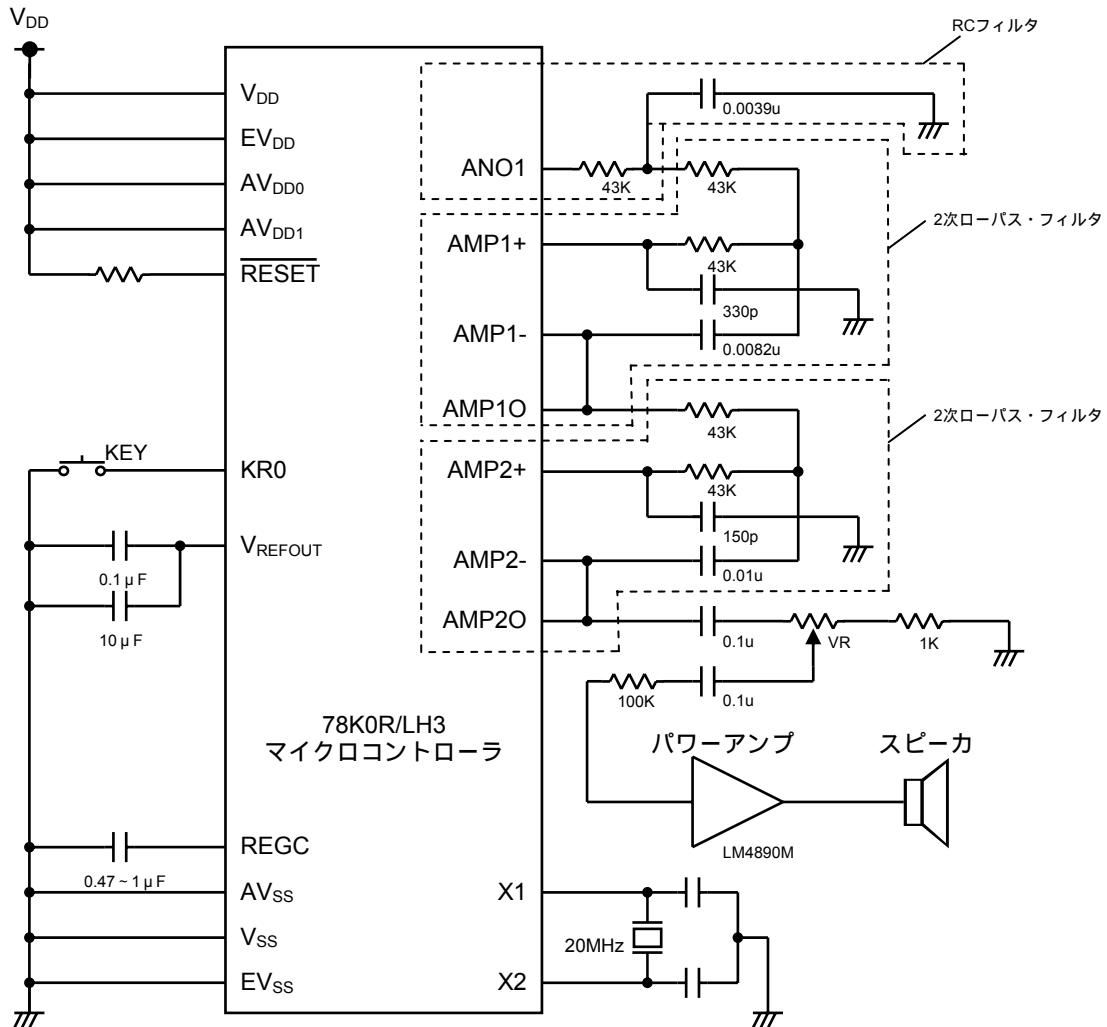
初期設定完了後、HALTモードに移行します。INTKR割り込みの発生によりHALTモードを解除し、キーのチャタリング対策を行い、キー入力中であるかを判断します。キー入力中と判断された場合は、ADPCMデータの再生を行います。なお、再生するADPCMデータは32kbps（サンプリング周波数:8kHz、量子化ビット数:4ビット）です。キー入力中と判断されなかった場合、またADPCMデータ再生終了後は、再びHALTモードに移行します。

## 第2章 回路イメージ

この章では、このサンプル・プログラムを使用する場合の回路イメージおよびマイコン以外の使用デバイスについて説明します。

### 2.1 回路イメージ

回路イメージを次に示します。



注意1. 2.7 V  $V_{DD}$  5.5 Vの電圧範囲で使用してください。

2.  $EV_{DD}$ ,  $AV_{DD0}$ , および  $AV_{DD1}$  は、 $V_{DD}$  と同電位にしてください。

3.  $AV_{SS}$  は  $EV_{SS}$ ,  $V_{SS}$  と同電位にし、GNDに直接接続してください。

4. ボルテージ・リファレンス動作時は、基準電圧安定化用に、 $V_{REFOUT}/AV_{REFP}$ 端子に必ずタンタル・コンデンサ（容量： $10\mu F \pm 30\%$ ，ESR：2（MAX.），ESL：10nH（MAX.））とセラミック・コンデンサ（容量： $0.1\mu F \pm 30\%$ ，ESR：2（MAX.），ESL：10nH（MAX.））を接続してください。また、ボルテージ・リファレンス動作時に、 $V_{REFOUT}/AV_{REFP}$ 端子から電圧を印加しないでください。

（注意は次ページに続きます。）

- 注意5. REGCはコンデンサ (0.47 ~ 1  $\mu$ F) を介し,  $V_{SS}$ に接続してください。
6. 回路イメージ中に記載のない未使用端子は以下のように処理してください。  
 入出力ポート : 出力モードに設定し, オープン (未接続) にしてください  
 入力ポート : 個別に抵抗を介して,  $V_{DD}$ または $V_{SS}$ に接続してください
7. このサンプル・プログラムでは, P40/TOOL0端子, およびP41/TOOL1端子をオンチップ・デバッグ用に使用します。
8. P70端子に内蔵プルアップを接続しています。

## 2.2 マイコン以外の使用デバイス

マイコン以外の使用デバイスを次に示します。

### (1) パワーアンプ

音声の出力値を増幅するために使用します。本アプリケーション例ではLM4890Mを使用します。

### (2) 可変抵抗器

音声出力時のボリューム調整に使用します。

### (3) スピーカ

音声を出力します。

### (4) キー

音声再生開始の入力として, キーを使用します。

## 2.3 端子機能一覧

使用する端子機能を次に示します。

名称	外部デバイス接続時の端子機能	
	機能	
ANO1	D/Aコンバータのアナログ出力	P111
AMP1+	オペアンプ入力 (+ 側)	P25/ANI5
AMP1-	オペアンプ入力 (- 側)	P23/ANI3
AMP1O	オペアンプ出力	P24/ANI4
AMP2+	オペアンプ入力 (+ 側)	P150/ANI8
AMP2-	オペアンプ入力 (- 側)	P26/ANI6
AMP2O	オペアンプ出力	P27/ANI7
KR0	キー入力	P70





## 第3章 ソフトウェアについて

この章では、ダウンロードする圧縮ファイルのファイル構成、使用するマイコンの内蔵周辺機能、サンプル・プログラムの使用する周辺の初期設定と動作概要、およびフロー・チャートを説明します。

### 3.1 ファイル構成

ダウンロードする圧縮ファイルのファイル構成は、次のようになっています。

ファイル名	説明	同封圧縮 (*.zip) ファイル	
			
main.asm (アセンブリ言語版) ----- main.c (C言語版)	マイコンのハードウェア初期化処理と、メイン処理のソース・ファイル	注1	注1
data_playrom.asm (アセンブリ言語版) ----- data_playrom.c (C言語版)	ADPCMデータ・テーブル・ファイル	注2	注2
op.asm	オプション・バイト設定用アセンブラ・ソース・ファイル (ウォッチドッグ・タイマの設定、高速内蔵発振クロック周波数の選択、LVIデフォルト・スタート機能の設定、オンチップ・デバッグ動作の設定などを行います。)		
78K0R_Lx3_PlayBack.prw	統合開発環境 PM+用ワーク・スペース・ファイル		
78K0R_Lx3_PlayBack.prj	統合開発環境 PM+用プロジェクト・ファイル		

注1. アセンブリ言語版には「main.asm」、C言語版には「main.c」が同封されています。

2. アセンブリ言語版には「data\_playrom.asm」、C言語版には「data\_playrom.c」が同封されています。

**注意** 本サンプル・プログラムでは、ADPCMライブラリを使用します。そのためC言語版プログラムにはADPCM-SPの標準ヘッダ・ファイル(adpcmsp.h)とADPCM-SPのライブラリ・ファイル(adpcmsp.lib)、アセンブリ言語版プログラムにはADPCM-SPのライブラリ・ファイル(adpcmsp.lib)が必要です。ADPCM-SPIは開発ツールのダウンロード・サイト(<http://necel.com/micro/ja/development/asia/78k0r.html>)より入手してください。

備考



: ソース・ファイルのみ同封



: 統合開発環境 PM+で使用するファイルを同封

## 3.2 使用する内蔵周辺機能

このサンプル・プログラムでは、マイコンに内蔵する次の周辺機能を使用します。

- ・タイマ・アレイ・ユニット0 (TAU0) のチャンネル0 :  
インターバル・タイマに設定し、キー入力のチャタリング対策と、ボルテージ・リファレンスの安定待ちに使用します。
- ・タイマ・アレイ・ユニット0 (TAU0) のチャンネル4 :  
インターバル・タイマに設定し、ADPCMデータ再生のサンプリング周波数8kHzを生成します。
- ・ボルテージ・リファレンス :  
D/Aコンバータの2.0Vの基準電圧を生成します。
- ・D/Aコンバータ :  
ADPCMデータを音声として出力します。
- ・オペアンプ :  
D/Aコンバータから出力される音声のフィルタ回路として使用します。
- ・低電圧検出回路 :  
2.7V  $V_{DD}$ の確認用に使用します。
- ・端子機能 :  
使用する端子機能を以下の表に示します。

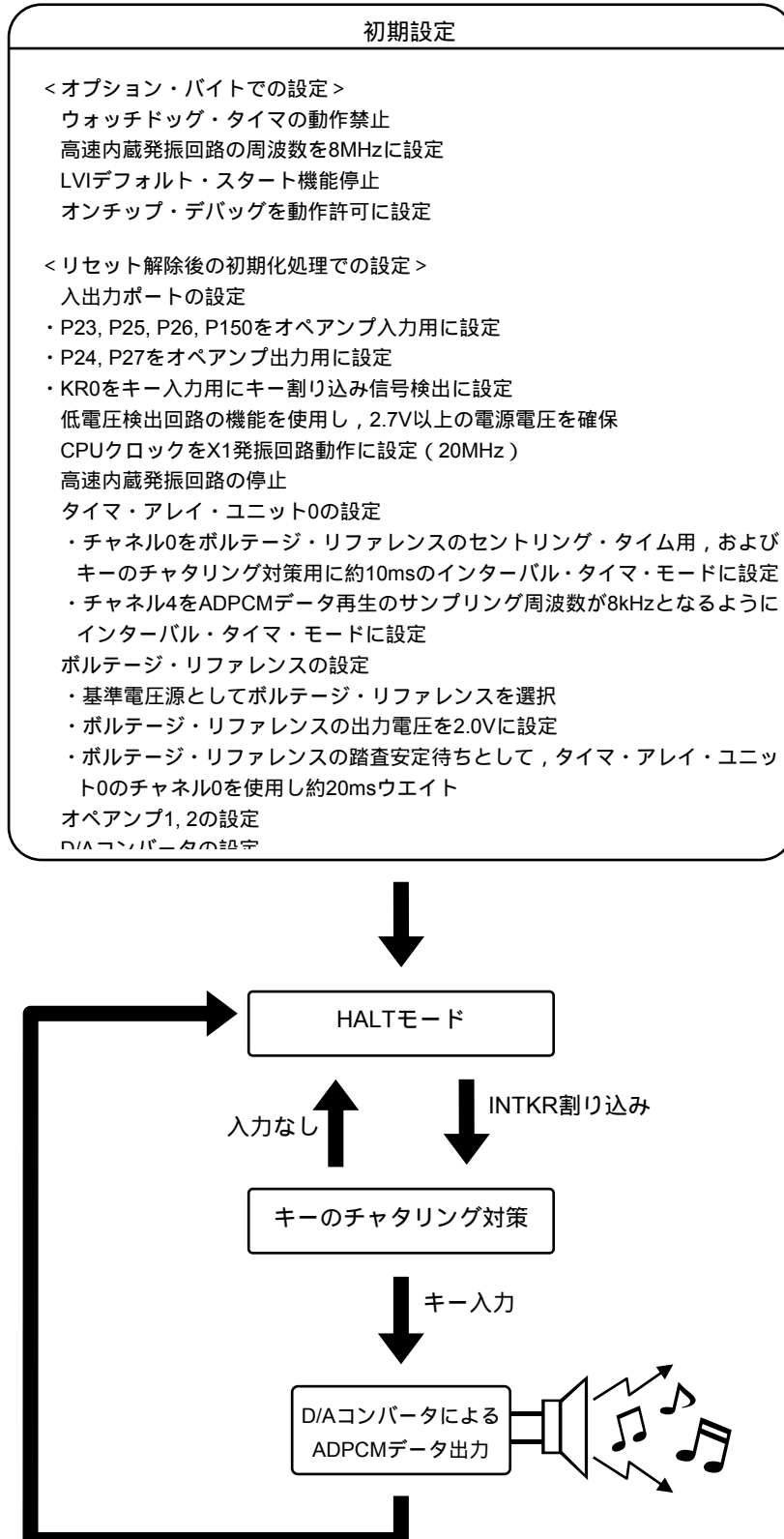
外部デバイス接続時の端子機能		兼用端子
名称	機能	
ANO1	D/Aコンバータのアナログ出力	P111
AMP1+	オペアンプ入力 (+ 側)	P25/ANI5
AMP1-	オペアンプ入力 (- 側)	P23/ANI3
AMP1O	オペアンプ出力	P24/ANI4
AMP2+	オペアンプ入力 (+ 側)	P150/ANI8
AMP2-	オペアンプ入力 (- 側)	P26/ANI6
AMP2O	オペアンプ出力	P27/ANI7
KR0	キー入力	P70

### 3.3 初期設定と動作概要

このサンプル・プログラムでは、初期設定にて、メイン・システム・クロックの選択や、入出力ポート、タイマ・アレイ・ユニット0、ボルテージ・リファレンス、オペアンプ、D/Aコンバータの設定などを行います。

初期設定完了後は、キーの入力により、ADPCMデータを再生します。

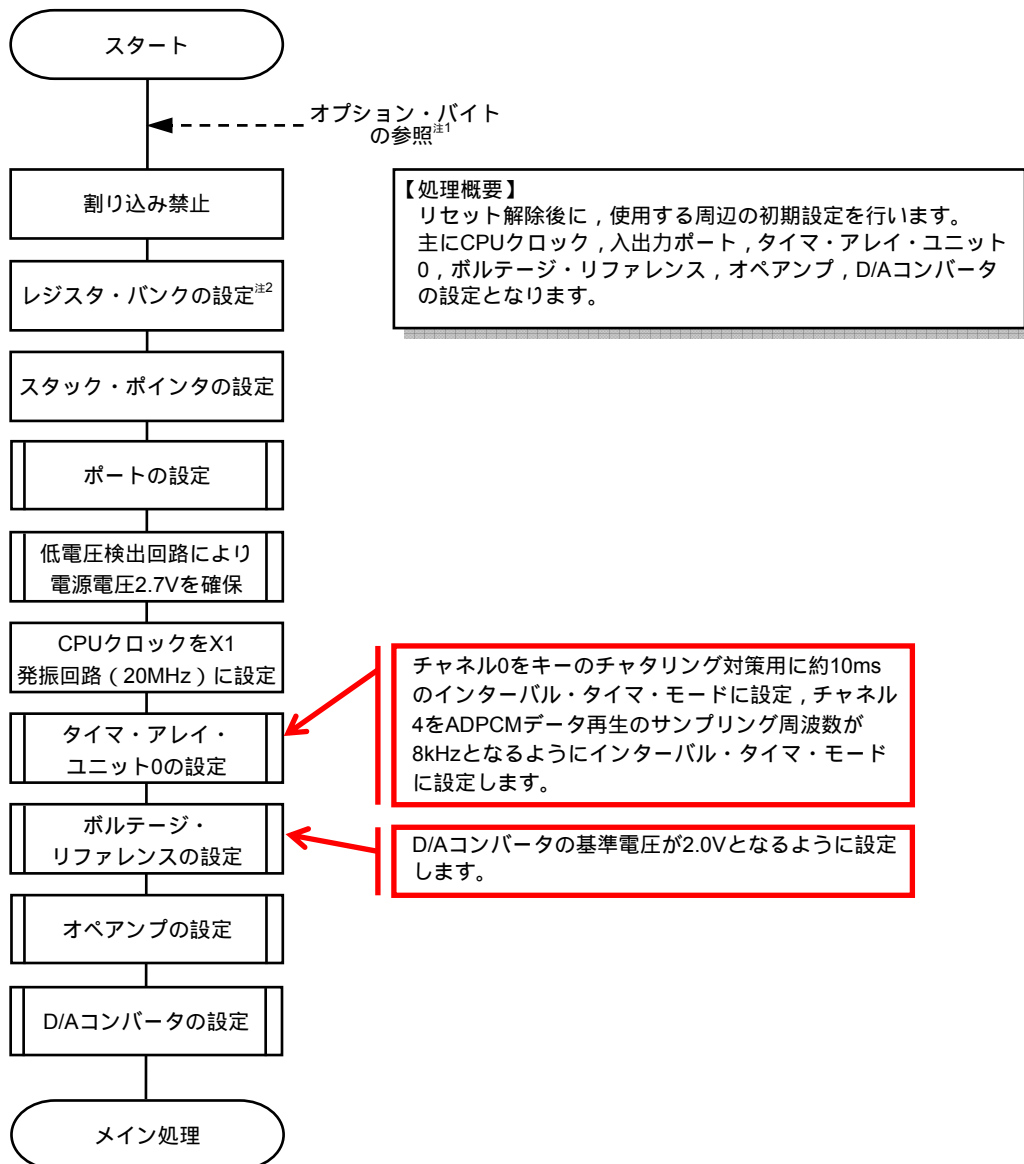
詳細については、次の状態遷移図（ステート・チャート）に示します。



### 3.4 フロー・チャート

このサンプル・プログラムのフロー・チャートを次に示します。

<リセット解除後の初期化処理での設定>



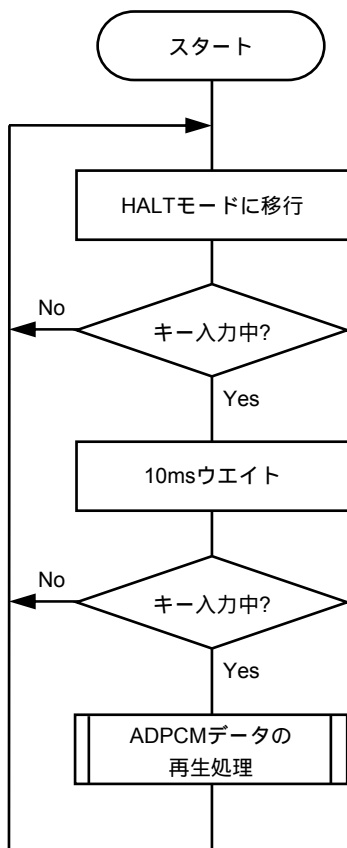
注1. オプション・バイトの参照は、リセット解除後にマイコンが自動的に行います。このサンプル・プログラムでは、オプション・バイトで以下の設定を行います。

- ・ウォッチドッグ・タイマの動作禁止
- ・高速内蔵発振回路の周波数を8MHzに設定
- ・LVIデフォルト・スタート機能停止
- ・オンチップ・デバッグを動作許可に設定

2. 78K0R/LH3の汎用レジスタは、4レジスタ・バンク構成になっていますので、通常処理で使用するレジスタと割り込み時で使用するレジスタのバンクを切り替えることにより、効率のよいプログラムを作成できます。なお、このサンプル・プログラムでは、レジスタ・バンク0のみを使用します。

注意 C言語版のサンプル・プログラムの場合、レジスタ・バンクの設定およびスタック・ポインタの設定をスタートアップ・ルーチンで行いますので、ソース・プログラム(main.c)には記述しません。なお、スタートアップ・ルーチンについての詳細はCC78K0Rの操作編のユーザーズ・マニュアルを参照してください。

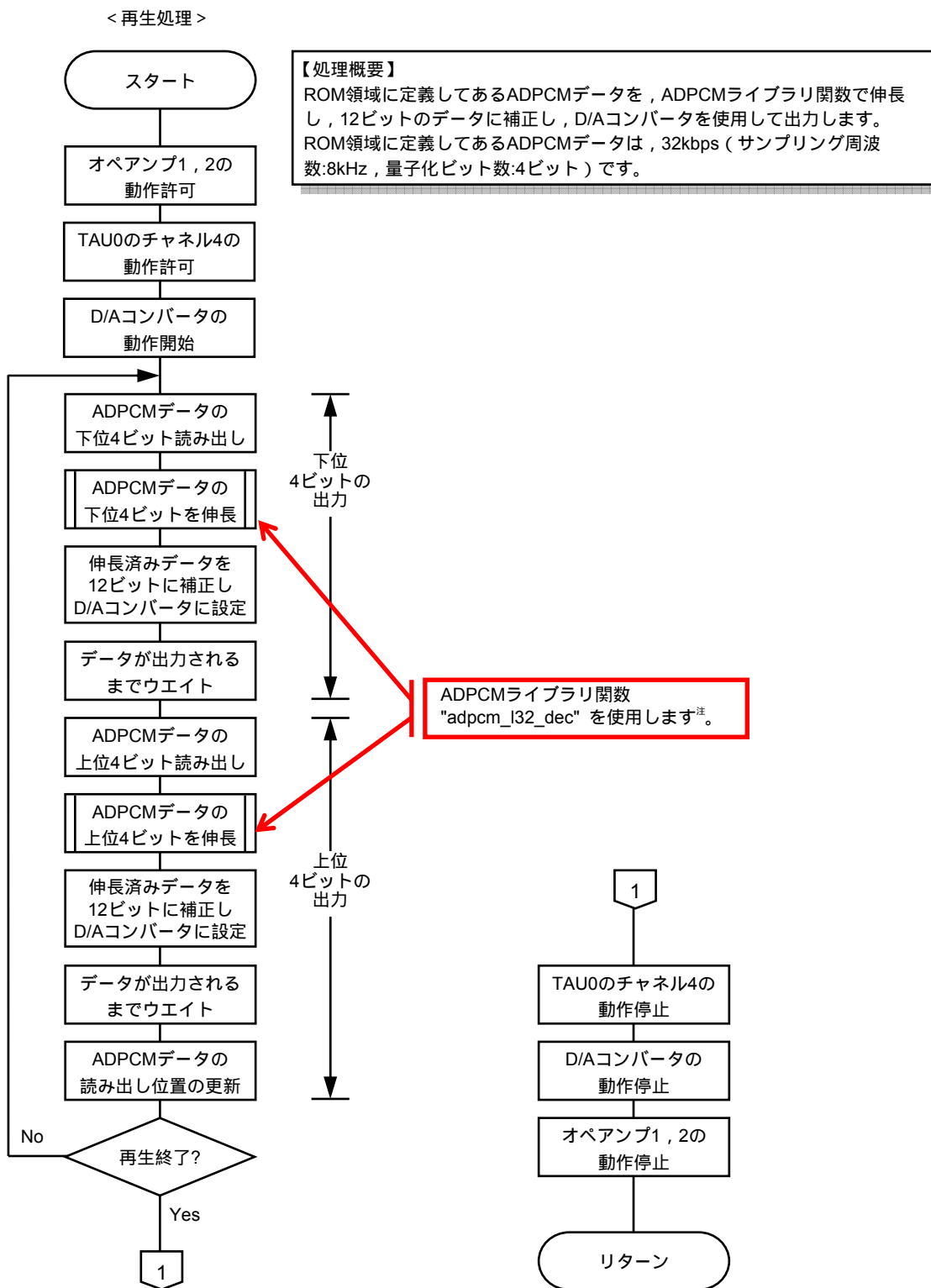
<メイン処理>



**【処理概要】**

キー入力があるとADPCMデータを再生します。

INTKR割り込みが発生するまで、HALTモードとなります。  
INTKR割り込みの発生によりHALTモードを解除し、キーのチャタリング対策として約10msウエイトし、キー入力中であるかを判断します。  
キー入力中であった場合は、ADPCMデータの再生処理を呼び出します。



**注** 関数の詳細は「ADPCM-SP 音声圧縮/伸長ソフトウェア・パッケージ 78K0Rマイクロコントローラ編 ユーザーズ・マニュアル」を参照してください。

**注意** ADPCMデータ再生処理が完了するまで，他の処理を割り込ませないでください。

## 第4章 設定方法について

この章では、周辺ハードウェア・マクロ使用の設定、タイマ・アレイ・ユニット0の設定、ボルテージ・リファレンスの設定、オペアンプの設定、D/Aコンバータの設定、ソフトウェア記述例、および再生処理について説明します。

その他の初期設定については、「78K0R/Lx3 サンプル・プログラム（初期設定） LED点灯のスイッチ制御編 アプリケーション・ノート」を参照してください。

レジスタ設定方法の詳細については、ユーザーズ・マニュアルを参照してください。

アセンブラ命令については、「78K0Rシリーズ 命令編 ユーザーズ・マニュアル」を参照してください。

### 4.1 周辺ハードウェア・マクロ使用の設定

周辺ハードウェア・マクロ使用の設定は、次のレジスタを使用します。

・周辺イネーブル・レジスタ0 (PER0)

#### (1) 周辺イネーブル・レジスタ0 (PER0)

各周辺ハードウェア・マクロの使用可否を設定するレジスタです。使用しないハードウェアへはクロック供給も停止させることで、低消費電力化とノイズ低減をはかります。

PER0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

図4-1 周辺イネーブル・レジスタ0 (PER0) のフォーマット

RTCEN	DACEN	ADCEN	IICAEN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
							<b>TAU0EN</b>
							0 入力クロック供給停止
							<b>1</b> 入力クロック供給
		<b>ADCEN</b>					A/Dコンバータ、オペアンプ、ボルテージ・リファレンスの入力クロックの制御
		0					入力クロック供給停止
		<b>1</b>					入力クロック供給
	<b>DACEN</b>						D/Aコンバータの入力クロックの制御
	0						入力クロック供給停止
	<b>1</b>						入力クロック供給

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

## 4.2 タイマ・アレイ・ユニット0の設定

タイマ・アレイ・ユニット0は、次の5種類のレジスタを使用します。

- ・周辺イネーブル・レジスタ0 (PER0)
- ・タイマ・クロック選択レジスタ0 (TPS0)
- ・タイマ・モード・レジスタ00, 04 (TMR00, TMR04)
- ・タイマ・データ・レジスタ00, 04 (TDR00, TDR04)
- ・タイマ・チャンネル開始レジスタ0 (TS0)

### 【音声データの再生を行う場合のタイマ・アレイ・ユニット0の設定手順例】

周辺イネーブル・レジスタ0 (PER0) のビット0 (TAU0EN) をセット (1) (4.1参照)

タイマ・クロック選択レジスタ0 (TPS0) でCK00を $f_{CLK}/2^2$ , CK01を $f_{CLK}$ に設定

タイマ・モード・レジスタ00 (TMR00) でチャンネル0の動作クロックをCK00, 動作モードをインターバル・タイマ・モードに設定

タイマ・データ・レジスタ00 (TDR00) でチャンネル0のインターバルを約10msに設定

タイマ・モード・レジスタ04 (TMR04) でチャンネル4の動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定

タイマ・データ・レジスタ04 (TDR04) でチャンネル4のインターバルを約125  $\mu$ sに設定



(1) タイマ・クロック選択レジスタ0 (TPS0)

TPS0は、タイマ・アレイ・ユニット0の各チャンネルに共通して供給される2種類の動作クロック (CK00, CK01) を選択する16ビット・レジスタです。TPS0の7-4ビットでCK01を、3-0ビットでCK00を選択します。

タイマ動作中のTPS0の書き換えは、次の場合のみ可能です。

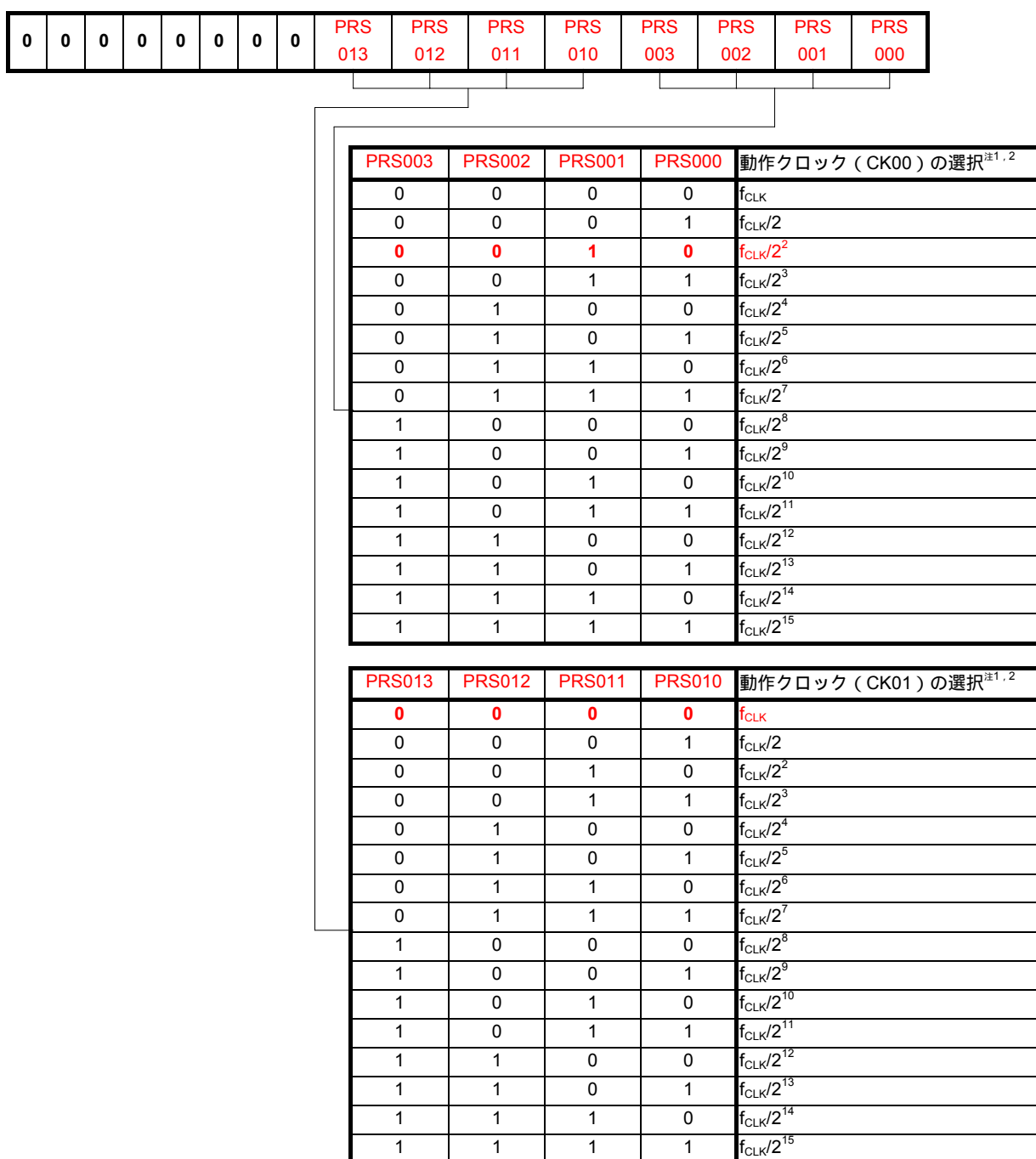
PRS000-PRS003ビットの書き換え：

CKS0n = 0に設定しているチャンネルがすべて動作停止状態 (TE0n = 0) の場合のみ可能

PRS010-PRS013ビットの書き換え：

CKS0n = 1に設定しているチャンネルがすべて動作停止状態 (TE0n = 0) の場合のみ可能

図4 - 2 タイマ・クロック選択レジスタ0 (TPS0) のフォーマット



- 注1.  $f_{CLK}$ に選択しているクロックを変更(システム・クロック制御レジスタ(CKC)の値を変更)する場合は、タイマ・アレイ・ユニットを停止(TT0 = 00FFH, TT1 = 000FH)させてください。
2. SDIV=0, CCSmn=1かつTISmn=1の場合にかぎり、CPUクロックを切り替えても、TAUmの使用は継続可能です(m = 0, 1 mn = 00-07, 10-13)。ただし、CPUクロックの切り替え時に、次の制限事項があります。
- ・CPUクロック切り替え時に、次のようにソース・クロックずれが発生します。
    - メイン・システム・クロック サブシステム・クロック(CSS=0 1): -1クロック
    - サブシステム・クロック メイン・システム・クロック(CSS=1 0): +1クロック

**注意** ビット15-8には、必ず0を設定してください。

- 備考1.**  $f_{CLK}$ : CPU/周辺ハードウェア・クロック周波数
2. 図の赤字部分がサンプル・プログラムでの設定値となります。

(2) タイマ・モード・レジスタ00, 04 (TMR00, TMR04)

TMR00, TMR04は、タイマ・アレイ・ユニット0のチャンネル0およびチャンネル4の動作モード設定レジスタです。動作クロック (MCK) の選択, カウント・クロックの選択, マスタ/スレーブの選択, スタート・トリガとキャプチャ・トリガの設定, タイマ入力の有効エッジ選択, 動作モード (インターバル, キャプチャ, イベント・カウンタ, ワンカウント, キャプチャ&ワンカウント) 設定を行います。

図4-3 タイマ・モード・レジスタ00 (TMR00) のフォーマット (1/2)

CKS00	0	0	CCS00	MAST ER00	STS002	STS001	STS000	CIS001	CIS000	0	0	MD003	MD002	MD001	MD000
			STS002			STS001	STS000	チャンネル0のスタート・トリガ, キャプチャ・トリガの設定							
			0			0	0	ソフトウェア・トリガ・スタートのみ有効 (他のトリガ要因を非選択にする)							
			0			0	1	Tlpq端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用							
			0			1	0	Tlpq端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用							
			1			0	0	マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)							
			上記以外			設定禁止									
MASTER00			チャンネル0の単体動作機能, 連動動作機能のスレーブ/連動動作機能のマスタの選択												
0			単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作												
1			連動動作機能でマスタ・チャンネルとして動作												
偶数チャンネルのみマスタ・チャンネル (MASTER00 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER00 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER00 = 0 にします。															
CCS00			チャンネル0のカウント・クロック (TCLK) の選択												
0			CKS00ビットで指定した動作クロックMCK												
1			Tlpq端子からの入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)												
カウント・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。															
CKS00			チャンネル0の動作クロック (MCK) の選択												
0			TPS0レジスタで設定した動作クロックCK00												
1			TPS0レジスタで設定した動作クロックCK01												
動作クロックMCKは, エッジ検出回路に使用されます。また, CCS00ビットの設定によりカウント・クロック (TCLK) を生成します。															

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-3 タイマ・モード・レジスタ00 (TMR00) のフォーマット (2/2)

CKS00	0	0	CCS00	MAST ER00	STS002	STS001	STS000	CIS001	CIS000	0	0	MD003	MD002	MD001	MD000
-------	---	---	-------	-----------	--------	--------	--------	--------	--------	---	---	-------	-------	-------	-------

動作モード (MD003-MD001で設定)		MD000	カウント・スタートと割り込みの設定				
・インターバル・タイマ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
	1	1	カウント開始時にタイマ割り込みを発生する (タイマ出力も変化させる)。				
・イベント・カウンタ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。				
・ワンカウント・モード	0	0	カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。				
	1	1	カウント動作中のスタート・トリガを有効とする <sup>注</sup> 。 その際に割り込みも発生する。				
・キャプチャ&ワンカウント・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。 カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。				
上記以外			設定禁止				
MD003	MD002	MD001	MD000	チャンネル0の動作モードの設定		TCRのカウント動作	単体動作
0	0	0	1/0	インターバル・タイマ・モード		ダウン・カウント	可
0	1	0	1/0	キャプチャ・モード		アップ・カウント	可
0	1	1	0	イベント・カウンタ・モード		ダウン・カウント	可
1	0	0	1/0	ワンカウント・モード		ダウン・カウント	不可
1	1	0	0	キャプチャ&ワンカウント・モード		アップ・カウント	可
上記以外			設定禁止				
CIS001	CIS000	T100端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジ選択 (使用するタイマ入力はTIS0レジスタで選択)					
0	0	立ち下がりエッジ					
0	1	立ち上がりエッジ					
1	0	両エッジ (ロウ・レベル幅測定時) スタート・トリガ: 立ち下がりエッジ, キャプチャ・トリガ: 立ち上がりエッジ					
1	1	両エッジ (ハイ・レベル幅測定時) スタート・トリガ: 立ち上がりエッジ, キャプチャ・トリガ: 立ち下がりエッジ					
STS002-STS000ビット = 010B時以外で両エッジ指定を使用する場合は, CIS001-CIS000ビット = 10Bに設定してください							

**注** 動作中にスタート・トリガ (TS00 = 1) が掛かると, カウンタをクリアし, 割り込みを発生して再カウント・スタートします。

**注意** ビット14, 13, 5, 4には, 必ず0を設定してください。

**備考** 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-4 タイマ・モード・レジスタ04 (TMR04) のフォーマット (1/2)

CKS04	0	0	CCS04	MAST ER04	STS042	STS041	STS040	CIS041	CIS040	0	0	MD043	MD042	MD041	MD040
STS042			STS041			STS040			チャンネル4のスタート・トリガ、キャプチャ・トリガの設定						
0			0			0			ソフトウェア・トリガ・スタートのみ有効 (他のトリガ要因を非選択にする)						
0			0			1			Tlpq端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジを, スタート・トリガ, キャプチャ・トリガの両方に使用						
0			1			0			Tlpq端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の両エッジを, スタート・トリガとキャプチャ・トリガに分けて使用						
1			0			0			マスタ・チャンネルの割り込み信号を使用 (連動動作機能のスレーブ・チャンネル時)						
上記以外			上記以外			上記以外			設定禁止						
MASTER04			チャンネル4の単体動作機能, 連動動作機能のスレーブ / 連動動作機能のマスタの選択												
0			単体動作機能, または連動動作機能でスレーブ・チャンネルとして動作												
1			連動動作機能でマスタ・チャンネルとして動作												
偶数チャンネルのみマスタ・チャンネル (MASTER04 = 1) に設定できます。 奇数チャンネルは必ずスレーブ・チャンネル (MASTER04 = 0) で使用してください。 また, 単体動作機能として使用するチャンネルは, MASTER04 = 0 にします。															
CCS04			チャンネル4のカウント・クロック (TCLK) の選択												
0			CKS04ビットで指定した動作クロックMCK												
1			Tlpq端子からの入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジ (使用するタイマ入力はTIS0レジスタで選択)												
カウント・クロック (TCLK) は, タイマ・カウンタ, 出力制御回路, 割り込み制御回路に使用されます。															
CKS04			チャンネル4の動作クロック (MCK) の選択												
0			TPS0レジスタで設定した動作クロックCK00												
1			TPS0レジスタで設定した動作クロックCK01												
動作クロックMCKは, エッジ検出回路に使用されます。また, CCS04ビットの設定によりカウント・クロック (TCLK) を生成します。															

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

図4-4 タイマ・モード・レジスタ04 (TMR04) のフォーマット (2/2)

CKS04	0	0	CCS04	MAST ER04	STS042	STS041	STS040	CIS041	CIS040	0	0	MD043	MD042	MD041	MD040
-------	---	---	-------	-----------	--------	--------	--------	--------	--------	---	---	-------	-------	-------	-------

動作モード (MD043-MD041で設定)		MD040	カウント・スタートと割り込みの設定			
・インターバル・タイマ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。			
	1	1	カウント開始時にタイマ割り込みを発生する (タイマ出力も変化させる)。			
・イベント・カウンタ・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。			
・ワンカウント・モード	0	0	カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。			
	1	1	カウント動作中のスタート・トリガを有効とする <sup>注</sup> 。 その際に割り込みも発生する。			
・キャプチャ&ワンカウント・モード	0	0	カウント開始時にタイマ割り込みを発生しない (タイマ出力も変化しない)。 カウント動作中のスタート・トリガは無効とする。 その際に割り込みも発生しない。			
上記以外			設定禁止			
MD043	MD042	MD041	MD040	チャンネル4の動作モードの設定	TCRのカウント動作	単体動作
0	0	0	1/0	インターバル・タイマ・モード	ダウン・カウント	可
0	1	0	1/0	キャプチャ・モード	アップ・カウント	可
0	1	1	0	イベント・カウンタ・モード	ダウン・カウント	可
1	0	0	1/0	ワンカウント・モード	ダウン・カウント	不可
1	1	0	0	キャプチャ&ワンカウント・モード	アップ・カウント	可
上記以外				設定禁止		
CIS041	CIS040	TI04端子入力信号, $f_{SUB}/2$ , $f_{SUB}/4$ , またはINTRTC1の有効エッジ選択 (使用するタイマ入力はTIS0レジスタで選択)				
0	0	立ち下がりエッジ				
0	1	立ち上がりエッジ				
1	0	両エッジ (ロウ・レベル幅測定時) スタート・トリガ: 立ち下がりエッジ, キャプチャ・トリガ: 立ち上がりエッジ				
1	1	両エッジ (ハイ・レベル幅測定時) スタート・トリガ: 立ち上がりエッジ, キャプチャ・トリガ: 立ち下がりエッジ				
STS042-STS040ビット = 010B時以外で両エッジ指定を使用する場合は, CIS041-CIS040ビット = 10Bに設定してください						

**注** 動作中にスタート・トリガ (TS04 = 1) が掛かると, カウンタをクリアし, 割り込みを発生して再カウント・スタートします。

**注意** ビット14, 13, 5, 4には, 必ず0を設定してください。

**備考** 図の赤字部分がサンプル・プログラムでの設定値となります。



### 4.3 ボルテージ・リファレンスの設定

ボルテージ・リファレンスは、次のレジスタを使用します。

- ・アナログ基準電圧制御レジスタ (ADVRC)

**【音声データの再生を行う場合のボルテージ・リファレンスの設定手順例】**

- 周辺イネーブル・レジスタ0 (PER0) のビット5 (ADCEN) をセット (1) (4.1参照)
- アナログ基準電圧制御レジスタ (ADVRC) のビット3 (VRSEL) をセット (1)
- アナログ基準電圧制御レジスタ (ADVRC) のビット1, 0 (VRGV, VRON) をセット (1)
- ボルテージ・リファレンスの動作安定待ちのため、ソフトウェアで約20msウエイト

**(1) アナログ基準電圧制御レジスタ (ADVRC)**

A/DコンバータとD/Aコンバータの基準電圧源の選択, A/Dコンバータ用入力ゲート昇圧回路の動作の制御, およびボルテージ・リファレンス (VR) の動作を制御するレジスタです。

ADVRCは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

図4-7 アナログ基準電圧制御レジスタ (ADVRC) のフォーマット

ADREF <sup>※</sup>	0	0	0	VRSEL	0	VRGV	VRON
--------------------	---	---	---	-------	---	------	------

VRSEL	VRGV	VRON	A/DコンバータとD/Aコンバータの+側の基準電圧源の選択	ボルテージ・リファレンスの動作制御	ボルテージ・リファレンスの出力電圧選択	A/Dコンバータ用入力ゲート昇圧回路の動作制御	使用する変換モードとの関係
0	0	0	AV <sub>REFP</sub> (外部基準電圧入力)	動作停止 (Hi-Z)	2.5V	動作停止	変換モード1の場合に設定可
0	1	0			2.0V	動作許可	変換モード2, 3の場合に設定可
1	0	0	V <sub>REFOUT</sub> (ボルテージ・リファレンス出力)	動作停止 (プルダウン出力)	2.5V	動作停止	-
1	0	1		動作許可	2.5V	動作許可	変換モード2, 3の場合に設定可
1	1	0		動作停止 (プルダウン出力)	2.0V		-
1	1	1		動作許可	2.0V		変換モード2, 3の場合に設定可
上記以外			設定禁止				

注 A/Dコンバータの機能なので、本サンプル・プログラムでは使用しません。

(注意と備考は次ページにあります。)



- 注意1. ビット6-4,2には、必ず0を設定してください。
2. ボルテージ・リファレンス動作時は、基準電圧安定化用に、 $V_{REFOUT}/AV_{REFP}$ 端子に必ずタンタル・コンデンサ（容量： $10\mu F \pm 30\%$ ，ESR：2（MAX.），ESL：10nH（MAX.））とセラミック・コンデンサ（容量： $0.1\mu F \pm 30\%$ ，ESR：2（MAX.），ESL：10nH（MAX.））を接続してください。また、ボルテージ・リファレンス動作時に、 $V_{REFOUT}/AV_{REFP}$ 端子から電圧を印加しないでください。
  3. A/Dコンバータの+側の基準電圧（ $AD_{REFP}$ ）とD/Aコンバータの+側の基準電圧（ $DA_{REFP}$ ）にボルテージ・リファレンス出力（ $V_{REFOUT}$ ）を使用する場合、必ずVRSELを1に設定したあとに、VRONを1に設定してください。
  4. A/Dコンバータの+側の基準電圧（ $AD_{REFP}$ ）とD/Aコンバータの+側の基準電圧（ $DA_{REFP}$ ）の両方に、ボルテージ・リファレンス出力（ $V_{REFOUT}$ ）を使用する（VRSEL = 1かつDAREF = 1）場合、A/D変換中にD/A変換値設定レジスタWn（DACSWn：n = 0, 1）を書き換えしないでください。DACSWnレジスタを書き換える場合は、A/D変換停止時（ADCS = 0）に行ってください。
  5. ボルテージ・リファレンスが動作している（VRON = 1）ときに、VRGVでボルテージ・リファレンスの出力電圧を変更しないでください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

## 4.4 オペアンプの設定

オペアンプは、次の4種類のレジスタを使用します。

- ・周辺イネーブル・レジスタ0 (PER0)
- ・オペアンプ制御レジスタ (OAC)
- ・A/Dポート・コンフィギュレーション・レジスタ (ADPC)
- ・ポート・モード・レジスタ2, 15 (PM2, PM15)

### 【音声データの再生を行う場合のオペアンプの設定手順例】

周辺イネーブル・レジスタ0 (PER0) のビット5 (ADCEN) をセット (1) (4.1参照)

ADPCレジスタで、使用する端子 (AMP1-, AMP1+, AMP1O, AMP2-, AMP2+, AMP2O) を、アナログ入力に設定

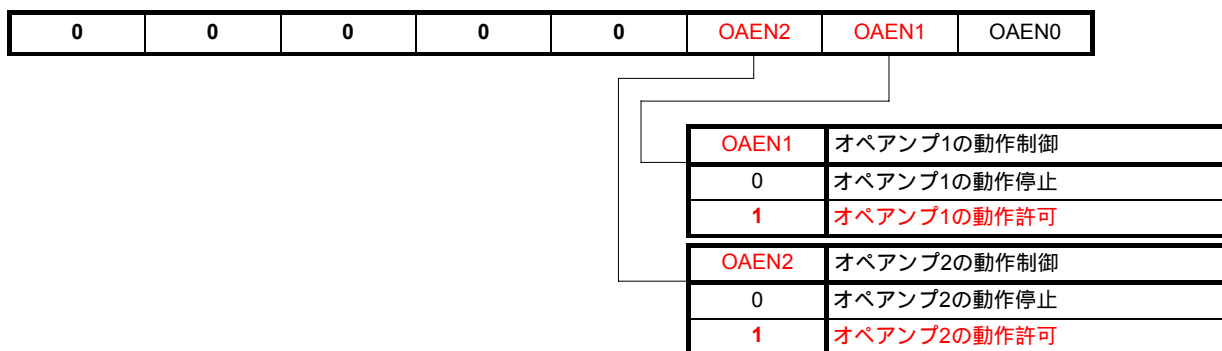
PM2レジスタ、およびPM15レジスタで、使用する端子 (AMP1-, AMP1+, AMP1O, AMP2-, AMP2+, AMP2O) を、入力モードに設定

OACレジスタのOAEN1ビット, OAEN2ビットをセット (1) し、オペアンプ1, 2の動作を許可  
オペアンプの動作安定待ちのため、約20  $\mu$ sウエイト

### (1) オペアンプ制御レジスタ (OAC)

OACレジスタはオペアンプ2-0の動作を制御するレジスタです。

図4 - 8 オペアンプ制御レジスタ (OAC) のフォーマット



注意1. オペアンプで使用する端子は、ADPCレジスタでアナログ入力に選択してください。

2. オペアンプ使用時に、オペアンプで使用していないポート2, 15の端子を、デジタル入力として使用する場合、入力レベルが固定になるようにしてください。
3. ビット7-3には必ず0を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

(2) A/Dポート・コンフィギュレーション・レジスタ (ADPC)

ANI0/AMP0-/P20-ANI7/AMP2O/P27, ANI8/AMP2+/P150-ANI10/P152, ANI15/AV<sub>REFM</sub>/P157端子を, アナログ入力 / ポートのデジタル入出力に切り替えるレジスタです。オペアンプで使用する端子は, アナログ入力に設定してください。

図4 - 9 A/Dポート・コンフィギュレーション・レジスタ (ADPC) のフォーマット

0	0	0	ADPC4	ADPC3	ADPC2	ADPC1	ADPC0
---	---	---	-------	-------	-------	-------	-------

AD PC					アナログ入力 (A) / デジタル入出力 (D) の切り替え											
4	3	2	1	0	ポート15				ポート2							
					ANI15 /AV <sub>REFP</sub> /P157	ANI10 /P152	ANI9 /P151	ANI8 /AMP2+ /P150	ANI7 /AMP2O /P27	ANI6 /AMP2- /P26	ANI5 /AMP1+ /P25	ANI4 /AMP1O /P24	ANI3 /AMP1- /P23	ANI2 /AMP0+ /P22	ANI1 /AMP0O /P21	ANI0 /AMP0- /P20
0	0	0	0	0	A	A	A	A	A	A	A	A	A	A	A	A
0	0	0	0	1	A	A	A	A	A	A	A	A	A	A	A	D
0	0	0	1	0	A	A	A	A	A	A	A	A	A	A	D	D
0	0	0	1	1	A	A	A	A	A	A	A	A	A	D	D	D
0	0	1	0	0	A	A	A	A	A	A	A	A	D	D	D	D
0	0	1	0	1	A	A	A	A	A	A	A	D	D	D	D	D
0	0	1	1	0	A	A	A	A	A	D	D	D	D	D	D	D
0	0	1	1	1	A	A	A	A	A	D	D	D	D	D	D	D
0	1	0	0	0	A	A	A	A	D	D	D	D	D	D	D	D
0	1	0	0	1	A	A	A	D	D	D	D	D	D	D	D	D
0	1	0	1	0	A	A	D	D	D	D	D	D	D	D	D	D
0	1	1	1	1	A	D	D	D	D	D	D	D	D	D	D	D
1	0	0	0	0	D	D	D	D	D	D	D	D	D	D	D	D
上記以外					設定禁止											

- 注意1. オペアンプで使用する端子は, ポート・モード・レジスタ2, 15 (PM2, PM15) で入力モードに選択してください。
2. ビット7-5には必ず0を設定してください。

備考 図の赤字部分がサンプル・プログラムでの設定値となります。

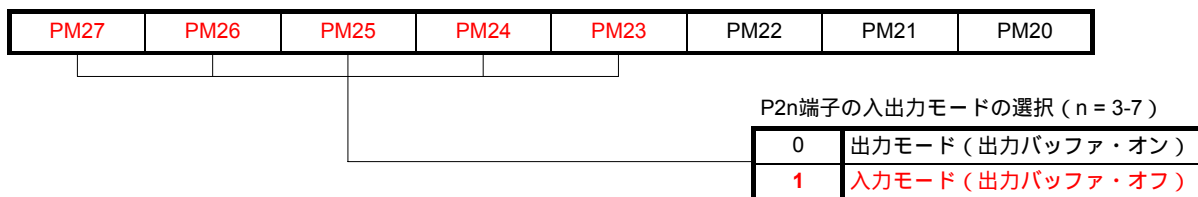
(3) ポート・モード・レジスタ2, 15 (PM2, PM15)

AMP1-/ANI3/P23 , AMP10/ANI4/P24 , AMP1+/ANI5/P25 , AMP2-/ANI6/P26 , AMP20/ANI7/P27 , AMP2+/ANI8/P150端子をオペアンプで使用するとき, PM23-PM27, PM150にそれぞれ1を設定してください。

このときP23-P27, P150の出力ラッチは, 0または1のどちらでもかまいません。

PM23-PM27, PM150にそれぞれ0を設定した場合は, オペアンプの端子として使用することはできません。

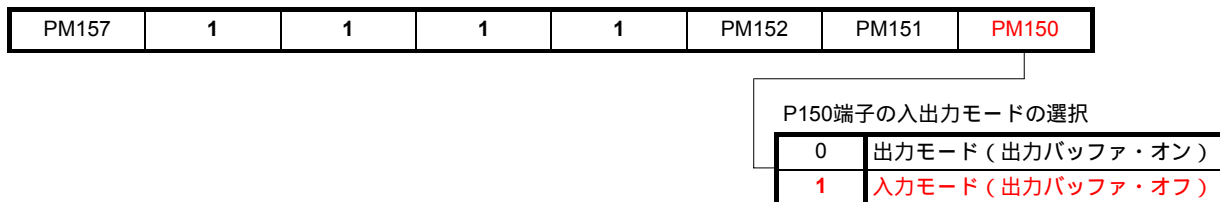
図4 - 10 ポート・モード・レジスタ2 (PM2) のフォーマット



**注意** アナログ入力ポートとして設定した端子を読み出した場合は, 端子レベルではなく常に0が読み出されます。

**備考** 図の赤字部分がサンプル・プログラムでの設定値となります。

図4 - 11 ポート・モード・レジスタ15 (PM15) のフォーマット



**注意** アナログ入力ポートとして設定した端子を読み出した場合は, 端子レベルではなく常に0が読み出されます。

**備考** 図の赤字部分がサンプル・プログラムでの設定値となります。

AMP1-/ANI3/P23 , AMP10/ANI4/P24 , AMP1+/ANI5/P25 , AMP2-/ANI6/P26 , AMP20/ANI7/P27 , AMP2+/ANI8/P150端子の機能は ,ADPCレジスタ ,ADSレジスタ ,PM2レジスタ ,PM15レジスタ ,OAENnビット ,ADREFビットの設定で決定します。

表4 - 1 ANI3/AMP1-/P23, ANI5/AMP1+/P25,ANI6/AMP2-/P26, ANI8/AMP2+/P150端子機能の設定

ADPCレジスタ	PM2, PM15レジスタ	OAENnビット	ADSレジスタ	ANI3/AMP1-/P23, ANI5/AMP1+/P25, ANI6/AMP2-/P26, ANI8/AMP2+/P150端子機能の設定
デジタル入出力選択	入力モード	0	-	デジタル入力
		1	-	設定禁止
	出力モード	0	-	デジタル出力
		1	-	設定禁止
アナログ入力選択	入力モード	0	ANI選択	アナログ入力 (A/D変換対象)
			ANI非選択	アナログ入力 (A/D変換非対象)
		1	ANI選択	設定禁止
			ANI非選択	オペアンプ入力
	出力モード	-	-	設定禁止

備考1. n = 1, 2

2. 表の赤字部分がサンプル・プログラムでの設定値となります。

**注意** オペアンプ使用時には ,AMPn+, AMPn-, AMPnO端子を使用するため ,端子に兼用するアナログ入力機能は使用できません。

表4 - 2 ANI4/AMP10/P24, ANI7/AMP20/P27端子機能の設定

ADPCレジスタ	PM2, PM15レジスタ	OAENnビット	ADSレジスタ	ANI4/AMP10/P24, ANI7/AMP20/P27端子機能の設定
デジタル入出力選択	入力モード	0	-	デジタル入力
		1	-	設定禁止
	出力モード	0	-	デジタル出力
		1	-	設定禁止
アナログ入力選択	入力モード	0	ANI選択	アナログ入力 (A/D変換対象)
			ANI非選択	アナログ入力 (A/D変換非対象)
		1	ANI選択	設定禁止
			ANI非選択	オペアンプ入力
	出力モード	-	-	設定禁止

備考1. n = 1, 2

2. 表の赤字部分がサンプル・プログラムでの設定値となります。

**注意** オペアンプ使用時には ,AMPn+, AMPn-, AMPnO端子を使用するため ,端子に兼用するアナログ入力機能は使用できません。ただし ,オペアンプ出力信号をアナログ入力として使用することはできます。

## 4.5 D/Aコンバータの設定

D/Aコンバータは、次の3種類のレジスタを使用します。

- ・周辺イネーブル・レジスタ0 (PER0)
- ・D/Aコンバータ・モード・レジスタ (DAM)
- ・D/A変換値設定レジスタW0 (DACSW0)

### 【音声データの再生を行う場合のD/Aコンバータの設定手順例】

周辺イネーブル・レジスタ0 (PER0) のビット6 (DACEN) をセット (1) (4.1参照)

D/Aコンバータ・モード・レジスタ (DAM) によりD/Aコンバータのチャンネル0の動作モードをリアルタイム出力モード、分解能を12ビット、D/Aコンバータの+側の基準電圧源を $V_{REFOUT}$ に設定

D/Aコンバータ・モード・レジスタ (DAM) のビット4 (DACE0) をセット (1) し、D/Aコンバータのチャンネル0のD/A変換動作を許可

D/A変換値設定レジスタW0 (DACSW0) にD/A変換値の初期値として、800H (P-P 2.0Vの1/2、**音声の振幅が0**) を設定

(1) D/Aコンバータ・モード・レジスタ (DAM)

D/Aコンバータの動作を制御するレジスタです。

DAMは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

図4 - 12 D/Aコンバータ・モード・レジスタ (DAM) のフォーマット



- 注1. D/Aコンバータの基準電圧は、2チャンネル共通なので、チャンネルごとに別々に設定できません。
2. ボルテージ・リファレンスの出力電圧をD/Aコンバータの+側の基準電圧 (DA<sub>REFP</sub>) として使用する場合は、D/A変換動作設定前に、ボルテージ・リファレンスの動作を開始してください (78K0R/Lx3 ユーザーズ・マニュアルを参照)。また、D/A変換動作中に、ボルテージ・リファレンスの設定を変更しないでください。

- 備考1. 図の赤字部分がサンプル・プログラムでの設定値となります。
2. DAREF、VRSELとVRONの設定により、D/Aコンバータの+側の基準電圧源は次のようになります。

表4 - 3 DAREF、VRSELとVRONの設定

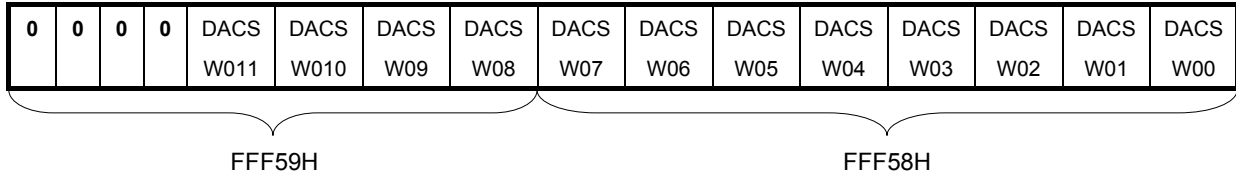
DAREF	VRSEL	VRON	D/Aコンバータの+側の基準電圧 (DA <sub>REFP</sub> )
0	x	x	AV <sub>DD1</sub>
1	0	0	AV <sub>REFP</sub>
1	1	1	V <sub>REFOUT</sub>

x : don't care

(2) D/A変換値設定レジスタW0 (DACSW0)

D/Aコンバータを使用する場合、ANO0端子に出力するアナログ電圧値を設定するレジスタです。  
DACSW0, DACSW1は、16ビット・メモリ操作命令で設定します。

図4 - 13 D/A変換値設定レジスタW0 (DACSW0) のフォーマット



**注意** A/Dコンバータの+側の基準電圧 (AD<sub>REFP</sub>) とD/Aコンバータの+側の基準電圧 (DA<sub>REFP</sub>) の両方に、ボルテージ・リファレンス出力 (V<sub>REFOUT</sub>) を使用する (VRSEL = 1かつDAREF = 1) 場合、A/D変換中にD/A変換値設定レジスタWn (DACSWn) を書き換えしないでください。DACSWnレジスタを書き換える場合は、A/D変換停止時 (ADCS = 0) に行ってください。

**備考1.** 分解能とD/Aコンバータのアナログ出力電圧 (V<sub>ANOn</sub>) の関係は、次のようになります。

・ 8ビット分解能 (DARES<sub>n</sub> = 0) :

$$V_{ANOn} = \text{D/Aコンバータ用基準電圧} \times (\text{DACSWn7-DACSWn0}) / 256$$

・ 12ビット分解能 (DARES<sub>n</sub> = 1) :

$$V_{ANOn} = \text{D/Aコンバータ用基準電圧} \times (\text{DACSWn11-DACSWn0}) / 4096$$

2. n = 0, 1



## 4.6 ソフトウェア記述例

ソフトウェアでの記述例として、サンプル・プログラムで行うタイマ・アレイ・ユニット0, ボルテージ・リファレンス, オペアンプ, およびD/Aコンバータの設定を以下に示します。

### (1) アセンブリ言語

#### タイマ・アレイ・ユニット0の設定

```

SINITTAU0:
    MOV TPS0L, #00000010B ; Timer Clock Select Register 0
    ... (略) ...

    ; CH0:for timing
    MOVW AX, #0000000000000000B ; Timer Mode Register 00
    MOVW TMR00, AX
    ... (略) ...

    ; CK00 = fCLK/2^3 = 5MHz -> 10ms
    MOVW TDR00, #(50000 - 1) ; set interval about 10ms
    SET1 TMMK00 ; set interrupt mask

    ; CH4:for play back sampling timing
    MOVW AX, #1000000000000000B ; Timer Mode Register 04
    MOVW TMR04, AX
    ... (略) ...

    ; CK01 = fCLK/2 = 20MHz -> 8kHz
    MOVW TDR04, #(2500 - 1) ; set interval about 125us (= 8kHz)
    SET1 TMMK04 ; set interrupt mask
    
```

CK00を $f_{CLK}/2^2$ , CK01を $f_{CLK}$ に設定

チャンネル0の動作クロックをCK00, 動作モードをインターバル・タイマ・モードに設定

チャンネル0のインターバルを約10msに設定

チャンネル0のタイマ割り込みをマスク設定

チャンネル4の動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定

チャンネル4のインターバルを125us (8kHz) に設定

チャンネル4のタイマ割り込みをマスク設定

チャンネル0の設定

チャンネル4の設定

ボルテージ・リファレンスの設定

```

SINITVR:
  MOV      ADVRC, #00001000B ; Analog-to-digital converter control register
  ; D/Aコンバータの基準電圧源をVREFOUT
  ; (ボルテージ・リファレンス出力)に設定

  ... (略) ...

  SET1     ADVRC.0 ; Enables operation
  ; ボルテージ・リファレンスの
  ; 出力電圧に2.0Vを選択

  SET1     ADVRC.1 ; Enables operation
  ; ボルテージ・リファレンスの
  ; 動作許可

  ; wait for settling time to 20ms (over 17msec)

  MOV      B, #2 ; set counter
  SET1     TSOL.0 ; start TAU0 CH0

JINIVR100:
  CLR1     TMIF00 ; clear interrupt request flag

JINIVR200:
  NOP
  BF       TMIF00, $JINIVR200 ;
  ;
  DEC      B
  BNZ     $JINIVR100 ; No

  CLR1     TMIF00 ; clear interrupt request flag
  SET1     TTOL.0 ; stop TAU0 CH0
  
```

D/Aコンバータの基準電圧源をV<sub>REFOUT</sub> (ボルテージ・リファレンス出力)に設定

ボルテージ・リファレンスの出力電圧に2.0Vを選択

ボルテージ・リファレンスの動作許可

ボルテージ・リファレンスの動作安定待ちとして約20msウェイト

オペアンプの設定

```

SINITPORT:
... (略) ...
MOV PM2, #11111111B ; Set P20-P27 to input port
... (略) ...
MOV PM15, #11111111B ; Set P150-P152,P157 to input port
... (略) ...
SINITAMP:
MOV ADPC, #0000000B ; A/D Port Configuration Register
... (略) ...
CLR1 OAC.1 ; Operational amplifier (AMP1) disable
CLR1 OAC.2 ; Operational amplifier (AMP2) disable
... (略) ...
SET1 OAC.1 ; Operational amplifier (AMP1) enable
SET1 OAC.2 ; Operational amplifier (AMP2) enable
; Use software to wait until the operational amplifier stabilizes (20us(max.))
MOV B, #80
JPDAC150:
DEC B
BNZ $JPDAC150 ; 20us elapsed ? , No
    
```

P23-P27, およびP150を  
 オペアンプの入出力用に  
 入力ポートに設定

オペアンプの入出力として使用する  
 端子(AMP1-,AMP1+,AMP10,  
 AMP2-,AMP2+,AMP20)を,ア  
 ナログ入力に設定

初期値として  
 オペアンプ1,2の  
 動作を停止

オペアンプを使用する際に  
 オペアンプ1,2の  
 動作を許可

オペアンプの動作安定待ち  
 のため約20μsウエイト

D/Aコンバータの設定

```

SINITDAC:
MOV DAM, #01000101B
... (略) ...
MOVW DACSW0, #0800H ; set initial data
    
```

D/Aコンバータのチャンネル0の  
 動作モードをリアルタイム出力モード,  
 分解能を12ビット,  
 D/Aコンバータの+側基準電圧源を  
 V<sub>REFOUT</sub>に設定

ANO0から出力する  
 アナログ電圧の初期値を設定

(2) C言語

タイマ・アレイ・ユニット0の設定

```

static void fn_InitTau0(void)
{
    TPS0L = 0b00000010; /* Timer Clock Select Register 0 */
    ... (略) ...

    /* CH0:for timing */
    TMR00 = 0b0000000000000000; /* TMR00 */
    ... (略) ...
    /* CK00 = fCLK/2^3 = 5MHz -> 10ms = 0.2[us/cclk] * 50000[count] */
    TDR00 = (50000 - 1); /* set interval time to 10 ms */
    TMMK00 = 1; /* disable timer interrupt */

    /* CH4:for play back sampling timing */
    TMR04 = 0b1000000000000000; /* TMR04 */
    ... (略) ...
    /* CK01 = fCLK/2 = 20MHz -> 125us (8kHz) = 0.05[us/cclk] * 2500[count] */
    TDR04 = (2500 - 1); /* set interval time to about 125us (= 8kHz) */
    TMMK04 = 1; /* disable timer interrupt */
}
    
```

CK00を $f_{CLK}/2^2$ , CK01を $f_{CLK}$ に設定

チャンネル0の動作クロックをCK00, 動作モードをインターバル・タイマ・モードに設定

チャンネル0のインターバルを約10msに設定

チャンネル0のタイマ割り込みをマスク設定

チャンネル0の設定

チャンネル4の動作クロックをCK01, 動作モードをインターバル・タイマ・モードに設定

チャンネル4のインターバルを125us (8kHz) に設定

チャンネル4のタイマ割り込みをマスク設定

チャンネル4の設定

## ボルテージ・リファレンスの設定

```
static void fn_InitVr(void)
{
    ADVRC = 0b00001000; /* Analog reference voltage control register */
    ... (略) ...
    ADVRC.0 = 1;
    ADVRC.1 = 1;

    /* wait for settling time to 20ms (over 17msec) */
    TSOL.0 = 1; /* start TAU0 CH0 */
    for(work = 2; work > 0; work--){ /* wait 10msec*2 */
        TMIF00 = 0; /* clear interrupt request flag */
        while(!TMIF00){
            NOP(); /* wait 10msec */
        }
    }
    TMIF00 = 0; /* clear interrupt request flag */
    TTOL.0 = 1; /* stop TAU0 CH0 */
}
}
```

D/Aコンバータの基準電圧源を $V_{REFOUT}$  (ボルテージ・リファレンス出力)に設定

ボルテージ・リファレンスの出力電圧に2.0Vを選択

ボルテージ・リファレンスの動作許可

ボルテージ・リファレンスの動作安定待ちとして約20msウエイト

オペアンプの設定

```

static void  fn_InitPort(void)
{
... (略) ...

    PM2 = 0b11111111; /* Set P20-P27 to input port */

... (略) ...

    PM15 = 0b11111111; /* Set P150-P152,P157 to input port */

... (略) ...

static void  fn_InitAmp(void)
{
    ADPC = 0b00000000; /* A/D Port

... (略) ...

    OAC.1 = 0; /* operation amplifier (AMP1) disable */
    OAC.2 = 0; /* operation amplifier (AMP2) disable */
}

... (略) ...

    OAC.1 = 1; /* operation amplifier (AMP1) enable */
    OAC.2 = 1; /* operation amplifier (AMP2) enable */

    /* Use software to wait until the operational amplifier stabilizes (20us(max.)) */
    for(loop = 40; loop > 0; loop--){
        NOP();
    }
}

```

P23-P27, およびP150を  
オペアンプの入出力用に  
入力ポートに設定

オペアンプの入出力として使用する  
端子(AMP1-,AMP1+,AMP1O,  
AMP2-,AMP2+,AMP2O)を,アナ  
ログ入力に設定

初期値として  
オペアンプ1,2の  
動作を停止

オペアンプを使用する際に  
オペアンプ1,2の  
動作を許可

オペアンプの動作安定待ち  
のため約20μsウエイト

## D/Aコンバータの設定

```
static void  fn_InitDac(void)
{
  DAM = 0b01000101; /*
  ... (略) ...
  DACSW0 = 0x0800; /*
}
```

D/Aコンバータのチャンネル0の  
動作モードをリアルタイム出力モード、  
分解能を12ビット、  
D/Aコンバータの+側基準電圧源をV<sub>REFOUT</sub>に設定

ANO0から出力する  
アナログ電圧の初期値を設定

## 4.7 再生処理について

再生処理について説明します。

アセンブリ言語の再生処理では、次の動作を行います。

再生準備として、ADPCMライブラリの初期化、変数の初期化、ハードウェアの動作許可を行います。

- (a) ROM領域に定義してあるADPCMデータ・テーブル（「TPLAYDATA」テーブル）の先頭アドレスを、ADPCMデータの読み出し位置に設定
- (b) ADPCMライブラリ関数を使用するための初期化処理呼び出し
- (c) オペアンプ1, 2の動作許可
- (d) D/Aコンバータの動作許可。
- (e) タイマ・アレイ・ユニット0のチャンネル4の動作許可
- (f) ADPCMデータの読み出しカウンタを初期化

ADPCMデータを読み出し、下位4ビットを伸長、補正し、D/Aコンバータで出力します。

- (a) ADPCMデータを読み出し、下位4ビットをADPCMライブラリ関数にて符号付き16ビットに伸張
  - (b) 符号付き16ビットに伸張したデータに8000Hを足し、符号なし16ビットのデータに補正
  - (c) 符号なし16ビットに補正したデータを2回右シフトし、符号なし12ビットのデータに補正
  - (d) 符号なし12ビットに補正したデータをD/A変換値設定レジスタW0（DACSW0）に設定
  - (e) D/A変換値設定レジスタW0（DACSW0）設定したデータがD/Aコンバータで出力されるまでウェイト
- ADPCMデータを読み出し、上位4ビットを伸長、補正し、D/Aコンバータで出力します。処理の詳細は、の(a)から(e)の処理と同様です。

ADPCMデータの読み出しカウンタと、ADPCMデータの読み出し位置を更新します。

ROM領域に定義してあるADPCMデータを全て出力するまで、 から までの処理を繰り返します。

再生終了処理として、ハードウェアの動作を停止させます。

- (a) タイマ・アレイ・ユニット0のチャンネル4の動作停止
- (b) D/Aコンバータの動作停止
- (c) オペアンプ1, 2の動作停止

再生するADPCMデータは、32kbps（サンプリング周波数:8kHz、量子化ビット数:4ビット）です。アセンブリ言語のADPCMデータ・テーブル（「TPLAYDATA」テーブル）とADPCMデータ・テーブルのサイズは「data\_playrom.asm」に定義していますので、外部参照してください。

ADPCMデータ・テーブルの最終データは、D/Aコンバータから出力する音声の振幅が0となるよう、上位4ビットを0にしてください。

ADPCMライブラリ関数を使用するには、ADPCMライブラリ関数の外部参照が必要です。以下のように関数名の先頭に「\_」をつけて外部参照をしてください。

```
EXTRN    _adpcm_init
EXTRN    _adpcm_l32_dec
```

ADPCMライブラリ関数を使用する際は、C言語ルーチンの呼び出しとなります。アセンブリ言語からのC言語ルーチンの呼び出し方法については、「CC78K0R Cコンパイラ 言語編 ユーザーズ・マニュアル」を参照してください。



ADPCMライブラリ関数の伸長処理で乗除算器を乗算モードで使用しますので、伸長処理を呼び出す直前に乗除算器の演算モードを乗算モードに設定してください。本プログラムで使用するADPCMライブラリは78K0R/Kx3向けですので、伸長処理の中で演算モードの切り替えをしていません。

ADPCMライブラリ関数の詳細は「ADPCM-SP 音声圧縮 / 伸長ソフトウェア・パッケージ 78K0Rマイクロコントローラ編 ユーザーズ・マニュアル」を参照してください。

```

SPLAYDAC:
    ;-----;
    ;   prepare for playing   ;
    ;-----;
(a) MOVW   HL,    #LOWW TPLAYDATA ; set start playing addr (low 16bit)
    MOV    ES,    #HIGHW TPLAYDATA; set start playing addr (high 4bit)

JPDAC100:
(b) MOVW   AX,    #LOWW RADPCMWORK
    CALL   !!_adpcm_init          ; ADPCM process Initialization

    ; operational amplifier setting
(c) SET1   OAC.1          ; operational amplifier (AMP1) enable
    SET1   OAC.2          ; operational amplifier (AMP2) enable
    ; Use software to wait until the operational amplifier stabilizes (20us (max.))
    MOV    B,    #80

JPDAC150:
    DEC    B
    BNZ    $JPDAC150           ; 20us elapsed ? , No

    ; D/A converter setting
(d) SET1   DACE0          ; D/A converter CH0 enable

    ; TAU0 CH4 setting for output timing
(e) CLR1   TMIF04          ; clear interrupt request flag
    SET1   TSOL.4         ; start TAU0 CH4

    ;*****;
    ;*-----* ;
    ;*  decode and play PCM data  * ;
    ;*-----* ;
    ;*****;
(f) MOVW   RPLAYCOUNT,#0    ; clear output data counter

JPDAC200:
    MOVW   AX,    RPLAYCOUNT ; get number of output times
    CMPW   AX,    !TPLAYSIZE  ; finished all data output ?
    BNC    $JPDAC300         ; Yes
    
```

```

;*****;
;* play low 4bit *;
;*****;
; decompression of ADPCM data (low 4bit -> 16bit)
(a) MOVW  AX,    #LOWW RADPCMWORK    ; set work area for _adpcm_132_dec
    PUSH  AX                      ; push argument
    MOV   A,    ES:[HL]            ; get compressed data
    AND  A,    #00FH              ; clear high 4bit
    MOV  X,    A
    CLRB  A
    CLR1  DIVMODE                ; set multiplication mode (for _adpcm_132_dec)
    CALL  !!_adpcm_132_dec        ; decompression of PCM data
    POP  AX                      ; pop argument
    MOVW  AX,    BC                ; get decompression data

; adjust play data
(b) ADDW  AX,    #8000H            ; adjust sign
(c) SHRW  AX,    (16-12)          ; right-align data

JPDAC220:
(d) MOVW  DACSW0, AX              ; set play data

; waiting for the output to be completed
(e) JPDAC230:
    NOP
    BF    TMIF04, $JPDAC230      ; the output to be completed ? , No
    CLR1  TMIF04                  ; clear interrupt request flag

;*****;
;* play high 4bit *;
;*****;
; decompression of ADPCM data (high 4bit -> 16bit)
(a) MOVW  AX,    #LOWW RADPCMWORK    ; set work area for _adpcm_132_dec
    PUSH  AX                      ; push argument
    MOV   A,    ES:[HL]            ; get compressed data
    SHR  A,    4
    MOV  X,    A
    CLRB  A
    CLR1  DIVMODE                ; set multiplication mode (for _adpcm_132_dec)
    CALL  !!_adpcm_132_dec        ; decompression of PCM data
    POP  AX                      ; pop argument
    MOVW  AX,    BC                ; get decompression data

; adjust play data
(b) ADDW  AX,    #8000H            ; adjust sign
(c) SHRW  AX,    (16-12)          ; right-align data

JPDAC270:
(d) MOVW  DACSW0, AX              ; set play data

; waiting for the output to be completed
(e) JPDAC280:
    NOP
    BF    TMIF04, $JPDAC280      ; the output to be completed ? , No
    CLR1  TMIF04                  ; clear interrupt request flag

```

```
INCW  RPLAYCOUNT      ; update play counter
INCW  HL                ; next play data
BR    JPDAC200

JPDAC300:
;-----;
;    finish playing    ;
;-----;
(a) SET1  TTOL.4        ; stop TAU0 CH4
(b) CLR1  DACE0         ; D/A converter CH0 disable
(c) CLR1  OAC.1         ; operational amplifier (AMP1) disable
CLR1  OAC.2         ; operational amplifier (AMP2) disable

JPDAC900:
RET
```

C言語の再生処理も、アセンブリ言語と同様な動作を行います。

C言語のACPCMデータ・テーブル(「aPlayData」テーブル)とADPCMデータ・テーブルのサイズは「data\_playrom.c」に定義していますので、外部参照してください。

ADPCMライブラリ関数を使用するには、#includeによる「adpcm.h」の取り込みが必要です。

```
static void    fn_PlayDac(void)
{
    unsigned char *    pucPlayData;    /* start playing addr */
    unsigned short    ushPlayCount;    /* output data counter */
    unsigned short    ushData;        /* decompression data */
    unsigned short    loop;            /* waiting counter */

    /*-----*/
    /*    prepare for playing    */
    /*-----*/
    /* set play data addr and size */
    pucPlayData = aPlayData;    /* set start playing address */

    adpcm_init(ushAdpcmWork);    /* ADPCM process Initialization */

    /* operational amplifier setting */
    OAC.1 = 1;    /* operational amplifier (AMP1) enable */
    OAC.2 = 1;    /* operational amplifier (AMP2) enable */
    /*Use software to wait until the operational amplifier stabilizes (20us(max.))*/
    for(loop = 40; loop > 0; loop--){
        NOP();
    }

    /* D/A converter setting */
    DACE0 = 1;    /* D/A converter CH0 enable */

    /* TAU0 CH4 setting for output timing */
    TMIF04 = 0;    /* clear interrupt request flag */
    TSOL.4 = 1;    /* start TAU0 CH4 */
}
```

```

/*****
/*-----*/
/*   decode and play PCM data   */
/*-----*/
/*****
for(ushPlayCount = 0; ushPlayCount < ushDataSize; ushPlayCount++){
  /***/
  /* play low 4bit */
  /***/
  /* decompression of ADPCM data (low 4bit -> 16bit)*/
  DIVMODE = 0; /* set multiplication mode (for _adpcm_132_dec) */
  ushData = (unsigned short)adpcm_132_dec(pucPlayData[ushPlayCount] & 0x0f,
ushAdpcmWork);

  /* adjust sign & right-align data */
  ushData = (unsigned short)((ushData + 0x8000) >> (16-12));

  /* set play data */
  DACSW0 = ushData;

  /* waiting for the output to be completed */
  while(!TMIF04){
    NOP();
  }
  TMIF04 = 0; /* clear interrupt request flag */

  /***/
  /* play high 4bit */
  /***/
  /* decompression of ADPCM data (high 4bit -> 16bit) */
  DIVMODE = 0; /* set multiplication mode (for _adpcm_132_dec) */
  ushData = (unsigned short)adpcm_132_dec((pucPlayData[ushPlayCount] >> 4)
& 0x0f, ushAdpcmWork);

  /* adjust sign & right-align data */
  ushData = (unsigned short)((ushData + 0x8000) >> (16-12));

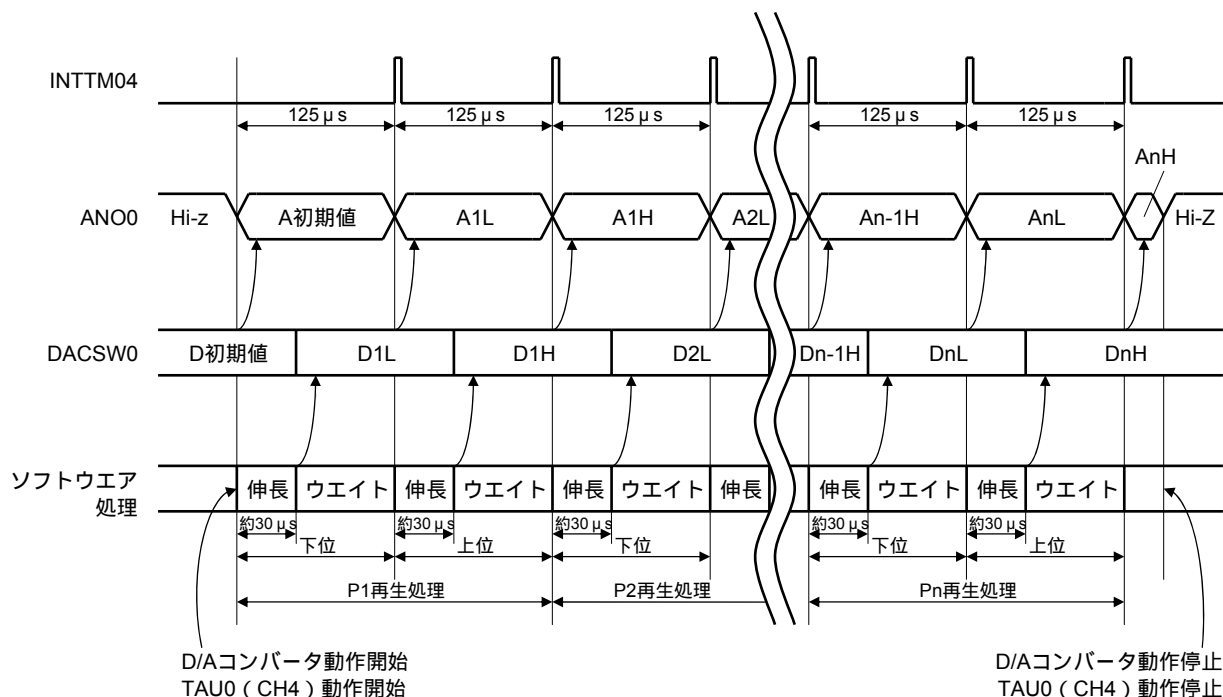
  /* set play data */
  DACSW0 = ushData;

  /* waiting for the output to be completed */
  while(!TMIF04){
    NOP();
  }
  TMIF04 = 0; /* clear interrupt request flag */
}

/*-----*/
/*   finish playing   */
/*-----*/
TT0L.4 = 1; /* stop TAU0 CH4 */
DACE0 = 0; /* D/A converter CH0 disable */
OAC.1 = 0; /* operational amplifier (AMP1) disable */
OAC.2 = 0; /* operational amplifier (AMP2) disable */
}

```

再生処理のタイミングチャートは以下のようになります。



**注意** ADPCMデータ・テーブルの最終データは、D/Aコンバータから出力する音声の振幅が0となるよう、上位4ビットを0にしてください。

**備考1.** n = ADPCMデータ・テーブルのサイズ

2. Pm : ADPCMデータ・テーブルの1バイトのADPCMデータ

3. D初期値 : 800H

DmL : Pmの下位4ビットを伸長、補正した値

DmH : Pmの上位4ビットを伸長、補正した値

4. A初期値 : DACSW0の設定値がD初期値 (800H) の時のアナログ出力電圧

Amk : DACSW0の設定値がDmkの時のアナログ出力電圧

5. m = 1 ~ ADPCMデータ・テーブルのサイズ

k = L, H

## 第5章 関連資料

資料名		和文 / 英文
78K0R/Lx3 ユーザーズ・マニュアル		PDF
78K0Rシリーズ 命令編 ユーザーズ・マニュアル		PDF
RA78K0R アセンブラ・パッケージ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
CC78K0R Cコンパイラ ユーザーズ・マニュアル	言語編	PDF
	操作編	PDF
ADPCM-SP 音声圧縮 / 伸長ソフトウェア・パッケージ 78K0Rマイクロコントローラ編 ユーザーズ・マニュアル		PDF
PM+ プロジェクト・マネージャ ユーザーズ・マニュアル		PDF

## 付録A プログラム・リスト

プログラム・リスト例として、ソース・プログラムを次に示します。なお、ADPCMデータ・テーブルを定義している「data\_playrom.asm」と「data\_playrom.c」のソース・プログラムについては、掲載を省略します。

```
main.asm (アセンブリ言語版)
;
; Copyright (C) NEC Electronics Corporation 2006
; NEC ELECTRONICS CONFIDENTIAL AND PROPRIETARY
; All rights reserved by NEC Electronics Corporation.
; This program must be used solely for the purpose for which
; it was furnished by NEC Electronics Corporation. No part of this
; program may be reproduced or disclosed to others, in any
; form, without the prior written permission of NEC Electronics
; Corporation. Use of copyright notice dose not evidence
; publication of the program.
;
;-----;
; Extern function ;
;-----;
EXTRN  _adpcm_init          ; ADPCM process initialize
EXTRN  _adpcm_l32_dec       ; ADPCM data decode
;-----;
; Extern variables/constants ;
;-----;
EXTRN  TPLAYDATA           ; sound data
EXTRN  TPLAYSIZE           ; size of sound data
;-----;
; Vector table initialize ;
;-----;
TVCT1  CSEG      AT      000000H
        DW      IRESET          ; (00H) RESET, POC, LVI, WDT, TRAP
TVCT2  CSEG      AT      000004H
        DW      IRESET          ; (04H) INTWDTI
        DW      IRESET          ; (06H) INTLVI
        DW      IRESET          ; (08H) INTPO
        DW      IRESET          ; (0AH) INTPI
```



DW	IRESET		; (0CH) INTP2
DW	IRESET		; (0EH) INTP3E
DW	IRESET		; (10H) INTP4
DW	IRESET		; (12H) INTP5
DW	IRESET		; (14H) INTST3
DW	IRESET		; (16H) INTSR3
DW	IRESET		; (18H) INTSRE3
DW	IRESET		; (1AH) INTDMA0
DW	IRESET		; (1CH) INTDMA1
DW	IRESET		; (1EH) INTST0, INTCSI00
DW	IRESET		; (20H) INTSR0, INTCSI01
DW	IRESET		; (22H) INTSRE0
DW	IRESET		; (24H) INTST1, INTCSI10, INTIIC10
DW	IRESET		; (26H) INTSR1
DW	IRESET		; (28H) INTSRE1
DW	IRESET		; (2AH) INTIICA
DW	IRESET		; (2CH) INTTM00
DW	IRESET		; (2EH) INTTM01
DW	IRESET		; (30H) INTTM02
DW	IRESET		; (32H) INTTM03
DW	IRESET		; (34H) INTAD
DW	IRESET		; (36H) INTRTC
DW	IRESET		; (38H) INTRTCI
DW	IRESET		; (3AH) INTKR
DW	IRESET		; (3CH) INTST2, INTCSI20, INTIIC20
DW	IRESET		; (3EH) INSR2
DW	IRESET		; (40H) INTSRE2
DW	IRESET		; (42H) INTTM04
DW	IRESET		; (44H) INTTM05
DW	IRESET		; (46H) INTTM06
DW	IRESET		; (48H) INTTM07
DW	IRESET		; (4AH) INTP6
DW	IRESET		; (4CH) INTP7
DW	IRESET		; (4EH) INTP8
DW	IRESET		; (50H) INTP9
DW	IRESET		; (52H) INTP10
DW	IRESET		; (54H) INTP11
DW	IRESET		; (56H) INTTM10
DW	IRESET		; (58H) INTTM11
DW	IRESET		; (5AH) INTTM12
DW	IRESET		; (5CH) INTTM13
DW	IRESET		; (5EH) INTMD
TVCT3	CSEG	AT	00007EH

```

        DW      IRESET          ;(7EH) BRK

;-----;
; Stack area definition          ;
;-----;
DSTK   DSEG   IHRAM          ;Stack Area address
STACKEND:
        DS      60H
STACKTOP:

;-----;
; Local constants                ;
;-----;
;-----;
; Global variables                ;
;-----;
;-----;
; Local variables                ;
;-----;
DPMAIN DSEG   SADDRP
        RADPCMWORK:   DS      32      ; Work area for ADPCM process
        RPLAYCOUNT: DS      2       ; Play data counter

;-----;
; Code                            ;
;-----;
XMAIN  CSEG   UNIT
;-----;
; Hardware initialization          ;
;-----;
        PUBLIC  IRESET
IRESET:
;-----;
;      Disable all interrupts      ;
;-----;
DI
;-----;
;      Set register bank          ;
;-----;
SEL    RBO          ; use register bank 0
;-----;

```

```

;          Set stack pointer          ;
;-----;
MOVW      SP,      #LOWW STACKTOP

;-----;
;          Initialization of port     ;
;-----;
CALL      !!SINITPORT

;-----;
;          Low-voltage detection      ;
;-----;
CALL      !!SINITLVI          ; ensures 2.7V to VDD

;-----;
;          Initialization of clock    ;
;-----;
MOV       CMC,     #01000011B      ; Clock Operation Mode Control Register
;|||||+--- : Control of high-speed system clock oscillation frequency
;||||| : 0 : 2 MHz <= fMX <= 10 MHz
;||||| : 1 : 10 MHz < fMX <= 20 MHz
;|||||
;|||||+---- : XT1 oscillator oscillation mode selection
;|||| : 0 0 : Low-consumption oscillation
;|||| : 0 1 : Normal oscillation
;|||| : 1 x : Super-low-consumption oscillation
;|||| x = don't care
;||||
;||||+----- : Be sure to set 0
;||||
;|||+----- : [1] Subsystem clock pin operation mode
;|||          [2] XT1/P123 pin and XT2/P124 pin
;||| : 0 : [1]Input port mode
;|||          [2]Input port
;|||
;||| : 1 : [1]XT1 oscillation mode
;|||          [2]Crystal resonator connection
;|||
;||+----- : Be sure to set 0
;||
;++----- : [1]EXCLK OSCSEL High-speed system clock pin operation mode
;          [2]X1/P121 pin
;          [3]X2/EXCLK/P122 pin

```

```

; : 0 0 : [1]Input port mode
;           [2][3]Input port
;
; : 0 1 : [1]X1 oscillation mode
;           [2][3]Crystal/ceramic resonator connection
;
; : 1 0 : [1]Input port mode
;           [2][3]Input port
;
; : 1 1 : [1]External clock input mode
;           [2]Input port
;           [3]External clock input

CLR1    MSTOP                ; X1 oscillator operating

MOV     OSMC, #00000001B      ; Operation Speed Mode Control Register
;|||||++--- : fCLK frequency selection
;||||| : 0 0 : Operates at a frequency of 10 MHz or less.
;||||| : 0 1 : Operates at a frequency higher than 10 MHz.
;||||| : 1 0 : Operates at a frequency of 1 MHz.
;|||||
;|+++++----- : Be sure to set 00000
;|
;+----- : Setting in subsystem clock HALT mode
; : 0 : Enables subsystem clock supply to peripheral functions.
;       (See Table 21-1 Operating Statuses in HALT Mode (2/3)
;       for the peripheral functions whose operations are enabled.)
; : 1 : Stops subsystem clock supply to peripheral functions
;       except real-time counter, clock output/buzzer output,
;       and LCD controller/driver.

BF      OSTC.0, $$           ; X1 oscillation stabilization finished ? , No

;*- Caution -----*;
;* To increase fCLK to 10 MHz or higher, set FSEL to '1', *;
;* then change fCLK after two or more clocks have elapsed. *;
;*------*;
NOP
NOP

MOV     CKC, #00010000B      ; System Clock Control Register
;|+|+++++--- : Selection of CPU/peripheral hardware clock (fCLK)
;| | : 0 0 x 0 0 0 : fIH

```

```

;| | : 0 0 x 0 0 1 : fIH/2 (default)
;| | : 0 0 x 0 1 0 : fIH/2^2
;| | : 0 0 x 0 1 1 : fIH/2^3
;| | : 0 0 x 1 0 0 : fIH/2^4
;| | : 0 0 x 1 0 1 : fIH/2^5
;| | : 0 1 x 0 0 0 : fMX
;| | : 0 1 x 0 0 1 : fMX/2
;| | : 0 1 x 0 1 0 : fMX/2^2
;| | : 0 1 x 0 1 1 : fMX/2^3
;| | : 0 1 x 1 0 0 : fMX/2^4
;| | : 0 1 x 1 0 1 : fMX/2^5
;| | : 1 x 0 x x x : fSUB
;| | : 1 x 1 x x x : fSUB/2
;| | : Other than above : Setting prohibited
;| | x = don't care
;| |
;| +----- : Status of Main system clock (fMAIN)
;| : 0 : Internal high-speed oscillation clock (fIH)
;| : 1 : High-speed system clock (fMX)
;|
;+----- : Status of CPU/peripheral hardware clock (fCLK)
; : 0 : Main system clock (fMAIN)
; : 1 : Subsystem clock (fSUB)

```

HRST100:; CPU is operating on a High-speed system clock ?

BT CLS, \$HRST100 ; No

BF MCS, \$HRST100 ; No

SET1 HI0STOP ; internal high-speed oscillation stopped

MOV OSTs, #00000111B ; Oscillation Stabilization Time Select Register

;|||+---- : Oscillation stabilization time selection

;||| : 0 0 0 : 2^8/fX

;||| : 0 0 1 : 2^9/fX

;||| : 0 1 0 : 2^10/fX

;||| : 0 1 1 : 2^11/fX

;||| : 1 0 0 : 2^13/fX

;||| : 1 0 1 : 2^15/fX

;||| : 1 1 0 : 2^17/fX

;||| : 1 1 1 : 2^18/fX

;|||

;+++----- : Be sure to set 0000

```

MOV     PER0,    #01100001B      ; Peripheral Enable Register 0
;|||||+--- : Control of timer array unit 0 input clock
;||||| : 0 : Stops input clock supply.
;|||||      * SFR used by timer array unit 0 cannot be written.
;|||||      * Timer array unit 0 is in the reset status.
;||||| : 1 : Supplies input clock.
;|||||      * SFR used by timer array unit 0 can be read and written.
;|||||
;|||||+---- : Control of timer array unit 1 input clock
;||||| : 0 : Stops input clock supply.
;|||||      * SFR used by timer array unit 1 cannot be written.
;|||||      * Timer array unit 1 is in the reset status.
;||||| : 1 : Supplies input clock.
;|||||      * SFR used by timer array unit 1 can be read and written.
;|||||
;||||+----- : Control of serial array unit 0 input clock
;|||| : 0 : Stops input clock supply.
;||||      * SFR used by the serial array unit 0 cannot be written.
;||||      * The serial array unit 0 is in the reset status.
;|||| : 1 : Supplies input clock.
;||||      * SFR used by the serial array unit 0 can be read and written.
;||||
;|||+----- : Control of serial array unit 1 input clock
;||| : 0 : Stops input clock supply.
;|||      * SFR used by the serial array unit 1 cannot be written.
;|||      * The serial array unit 1 is in the reset status.
;||| : 1 : Supplies input clock.
;|||      * SFR used by the serial array unit 1 can be read and written.
;|||
;|||+----- : Control of serial interface IICA input clock
;||| : 0 : Stops input clock supply.
;|||      * SFR used by the serial interface IICA cannot be written.
;|||      * The serial interface IICA is in the reset status.
;||| : 1 : Supplies input clock.
;|||      * SFR used by the serial interface IICA can be read and written.
;|||
;||+----- : Control of A/D converter, operational amplifier, and voltage
reference input clock
;|| : 0 : Stops input clock supply.
;||      * SFR used by the A/D converter, operational amplifier, and voltage
reference cannot be written.
;||      * The A/D converter, operational amplifier, and voltage reference
is in the reset status.

```

```

;|| : 1 : Supplies input clock.
;||      * SFR used by the A/D converter, operational amplifier, and voltage
reference can be read and written.
;||
;|+----- : Control of D/A converter input clock
;| : 0 : Stops input clock supply.
;|      * SFR used by D/A converter cannot be written.
;|      * The D/A converter is in the reset status.
;| : 1 : Supplies input clock.
;|      * SFR used by the D/A converter can be read and written.
;|
;|+----- : Control of real-time counter (RTC) input clock
;| : 0 : Stops input clock supply.
;|      * SFR used by the real-time counter (RTC) cannot be written.
;|      * The real-time counter (RTC) is in the reset status.
;| : 1 : Supplies input clock.
;|      * SFR used by the real-time counter (RTC) can be read and written.

```

```

;-----;
; Initialize of Key Interrupt Function ;
;-----;

```

CALL !!SINITKR

```

;-----;
; Initialization of timer ;
;-----;

```

CALL !!SINITTAU0

```

;-----;
; Initialization of voltage reference ;
;-----;

```

CALL !!SINITVR

```

;-----;
; Initialization of Operational amplifier ;
;-----;

```

CALL !!SINITAMP

```

;-----;
; Initialization of D/A Converter ;
;-----;

```

CALL !!SINITDAC

```

;=====;
;-----;
;           Main Loop           ;
;-----;
;=====;

MMAIN:
;*****;
;*                                     *;
;*   play melody when the key is input   *;
;*                                     *;
;*****;
;*-----* ;
;*   wait key input   * ;
;*-----* ;

CLR1    KRMK           ; clear key interrupt
CLR1    KRIF           ; clear key interrupt request flag
HALT                    ; sets the HALT mode
SET1    KRMK           ; set key interrupt
CLR1    KRIF           ; clear key interrupt request flag

;*-----* ;
;* remove key input noise * ;
;*-----* ;
BT      P7.0,    $LMAIN200 ; key input ? , No

SET1    TSOL.0                ; start TAU0 CHO
CLR1    TMIF00                ; clear interrupt request flag

LMAIN100:
NOP
BF      TMIF00,$LMAIN100    ; wait 10msec

CLR1    TMIF00                ; clear interrupt request flag
SET1    TTOL.0                ; stop TAU0 CHO

BT      P7.0,    $LMAIN200 ; key input ? , No
;*-----* ;
;*   play melody   * ;
;*-----* ;
CALL    !!SPLAYDAC

LMAIN200:
;*****;
;*                                     *;

```



```

;*   The main processing writes here   *;
;*           if there is something     *;
;*                                     *;
;*****;
;

BR      MMAIN          ; continue main process

;-----;
; Module:      SINITPORT          ;
; Description:  Setting of I/O ports          ;
;   parameter: --          ;
;   return   : --          ;
;-----;
SINITPORT:
;-----;
;   Setting of Port 0
;-----;
MOV     P0,      #00000000B      ; Set P00-P02 Output latch to Low
MOV     PM0,     #11111000B      ; Set P00-P02 to output port
; P00-P02:Unused

;-----;
;   Setting of Port 1
;-----;
MOV     P1,      #00000000B      ; Set P10-P17 Output latch to Low
MOV     PM1,     #00000000B      ; Set P10-P17 to output port
; P10-P15:Unused

;-----;
;   Setting of Port 2
;-----;
MOV     P2,      #00000000B      ; Set P20-P27 Output latch to Low
MOV     PM2,     #11111111B      ; Set P20-P27 to input port
; PM23:filter circuit(AMP1-)
; PM24:filter circuit(AMP10)
; PM25:filter circuit(AMP1+)
; PM26:filter circuit(AMP2-)
; PM27:filter circuit(AMP20)
; P20-P22:Unused

;-----;
;   Setting of Port 3
;-----;

```

```

MOV      P3,      #00001000B      ; Set P30-P32,P34 Output latch to Low
                                           ; Set P33 Output latch High
MOV      PM3,     #11110000B      ; Set P30-P34 to output port
                                           ; P30-P34:Unused

;-----
;      Setting of Port 4
;-----
MOV      P4,      #00000000B      ; Set P40-P41 Output latch to Low
MOV      PM4,     #11111100B      ; Set P40-P41 to output port
                                           ; P40-P41:Unused

;-----
;      Setting of Port 5
;-----
MOV      P5,      #00000000B      ; Set P50-P57 Output latch to Low
MOV      PM5,     #11111000B      ; Set P50-P57 to output port
                                           ; P50-P57:Unused

;-----
;      Setting of Port 6
;-----
MOV      P6,      #00000000B      ; Set P60-P61 Output latch to Low
MOV      PM6,     #11111100B      ; Set P60-P61 to output port
                                           ; P60-P61:Unused

;-----
;      Setting of Port 7
;-----
MOV      P7,      #00000000B      ; Set P70-P77 Output latch to Low
MOV      PM7,     #00000001B      ; et P70 to input port, P71-P77 to output port
MOV      PU7,     #00000001B      ; P70 on-chip pull-up resistor connected
                                           ; P70:key input port
                                           ; P71-P77:Unused

;-----
;      Setting of Port 8
;-----
MOV      P8,      #00000000B      ; Set P80-P88 Output latch to Low
MOV      PM8,     #00000000B      ; Set P80-P88 to output port
                                           ; P80-P88:Unused

;-----

```

```

;      Setting of Port 9
;-----
MOV     P9,      #00000000B      ; Set P90-P97 Output latch to Low
MOV     PM9,     #00000000B      ; Set P90-P97 to output port
                                           ; P90-P97:Unused

;-----
;      Setting of Port 10
;-----
MOV     P10,     #00000000B      ; Set P100-P102 Output latch to Low
MOV     PM10,    #11111000B      ; Set P100-P102 to output port
                                           ; P100-P102:Unused

;-----
;      Setting of Port 11
;-----
MOV     P11,     #00000000B      ; Set P110-P111 Output latch to Low
MOV     PM11,    #11111101B      ; Set P110 to input port, P111 to output port
                                           ; P110:play back output(AN00)
                                           ; P111:Unused

;-----
;      Setting of Port 12
;-----
MOV     P12,     #00000000B      ; Set P120 Output latch to Low
MOV     PM12,    #11111110B      ; Set P120 to output port
                                           ; P120-P124:Unused
                                           ; *P121-P124:Input port

;-----
;      Setting of Port 13
;-----
MOV     P13,     #00000000B      ; Set P130 Output latch to Low
                                           ; P130:Unused

;-----
;      Setting of Port 14
;-----
MOV     P14,     #00000000B      ; Set P140-P147 Output latch to Low
MOV     PM14,    #00000000B      ; Set P140-P147 to output port
                                           ; P140-P147:Unused
;-----

```

; Setting of Port 15

-----

```
MOV    P15,    #0000000B    ; Set P150-P152,P157 Output latch to Low
MOV    PM15,   #1111111B    ; Set P150-P152,P157 to input port
                                ; PM150:filter circuit(AMP2+)
                                ; P151-P152,P157:Unused
```

RET

-----;

```
; Module:      SINITLVI      ;
; Description: Ensures 2.7V to the power supply voltage ;
; parameter:  --            ;
; return   :  --            ;
```

-----;

SINITLVI:

; setting of Low-Voltage Detector

```
SET1   LVIMK                ; disable LVI interrupt
CLR1   LVISEL               ; detects level of VDD
MOV    LVIS,    #00001001B  ; Low-Voltage Detection Level Select Register
                                ; ||||+--- : Detection level
                                ; |||| : 0 0 0 0 : VLV10 (4.22V)
                                ; |||| : 0 0 0 1 : VLV11 (4.07V)
                                ; |||| : 0 0 1 0 : VLV12 (3.92V)
                                ; |||| : 0 0 1 1 : VLV13 (3.76V)
                                ; |||| : 0 1 0 0 : VLV14 (3.61V)
                                ; |||| : 0 1 0 1 : VLV15 (3.45V)
                                ; |||| : 0 1 1 0 : VLV16 (3.30V)
                                ; |||| : 0 1 1 1 : VLV17 (3.15V)
                                ; |||| : 1 0 0 0 : VLV18 (2.99V)
                                ; |||| : 1 0 0 1 : VLV19 (2.84V)
                                ; |||| : 1 0 1 0 : VLV110 (2.68V)
                                ; |||| : 1 0 1 1 : VLV111 (2.53V)
                                ; |||| : 1 1 0 0 : VLV112 (2.38V)
                                ; |||| : 1 1 0 1 : VLV113 (2.22V)
                                ; |||| : 1 1 1 0 : VLV114 (2.07V)
                                ; |||| : 1 1 1 1 : VLV115 (1.91V)
                                ; ||||
                                ; ++++----- : Be sure to set 0000
```

```
CLR1   LVIMD                ; generates an internal interrupt signal when detect the
low-voltage
```

```
SET1   LVION                ; enables low-voltage detection operation
```

```

; software to wait for the operation stabilization time (over 10us)
MOV     B,     #10           ; set counter
HRES100:
NOP                    ;                               (1clk)
DEC     B                    ;                               (1clk)
BNZ     $HRES100          ; finished waiting ? , No (2clk/4clk)

; wait for VDD to become VLVI or more
HRES300:
NOP
BT      LVIF,     $HRES300 ; VDD < VLVI? , Yes
CLR1    LVION          ; disables low-voltage detection operation

RET

;-----;
; Module:      SINITKR          ;
; Description:  Setting of Key Interrupt Function      ;
; parameter:  --                ;
; return :  --                ;
;-----;
SINITKR:
SET1    KRMK          ; disable key interrupt
MOV     KRM,     #00000001B ; Key Return Mode Register
;|||||+--- : KR0 interrupt mode control
;|||||+---- : KR1 interrupt mode control
;||||+----- : KR2 interrupt mode control
;|||+----- : KR3 interrupt mode control
;||+----- : KR4 interrupt mode control
;|+----- : KR5 interrupt mode control
;+----- : KR6 interrupt mode control
;+----- : KR7 interrupt mode control
; : 0 : Does not detect key interrupt signal
; : 1 : Detects key interrupt signal

NOP     ; 250ns interval from set KRM to clear KRIF
NOP
NOP
NOP
NOP
CLR1    KRIF          ; clear key interrupt request flag

RET

```

```

;-----;
; Module:      SINITTAU0                                ;
; Description:  Setting of Timer array unit 0           ;
;   parameter: --                                       ;
;   return  : --                                       ;
;-----;
SINITTAU0:
    MOV     TPSOL, #00000010B ; Timer Clock Select Register 0
           ;|||+--- : Selection of operation clock (CK00)
           ;+++----- : Selection of operation clock (CK01)
           ; : 0 0 0 0 : CK0m = fCLK
           ; : 0 0 0 1 : CK0m = fCLK/2
           ; : 0 0 1 0 : CK0m = fCLK/2^2
           ; : 0 0 1 1 : CK0m = fCLK/2^3
           ; : 0 1 0 0 : CK0m = fCLK/2^4
           ; : 0 1 0 1 : CK0m = fCLK/2^5
           ; : 0 1 1 0 : CK0m = fCLK/2^6
           ; : 0 1 1 1 : CK0m = fCLK/2^7
           ; : 1 0 0 0 : CK0m = fCLK/2^8
           ; : 1 0 0 1 : CK0m = fCLK/2^9
           ; : 1 0 1 0 : CK0m = fCLK/2^10
           ; : 1 0 1 1 : CK0m = fCLK/2^11
           ; : 1 1 0 0 : CK0m = fCLK/2^12
           ; : 1 1 0 1 : CK0m = fCLK/2^13
           ; : 1 1 1 0 : CK0m = fCLK/2^14
           ; : 1 1 1 1 : CK0m = fCLK/2^15
           ; m = 0, 1

; CH0:for timing
MOVW     AX, #0000000000000000B ; Timer Mode Register 00
MOVW     TMRO0,AX;|||+--- : [1]Operation mode of channel 0
           ;||| : [2]Count operation of TCR
           ;||| : [3]Independent operation
           ;||| : [4]Setting of starting counting and interrupt
           ;||| : 0 0 0 0 : [1]Interval timer mode
           ;||| : [2]Counting down
           ;||| : [3]Possible
           ;||| : [4]Timer interrupt is not generated
           ;||| when counting is started
           ;||| (timer output does not change, either).
           ;|||
           ;||| : 0 0 0 1 : [1]Interval timer mode
           ;||| : [2]Counting down

```

```

;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is generated
;||||||||| when counting is started
;||||||||| (timer output also changes).
;|||||||||
;||||||||| : 0 1 0 0 : [1]Capture mode
;||||||||| [2]Counting up
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).
;|||||||||
;||||||||| : 0 1 0 1 : [1]Capture mode
;||||||||| [2]Counting up
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is generated when
;||||||||| counting is started
;||||||||| (timer output also changes).
;|||||||||
;||||||||| : 0 1 1 0 : [1]Event counter mode
;||||||||| [2]Counting down
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).
;|||||||||
;||||||||| : 1 0 0 0 : [1]One-count mode
;||||||||| [2]Counting down
;||||||||| [3]Impossible
;||||||||| [4]Start trigger is invalid
;||||||||| during counting operation.
;||||||||| At that time, interrupt
;||||||||| is not generated, either.
;|||||||||
;||||||||| : 1 0 0 1 : [1]One-count mode
;||||||||| [2]Counting down
;||||||||| [3]Impossible
;||||||||| [4]Start trigger is valid
;||||||||| during counting operation.
;||||||||| At that time, interrupt
;||||||||| is also generated.
;|||||||||
;||||||||| : 1 1 0 0 : [1]Capture & one-count mode

```

```

;||||| [2]Counting up
;||||| [3]Possible
;||||| [4]Timer interrupt is not generated
;||||| when counting is started
;||||| (timer output does not change, either).
;||||| Start trigger is invalid
;||||| during counting operation.
;||||| At that time interrupt
;||||| is not generated, either.
;|||||
;||||| : Other than above : Setting prohibited
;|||||
;|||||++----- : Be sure to set 00
;|||||
;|||||++----- : Selection of TI00 pin input signal,
;||||| fSUB/2, fSUB/4, or INTRTC1 valid edge
;||||| (the timer input used with channel 0
;||||| is selected by using TIS0 register).
;||||| : 0 0 : Falling edge
;||||| : 0 1 : Rising edge
;||||| : 1 0 : Both edges (when low-level width is measured)
;||||| Start trigger: Falling edge, Capture trigger: Rising edge
;||||| : 1 1 : Both edges (when high-level width is measured)
;||||| Start trigger: Rising edge, Capture trigger: Falling edge
;|||||
;|||||+++----- : Setting of start trigger or capture trigger of channel
0
;||||| : 0 0 0 : Only software trigger start is valid
;||||| (other trigger sources are unselected).
;||||| : 0 0 1 : Valid edge of TI00 pin input signal, fSUB/2, fSUB/4, or INTRTC1
;||||| is used as both the start trigger and capture trigger.
;||||| : 0 1 0 : Both the edges of TI00 pin input signal, fSUB/2, fSUB/4, or
INTRTC1
;||||| are used as a start trigger and a capture trigger.
;||||| : 1 0 0 : Interrupt signal of the master channel is used
;||||| (when the channel is used as a slave channel
;||||| with the combination operation function).
;||||| : Other than above : Setting prohibited
;|||||
;|||||+----- : Selection of slave/master of channel 0
;||||| : 0 : Operates as slave channel with combination operation function.
;||||| : 1 : Operates as master channel with combination operation function.
;|||||

```

0

INTRTC1



```

;|||+----- : Selection of count clock (TCLK) of channel 0
;||| : 0 : Operation clock MCK specified by CKS00 bit
;||| : 1 : Valid edge of input signal input from T100 pin, fSUB/2, fSUB/4, or
INTRTC1
;||| (the timer input used with channel 0 is selected by using T1S0
register).
;|||
;|++----- : Be sure to set 00
;|
;+----- : Selection of operation clock (MCK) of channel 0
; : 0 : Operation clock CK00 set by TPS0 register
; : 1 : Operation clock CK01 set by TPS0 register
; CK00 = fCLK/2^3 = 5MHz -> 10ms = 0.2[us/cik] * 50000[count]
MOVW TDR00, #(50000 - 1) ; set interval time to 10 ms

SET1 TMMK00 ; disable interrupt

; CH4:for play back sampling timing
MOVW AX, #1000000000000000B ; Timer Mode Register 04
MOVW TMR04,AX;|||||||||++++--- : [1]Operation mode of channel 4
;||||||||| [2]Count operation of TCR
;||||||||| [3]Independent operation
;||||||||| [4]Setting of starting counting and interrupt
;||||||||| : 0 0 0 0 : [1]Interval timer mode
;||||||||| [2]Counting down
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).
;|||||||||
;||||||||| : 0 0 0 1 : [1]Interval timer mode
;||||||||| [2]Counting down
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is generated
;||||||||| when counting is started
;||||||||| (timer output also changes).
;|||||||||
;||||||||| : 0 1 0 0 : [1]Capture mode
;||||||||| [2]Counting up
;||||||||| [3]Possible
;||||||||| [4]Timer interrupt is not generated
;||||||||| when counting is started
;||||||||| (timer output does not change, either).

```

```

;|||||||
;||||||| : 0 1 0 1 : [1]Capture mode
;||||||| [2]Counting up
;||||||| [3]Possible
;||||||| [4]Timer interrupt is generated
;||||||| when counting is started
;||||||| (timer output also changes).
;|||||||
;||||||| : 0 1 1 0 : [1]Event counter mode
;||||||| [2]Counting down
;||||||| [3]Possible
;||||||| [4]Timer interrupt is not generated
;||||||| when counting is started
;||||||| (timer output does not change, either).
;|||||||
;||||||| : 1 0 0 0 : [1]One-count mode
;||||||| [2]Counting down
;||||||| [3]Impossible
;||||||| [4]Start trigger is invalid
;||||||| during counting operation.
;||||||| At that time, interrupt
;||||||| is not generated, either.
;|||||||
;||||||| : 1 0 0 1 : [1]One-count mode
;||||||| [2]Counting down
;||||||| [3]Impossible
;||||||| [4]Start trigger is valid
;||||||| during counting operation.
;||||||| At that time, interrupt
;||||||| is also generated.
;|||||||
;||||||| : 1 1 0 0 : [1]Capture & one-count mode
;||||||| [2]Counting up
;||||||| [3]Possible
;||||||| [4]Timer interrupt is not generated
;||||||| when counting is started
;||||||| (timer output does not change, either).
;||||||| Start trigger is invalid
;||||||| during counting operation.
;||||||| At that time interrupt
;||||||| is not generated, either.
;|||||||
;||||||| : Other than above : Setting prohibited

```

```

;|||||
;|||||++----- : Be sure to set 00
;|||||
;|||||++----- : Selection of TI04 pin input signal,
;|||||          fSUB/2, fSUB/4, or INTRTC1 valid edge
;|||||          (the timer input used with channel 4
;|||||          is selected by using TIS0 register).
;||||| : 0 0 : Falling edge
;||||| : 0 1 : Rising edge
;||||| : 1 0 : Both edges (when low-level width is measured)
;|||||          Start trigger: Falling edge, Capture trigger: Rising edge
;||||| : 1 1 : Both edges (when high-level width is measured)
;|||||          Start trigger: Rising edge, Capture trigger: Falling edge
;|||||
;|||||++----- : Setting of start trigger or capture trigger of channel
4
;||||| : 0 0 0 : Only software trigger start is valid
;|||||          (other trigger sources are unselected).
;||||| : 0 0 1 : Valid edge of TI04 pin input signal, fSUB/2, fSUB/4, or INTRTC1
;|||||          is used as both the start trigger and capture trigger.
;||||| : 0 1 0 : Both the edges of TI04 pin input signal, fSUB/2, fSUB/4, or
INTRTC1
;|||||          are used as a start trigger and a capture trigger.
;||||| : 1 0 0 : Interrupt signal of the master channel is used
;|||||          (when the channel is used as a slave channel
;|||||          with the combination operation function).
;||||| : Other than above : Setting prohibited
;|||||
;|||||+----- : Selection of slave/master of channel 4
;||||| : 0 : Operates as slave channel with combination operation function.
;||||| : 1 : Operates as master channel with combination operation function.
;|||||
;|||||+----- : Selection of count clock (TCLK) of channel 0
;||||| : 0 : Operation clock MCK specified by CKS04 bit
;||||| : 1 : Valid edge of input signal input from TI04 pin, fSUB/2, fSUB/4, or
INTRTC1
;|||||          (the timer input used with channel 4 is selected by using TIS0
register).
;|||||
;|||||+----- : Be sure to set 00
;|
;|+----- : Selection of operation clock (MCK) of channel 4
; : 0 : Operation clock CK00 set by TPS0 register

```

```

; : 1 : Operation clock CK01 set by TPS0 register
; CK01 = fCLK/2 = 20MHz -> 8kHz(0.125ms) = 0.05[us/cik] * 2500[count]
MOVW    TDR04,    #(2500 - 1)    ; set interval time to about 125us (= 8kHz)

SET1    TMMK04    ; disable interrupt

RET

;-----;
; Module:      SINITVR          ;
; Description:  Setting of Voltage reference      ;
; parameter:  --                ;
; return      :  --                ;
;-----;
SINITVR:
MOV     ADVRC,    #00001000B    ; Analog reference voltage control register
;| | | |+|++--- : [1]Positive reference voltage supplies selection of A/D and D/A
converters
;| | | |         [2]Operationcontrol of voltage reference
;| | | |         [3]Output voltage selection of voltage reference
;| | | |         [4]Operation control of input gate voltage boost circuit for
A/D converter
;| | | |         [5]Relationship with the conversion mode used
;| | | |
;| | | | : 0 0 0 : [1]AVREFP (external voltage reference input)
;| | | |         [2]Stops operation (Hi-Z)
;| | | |         [3]2.5 V
;| | | |         [4]Stops operation
;| | | |         [5]Can be set in conversion mode 1
;| | | |
;| | | | : 0 1 0 : [1]AVREFP (external voltage reference input)
;| | | |         [2]Stops operation (Hi-Z)
;| | | |         [3]2.0 V
;| | | |         [4]Enables operation
;| | | |         [5]Can be set in conversion mode 2 or 3
;| | | |
;| | | | : 1 0 0 : [1]VREFOUT (voltage reference output)
;| | | |         [2]Stops operation (pull-down output)
;| | | |         [3]2.5 V
;| | | |         [4]Stops operation
;| | | |         [5] -
;| | | |
;| | | | : 1 0 1 : [1]VREFOUT (voltage reference output)

```

```

;| | | | | [2]Enables operation
;| | | | | [3]2.5 V
;| | | | | [4]Enables operation
;| | | | | [5]Can be set in conversion mode 2 or 3
;| | | | |
;| | | | | : 1 1 0 : [1]VREFOUT (voltage reference output)
;| | | | | [2]Stops operation (pull-down output)
;| | | | | [3]2.0 V
;| | | | | [4]Enables operation
;| | | | | [5] -
;| | | | |
;| | | | | : 1 1 1 : [1]VREFOUT (voltage reference output)
;| | | | | [2]Enables operation
;| | | | | [3]2.0 V
;| | | | | [4]Enables operation
;| | | | | [5]Can be set in conversion mode 2 or 3
;| | | | |
;| | | | | : Other than the above : Setting prohibited
;| | | | |
;|+++----- : Be sure to set 0000
;|
;+----- : Reference voltage supply (negative side) of A/D converter
selection

; : 0 : AVSS
; : 1 : AVREFM (external voltage reference input)

SET1    ADVRC.0 ; Enables operation
SET1    ADVRC.1 ; Output 2.0 V

; wait for settling time to 20ms (over 17msec)
MOV     B,      #2 ; set counter
SET1    TSOL.0 ; start TAU0 CHO
JINIVR100:
CLR1    TMIF00 ; clear interrupt request flag
JINIVR200:
NOP
BF      TMIF00,$JINIVR200 ; wait 10msec
DEC     B ; 20msec elapsed ?
BNZ     $JINIVR100 ; No

CLR1    TMIF00 ; clear interrupt request flag
SET1    TTOL.0 ; stop TAU0 CHO

```

RET

```

;-----;
; Module:      SINITAMP      ;
; Description:  Setting of Operational amplifier      ;
;   parameter: --           ;
;   return   : --           ;
;-----;

```

SINITAMP:

```

MOV    ADPC,    #0000000B      ; A/D Port Configuration Register
;|||+---- : Analog input (A)/digital I/O (D) switching
;||| :          +----- ANI15/AVREFM/P157
;||| :          | +-+----- ANI10/P152 - ANI8/AMP2+/P150
;||| :          | | | +-+----- ANI7/AMP20/P27 - ANI0/AMP0-/P20
;||| : 0 0 0 0 0 : A A A A A A A A A A
;||| : 0 0 0 0 1 : A A A A A A A A A D
;||| : 0 0 0 1 0 : A A A A A A A A D D
;||| : 0 0 0 1 1 : A A A A A A A A D D D
;||| : 0 0 1 0 0 : A A A A A A A D D D D
;||| : 0 0 1 0 1 : A A A A A A A D D D D D
;||| : 0 0 1 1 0 : A A A A A A D D D D D D
;||| : 0 0 1 1 1 : A A A A A D D D D D D D
;||| : 0 1 0 0 0 : A A A A D D D D D D D D
;||| : 0 1 0 0 1 : A A A D D D D D D D D D
;||| : 0 1 0 1 0 : A A D D D D D D D D D D
;||| : 0 1 1 1 1 : A D D D D D D D D D D D
;||| : 1 0 0 0 0 : D D D D D D D D D D D D
;|||
;+++----- : Be sure to set 000

CLR1   OAC.1      ; operational amplifier (AMP1) disable
CLR1   OAC.2      ; operational amplifier (AMP2) disable

```

RET

```

;-----;
; Module:      SINITDAC      ;
; Description:  Setting of D/A Converter      ;
;   parameter: --           ;
;   return   : --           ;
;-----;

```

SINITDAC:

```

MOV    DAM,    #0100010B      ; D/A Converter Mode Register

```

```

;|||||+--- : DAMD0 D/A converter operation mode selection
;|||||+---- : DAMD1 D/A converter operation mode selection
;||||| : 0 : Normal mode
;||||| : 1 : Real-time output mode
;|||||
;|||||+----- : DARESO D/A converter resolution selection
;||||+----- : DARES1 D/A converter resolution selection
;|||| : 0 : 8-bit
;|||| : 1 : 12-bit
;||||
;|||+----- : D/A conversion operation Control (channel 0)
;||+----- : D/A conversion operation Control (channel 1)
;|| : 0 : Stops conversion operation
;|| : 1 : Enables conversion operation
;||
;|+----- : Positive reference voltage supply selection of D/A converter
;| : 0 : AVDD1 (power supply for D/A converter analog circuit)
;| : 1 : VREFOUT (voltage reference output) / AVREFP (external voltage reference
input)
;| (Reference voltage supply negative side is AVSS, positive side is
AVREFP -> AVREFP)
;| (Reference voltage supply negative side is AVREFM, positive side is
VREFOUT -> VREFOUT)
;|
;+----- : Be sure to set 0
MOVW DACSWO, #0800H ; set initial data

RET

;-----;
; Module: SPLAYDAC ;
; Description: Play PCM data by D/A ;
; parameter: -- ;
; return : -- ;
;-----;
SPLAYDAC:
;-----;
; prepare for playing ;
;-----;
MOVW HL, #LOWW TPLAYDATA ; set start playing addr (low 16bit)
MOV ES, #HIGHW TPLAYDATA; set start playing addr (high 4bit)

```

JPDAC100:

```

MOVW    AX,      #LOWW RADPCMWORK
CALL    !!_adpcm_init          ; ADPCM process Initialization

; operational amplifier setting
SET1    OAC.1          ; operational amplifier (AMP1) enable
SET1    OAC.2          ; operational amplifier (AMP2) enable
; Use software to wait until the operational amplifier stabilizes(20us(max.))
MOV     B,      #80
JPDAC150:
DEC     B
BNZ     $JPDAC150          ; 20us elapsed ? , No

; D/A converter setting
SET1    DACE0          ; D/A converter CH0 enable

; TAU0 CH4 setting for output timing
CLR1    TMIF04          ; clear interrupt request flag
SET1    TSOL.4          ; start TAU0 CH4

;*****;
;*-----*
;*  decode and play PCM data  *
;*-----*
;*****;
MOVW    RPLAYCOUNT,#0          ; clear output data counter

JPDAC200:
MOVW    AX,      RPLAYCOUNT          ; get number of output times
CMPW    AX,      !TPLAYSIZE          ; finished all data output ?
BNC     $JPDAC300          ; Yes

;*****;
;*  play low 4bit  *
;*****;
; decompression of ADPCM data (low 4bit -> 16bit)
MOVW    AX,      #LOWW RADPCMWORK          ; set work area for _adpcm_l32_dec
PUSH    AX
MOV     A,      ES:[HL]          ; get compressed data
AND     A,      #00FH          ; clear high 4bit
MOV     X,      A
CLRB    A
CLR1    DIVMODE          ; set multiplication mode (for _adpcm_l32_dec)
CALL    !!_adpcm_l32_dec          ; decompression of PCM data

```



```

POP      AX
MOVW     AX,      BC          ; get decompression data

; adjust play data
ADDW     AX,      #8000H      ; adjust sign
SHRW     AX,      (16-12)     ; right-align data

JPDAC220:
MOVW     DACSW0,  AX          ; set play data

; waiting for the output to be completed
JPDAC230:
NOP
BF       TMIF04,  $JPDAC230 ; the output to be completed ? , No
CLR1     TMIF04          ; clear interrupt request flag

;*****;
;*  play high 4bit  *;
;*****;
; decompression of ADPCM data (high 4bit -> 16bit)
MOVW     AX,      #LOWW RADPCMWORK ; set work area for _adpcm_l32_dec
PUSH     AX
MOV      A,      ES:[HL]      ; get compressed data
SHR      A,      4
MOV      X,      A
CLRB     A
CLR1     DIVMODE          ; set multiplication mode (for _adpcm_l32_dec)
CALL     !!_adpcm_l32_dec ; decompression of PCM data
POP      AX              ; pop argument
MOVW     AX,      BC          ; get decompression data

; adjust play data
ADDW     AX,      #8000H      ; adjust sign
SHRW     AX,      (16-12)     ; right-align data

JPDAC270:
MOVW     DACSW0,  AX          ; set play data

; waiting for the output to be completed
JPDAC280:
NOP
BF       TMIF04,  $JPDAC280 ; the output to be completed ? , No
CLR1     TMIF04          ; clear interrupt request flag

```

```

INCW    RPLAYCOUNT          ; update play counter
INCW    HL                    ; next play data
BR      JPDAC200

```

JPDAC300:

```

;-----;
;      finish playing      ;
;-----;
SET1    TTOL.4                ; stop TAU0 CH4
CLR1    DACE0                 ; D/A converter CH0 disable
CLR1    OAC.1                 ; operational amplifier (AMP1) disable
CLR1    OAC.2                 ; operational amplifier (AMP2) disable

```

JPDAC900:

```

RET

```

end

```
main.c (C言語版)

/*
 * Copyright (C) NEC Electronics Corporation 2006
 * NEC ELECTRONICS CONFIDENTIAL AND PROPRIETARY
 * All rights reserved by NEC Electronics Corporation.
 * This program must be used solely for the purpose for which
 * it was furnished by NEC Electronics Corporation. No part of this
 * program may be reproduced or disclosed to others, in any
 * form, without the prior written permission of NEC Electronics
 * Corporation. Use of copyright notice dose not evidence
 * publication of the program.
 */

/*-----*/
/* #pragma directive for CC78K0
                                     */
/*-----*/
#pragma      SFR
#pragma      DI
#pragma      EI
#pragma      HALT
#pragma      NOP

/*-----*/
/* Include files
                                     */
/*-----*/
#include"adpcmsp.h"

/*-----*/
/* Function prototyps
                                     */
/*-----*/
static void  fn_InitPort(void); /* Setting of I/O ports */
static void  fn_InitLvi(void); /* Low-voltage detection */
static void  fn_InitKr(void); /* Setting of Key Interrupt Function */
static void  fn_InitTau0(void); /* Setting of Timer array unit 0 */
static void  fn_InitVr(void); /* Setting of Voltage reference */
static void  fn_InitAmp(void); /* Setting of Operational amplifier */
static void  fn_InitDac(void); /* Setting of D/A Converter */

/*-----*/
/* Extern variables/constants
```

```

*/

/*-----*/
extern const unsigned char aPlayData[]; /* sound data */
extern unsigned short ushDataSize; /* size of sound data */

/*-----*/
/* Local constants
*/
/*-----*/
/*-----*/
/* Global variables
*/
/*-----*/
/*-----*/
/* Local variables
*/
/*-----*/

/* for play */
unsigned short ushAdpcmWork[16]; /* Work area for ADPCM process */

/*-----*/
/* Code
*/

/*-----*/
/*-----*/
/* Hardware initialization
*/
/*-----*/

void hdwinit(void)
{
    DI(); /* disable all interrupts */

    /*-----*/
    /* Initialization of port */
    /*-----*/
    fn_InitPort();

    /*-----*/
    /* Low-voltage detection */
    /*-----*/
    fn_InitLvi(); /* ensures 2.7V to VDD */

    /*-----*/

```

```

/*      Initialization of clock      */
/*-----*/
CMC =      0b01000011;      /* Clock Operation Mode Control Register */
/*|||||+--- : Control of high-speed system clock oscillation frequency */
/*||||| : 0 : 2 MHz <= fMX <= 10 MHz */
/*||||| : 1 : 10 MHz < fMX <= 20 MHz */
/*||||| */
/*|||||+---- : XT1 oscillator oscillation mode selection */
/*|||| : 0 0 : Low-consumption oscillation */
/*|||| : 0 1 : Normal oscillation */
/*|||| : 1 x : Super-low-consumption oscillation */
/*|||| x = don't care */
/*|||| */
/*||||+----- : Be sure to set 0 */
/*|||| */
/*|||+----- : [1] Subsystem clock pin operation mode */
/*|||      [2] XT1/P123 pin and XT2/P124 pin */
/*||| : 0 : [1]Input port mode */
/*|||      [2]Input port */
/*||| */
/*||| : 1 : [1]XT1 oscillation mode */
/*|||      [2]Crystal resonator connection */
/*||| */
/*||+----- : Be sure to set 0 */
/*|| */
/*++----- : [1]EXCLK OSCSEL High-speed system clock pin operation mode */
/*      [2]X1/P121 pin */
/*      [3]X2/EXCLK/P122 pin */
/* : 0 0 : [1]Input port mode */
/*      [2][3]Input port */
/* */
/* : 0 1 : [1]X1 oscillation mode */
/*      [2][3]Crystal/ceramic resonator connection */
/* */
/* : 1 0 : [1]Input port mode */
/*      [2][3]Input port */
/* */
/* : 1 1 : [1]External clock input mode */
/*      [2]Input port */
/*      [3]External clock input */

MSTOP = 0; /* X1 oscillator operating */

```

```
OSMC =      0b00000001;      /* Operation Speed Mode Control Register */
/*|||||++--- : fCLK frequency selection */
/*||||| : 0 0 : Operates at a frequency of 10 MHz or less. */
/*||||| : 0 1 : Operates at a frequency higher than 10 MHz. */
/*||||| : 1 0 : Operates at a frequency of 1 MHz. */
/*||||| */
/*|+++++----- : Be sure to set 00000 */
/*| */
/*+----- : Setting in subsystem clock HALT mode */
/* : 0 : Enables subsystem clock supply to peripheral functions. */
/*      (See Table 21-1 Operating Statuses in HALT Mode (2/3) */
/*      for the peripheral functions whose operations are enabled.) */
/* : 1 : Stops subsystem clock supply to peripheral functions */
/*      except real-time counter, clock output/buzzer output, */
/*      and LCD controller/driver. */
```

```
while(OSTC.0 != 1){          /* wait X1 oscillation stabilization */
    NOP();
}
}
```

```
/*-- Caution -----*/
/* To increase fCLK to 10 MHz or higher, set FSEL to '1', */
/* then change fCLK after two or more clocks have elapsed. */
/*-----*/
NOP();
NOP();
```

```
CKC =      0b00010000;      /* System Clock Control Register */
/*|+|+++++----- : Selection of CPU/peripheral hardware clock (fCLK) */
/*| | : 0 0 x 0 0 0 : fIH */
/*| | : 0 0 x 0 0 1 : fIH/2 (default) */
/*| | : 0 0 x 0 1 0 : fIH/2^2 */
/*| | : 0 0 x 0 1 1 : fIH/2^3 */
/*| | : 0 0 x 1 0 0 : fIH/2^4 */
/*| | : 0 0 x 1 0 1 : fIH/2^5 */
/*| | : 0 1 x 0 0 0 : fMX */
/*| | : 0 1 x 0 0 1 : fMX/2 */
/*| | : 0 1 x 0 1 0 : fMX/2^2 */
/*| | : 0 1 x 0 1 1 : fMX/2^3 */
/*| | : 0 1 x 1 0 0 : fMX/2^4 */
/*| | : 0 1 x 1 0 1 : fMX/2^5 */
/*| | : 1 x 0 x x x : fSUB */
/*| | : 1 x 1 x x x : fSUB/2 */
```

```

/*| | : Other than above : Setting prohibited */
/*| | x = don't care */
/*| | */
/*| +----- : Status of Main system clock (fMAIN) */
/*| : 0 : Internal high-speed oscillation clock (fIH) */
/*| : 1 : High-speed system clock (fMX) */
/*| */
/*+----- : Status of CPU/peripheral hardware clock (fCLK) */
/* : 0 : Main system clock (fMAIN) */
/* : 1 : Subsystem clock (fSUB) */

/* Confirming the CPU clock status */
while((CLS != 0)||MCS != 1){
    NOP();
}
/* CPU is operating on a High-speed system clock */
HIOSTOP = 1;          /* internal high-speed oscillation stopped */

OSTS =    0b00000111;    /* Oscillation Stabilization Time Select Register */
/*||||+++++--- : Oscillation stabilization time selection */
/*|||| : 0 0 0 : 2^8/fX */
/*|||| : 0 0 1 : 2^9/fX */
/*|||| : 0 1 0 : 2^10/fX */
/*|||| : 0 1 1 : 2^11/fX */
/*|||| : 1 0 0 : 2^13/fX */
/*|||| : 1 0 1 : 2^15/fX */
/*|||| : 1 1 0 : 2^17/fX */
/*|||| : 1 1 1 : 2^18/fX */
/*|||| */
/*+++++----- : Be sure to set 000000 */

PER0 =    0b01100001;    /* Peripheral Enable Register 0 */
/*|||||||+--- : Control of timer array unit 0 input clock */
/*||||||| : 0 : Stops input clock supply. */
/*|||||||      * SFR used by timer array unit 0 cannot be written. */
/*|||||||      * Timer array unit 0 is in the reset status. */
/*||||||| : 1 : Supplies input clock. */
/*|||||||      * SFR used by timer array unit 0 can be read and written. */
/*||||||| */
/*|||||||+----- : Control of timer array unit 1 input clock */
/*||||||| : 0 : Stops input clock supply. */
/*|||||||      * SFR used by timer array unit 1 cannot be written. */
/*|||||||      * Timer array unit 1 is in the reset status. */

```

```

/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by timer array unit 1 can be read and written. */
/*||||| */
/*|||||+----- : Control of serial array unit 0 input clock */
/*||||| : 0 : Stops input clock supply. */
/*|||||      * SFR used by the serial array unit 0 cannot be written. */
/*|||||      * The serial array unit 0 is in the reset status. */
/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by the serial array unit 0 can be read and written. */
/*||||| */
/*|||||+----- : Control of serial array unit 1 input clock */
/*||||| : 0 : Stops input clock supply. */
/*|||||      * SFR used by the serial array unit 1 cannot be written. */
/*|||||      * The serial array unit 1 is in the reset status. */
/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by the serial array unit 1 can be read and written. */
/*||||| */
/*|||||+----- : Control of serial interface IICA input clock */
/*||||| : 0 : Stops input clock supply. */
/*|||||      * SFR used by the serial interface IICA cannot be written. */
/*|||||      * The serial interface IICA is in the reset status. */
/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by the serial interface IICA can be read and written. */
/*||||| */
/*|||||+----- : Control of A/D converter, operational amplifier, and voltage reference input clock */
/*||||| : 0 : Stops input clock supply. */
/*|||||      * SFR used by the A/D converter, operational amplifier, and voltage reference cannot be
written. */
/*|||||      * The A/D converter, operational amplifier, and voltage reference is in the reset status. */
/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by the A/D converter, operational amplifier, and voltage reference can be
read and written. */
/*||||| */
/*|||||+----- : Control of D/A converter input clock */
/*||||| : 0 : Stops input clock supply. */
/*|||||      * SFR used by D/A converter cannot be written. */
/*|||||      * The D/A converter is in the reset status. */
/*||||| : 1 : Supplies input clock. */
/*|||||      * SFR used by the D/A converter can be read and written. */
/*||||| */
/*|+----- : Control of real-time counter (RTC) input clock */
/*| : 0 : Stops input clock supply. */
/*|      * SFR used by the real-time counter (RTC) cannot be written. */

```



```

/*          * The real-time counter (RTC) is in the reset status. */
/* : 1 : Supplies input clock. */
/*          * SFR used by the real-time counter (RTC) can be read and written. */

/*-----*/
/* Initialize of Key Interrupt Function */
/*-----*/
fn_InitKr();

/*-----*/
/*          Initialization of timer          */
/*-----*/
fn_InitTau0();

/*-----*/
/* Initialization of voltage reference */
/*-----*/
fn_InitVr();

/*-----*/
/* Initialization of Operational amplifier */
/*-----*/
fn_InitAmp();

/*-----*/
/*          Initialization of D/A Converter          */
/*-----*/
fn_InitDac();
}

/*-----*/
/* Module:          fn_InitPort
                   */
/* Description:     Setting of I/O ports
                   */
/* parameter: --
                   */
/* return : --
                   */
/*-----*/
static void fn_InitPort(void)
{
/*-----*/
/*          Setting of Port 0

```

```

        */
/*-----*/
    P0 =      0b00000000;    /* Set P00-P02 Output latch to Low */
    PM0 =     0b11111000;    /* Set P00-P02 to output port */
                          /* P00-P02:Unused */

/*-----*/
/*      Setting of Port 1
        */
/*-----*/
    P1 =      0b00000000;    /* Set P10-P17 Output latch to Low */
    PM1 =     0b00000000;    /* Set P10-P17 to output port */
                          /* P10-P15:Unused */

/*-----*/
/*      Setting of Port 2
        */
/*-----*/
    P2 =      0b00000000;    /* Set P20-P27 Output latch to Low */
    PM2 =     0b11111111;    /* Set P20-P27 to input port */
                          /* PM23:filter circuit(AMP1-) */
                          /* PM24:filter circuit(AMP1O) */
                          /* PM25:filter circuit(AMP1+) */
                          /* PM26:filter circuit(AMP2-) */
                          /* PM27:filter circuit(AMP2O) */
                          /* P20-P22:Unused */

/*-----*/
/*      Setting of Port 3
        */
/*-----*/
    P3 =      0b00001000;    /* Set P30-P32,P34 Output latch to Low */
                          /* Set P33 Output latch High */
    PM3 =     0b11100000;    /* Set P30-P34 to output port */
                          /* P30-P34:Unused */

/*-----*/
/*      Setting of Port 4
        */
/*-----*/
    P4 =      0b00000000;    /* Set P40-P41 Output latch to Low */
    PM4 =     0b11111100;    /* Set P40-P41 to output port */
                          /* P40-P41:Unused */

```

```

/*-----*/
/*      Setting of Port 5
          */
/*-----*/
P5 =      0b00000000;      /* Set P50-P57 Output latch to Low */
PM5 =     0b11110000;      /* Set P50-P57 to output port */
                          /* P50-P57:Unused */

/*-----*/
/*      Setting of Port 6
          */
/*-----*/
P6 =      0b00000000;      /* Set P60-P61 Output latch to Low */
PM6 =     0b11111100;      /* Set P60-P61 to output port */
                          /* P60-P61:Unused */

/*-----*/
/*      Setting of Port 7
          */
/*-----*/
P7 =      0b00000000;      /* Set P70-P77 Output latch to Low */
PM7 =     0b00000001;      /* Set P70 to input port, P71-P77 to output port */
PU7 =     0b00000001;      /* P70 on-chip pull-up resistor connected */
                          /* P70:key input port */
                          /* P71-P77:Unused */

/*-----*/
/*      Setting of Port 8
          */
/*-----*/
P8 =      0b00000000;      /* Set P80-P88 Output latch to Low */
PM8 =     0b00000000;      /* Set P80-P88 to output port */
                          /* P80-P88:Unused */

/*-----*/
/*      Setting of Port 9
          */
/*-----*/
P9 =      0b00000000;      /* Set P90-P97 Output latch to Low */
PM9 =     0b00000000;      /* Set P90-P97 to output port */
                          /* P90-P97:Unused */

```

```

/*-----*/
/*      Setting of Port 10
          */
/*-----*/
P10 =    0b00000000;    /* Set P100-P102 Output latch to Low */
PM10 =   0b11111000;    /* Set P100-P102 to output port */
          /* P100-P102:Unused */

/*-----*/
/*      Setting of Port 11
          */
/*-----*/
P11 =    0b00000000;    /* Set P110-P111 Output latch to Low */
PM11 =   0b11111101;    /* Set P110 to input port, P111 to output port */
          /* P110:play back output(ANO0) */
          /* P111:Unused */

/*-----*/
/*      Setting of Port 12
          */
/*-----*/
P12 =    0b00000000;    /* Set P120 Output latch to Low */
PM12 =   0b11111110;    /* Set P120 to output port */
          /* P120-P124:Unused */
          /* *P121-P124:Input port */

/*-----*/
/*      Setting of Port 13
          */
/*-----*/
P13 =    0b00000000;    /* Set P130 Output latch to Low */
          /* P130:Unused */

/*-----*/
/*      Setting of Port 14
          */
/*-----*/
P14 =    0b00000000;    /* Set P140-P147 Output latch to Low */
PM14 =   0b00000000;    /* Set P140-P147 to output port */
          /* P140-P147:Unused */

/*-----*/
/*      Setting of Port 15

```

```

        */
/*-----*/
    P15 =      0b00000000;    /* Set P150-P152,P157 Output latch to Low */
    PM15 =     0b11111111;    /* Set P150-P152,P157 to input port */
                                /* PM150:filter circuit(AMP2+) */
                                /* P151-P152,P157:Unused */
}

/*-----*/
/* Module:      fn_InitLvi
                */

/* Description:  Ensures 2.7V to the power supply voltage      */
/* parameter:  --
                */

/* return   :  --
                */

/*-----*/
static void    fn_InitLvi(void)
{
    unsigned char ucCounter;    /* counter */

    /* setting of Low-Voltage Detector */
    LVIMK      = 1;    /* disable LVI interrupt */
    LVISEL     = 0;    /* detects level of VDD */
    LVIS =     0b00001001;    /* Low-Voltage Detection Level Select Register */
    /*|||+++++--- : Detection level */
    /*||| : 0 0 0 0 : VLVI0 (4.22V) */
    /*||| : 0 0 0 1 : VLVI1 (4.07V) */
    /*||| : 0 0 1 0 : VLVI2 (3.92V) */
    /*||| : 0 0 1 1 : VLVI3 (3.76V) */
    /*||| : 0 1 0 0 : VLVI4 (3.61V) */
    /*||| : 0 1 0 1 : VLVI5 (3.45V) */
    /*||| : 0 1 1 0 : VLVI6 (3.30V) */
    /*||| : 0 1 1 1 : VLVI7 (3.15V) */
    /*||| : 1 0 0 0 : VLVI8 (2.99V) */
    /*||| : 1 0 0 1 : VLVI9 (2.84V) */
    /*||| : 1 0 1 0 : VLVI10 (2.68V) */
    /*||| : 1 0 1 1 : VLVI11 (2.53V) */
    /*||| : 1 1 0 0 : VLVI12 (2.38V) */
    /*||| : 1 1 0 1 : VLVI13 (2.22V) */
    /*||| : 1 1 1 0 : VLVI14 (2.07V) */
    /*||| : 1 1 1 1 : VLVI15 (1.91V) */
    /*||| */
}

```

```

    /*++++----- : Be sure to set 0000 */
LVIMD      = 0;      /* generates an internal interrupt signal when detect the low-voltage */
LVION      = 1;      /* enables low-voltage detection operation */

/* software to wait for the operation stabilization time (over 10us) */
for(ucCounter = 0; ucCounter < 4; ucCounter++){
    NOP();
}

/* wait for VDD to become VLVI or more */
while(LVIF){
    NOP();
}
LVION      = 0;      /* disables low-voltage detection operation */
}

/*-----*/
/* Module:      fn_InitKr
                */
/* Description:  Setting of Key Interrupt Function
                */
/* parameter: --
                */
/* return  : --
                */
/*-----*/
static void    fn_InitKr(void)
{
    KRMK = 1;      /* disable key interrupt */
    KRM = 0b0000001; /* Key Return Mode Register */
    /*|||||+--- : KR0 interrupt mode control */
    /*|||||+---- : KR1 interrupt mode control */
    /*|||||+----- : KR2 interrupt mode control */
    /*|||+----- : KR3 interrupt mode control */
    /*||+----- : KR4 interrupt mode control */
    /*|+----- : KR5 interrupt mode control */
    /*+----- : KR6 interrupt mode control */
    /*+----- : KR7 interrupt mode control */
    /* : 0 : Does not detect key interrupt signal */
    /* : 1 : Detects key interrupt signal */

    NOP();      /* 250ns interval from set KRM to clear KRIF */
    NOP();
    NOP();
    NOP();
}

```

```

NOP();
KRIF = 0; /* clear key interrupt request flag */
}

/*-----*/
/* Module:      fn_InitTau0
*/
/* Description:  Setting of Timer array unit 0
*/
/* parameter: --
*/
/* return  : --
*/
/*-----*/
static void fn_InitTau0(void)
{
    TPSOL = 0b00000010; /* Timer Clock Select Register 0 */
    /*|||++++--- : Selection of operation clock (CK0) */
    /*++++----- : Selection of operation clock (CK01) */
    /* : 0 0 0 0 : CK0m = fCLK */
    /* : 0 0 0 1 : CK0m = fCLK/2 */
    /* : 0 0 1 0 : CK0m = fCLK/2^2 */
    /* : 0 0 1 1 : CK0m = fCLK/2^3 */
    /* : 0 1 0 0 : CK0m = fCLK/2^4 */
    /* : 0 1 0 1 : CK0m = fCLK/2^5 */
    /* : 0 1 1 0 : CK0m = fCLK/2^6 */
    /* : 0 1 1 1 : CK0m = fCLK/2^7 */
    /* : 1 0 0 0 : CK0m = fCLK/2^8 */
    /* : 1 0 0 1 : CK0m = fCLK/2^9 */
    /* : 1 0 1 0 : CK0m = fCLK/2^10 */
    /* : 1 0 1 1 : CK0m = fCLK/2^11 */
    /* : 1 1 0 0 : CK0m = fCLK/2^12 */
    /* : 1 1 0 1 : CK0m = fCLK/2^13 */
    /* : 1 1 1 0 : CK0m = fCLK/2^14 */
    /* : 1 1 1 1 : CK0m = fCLK/2^15 */
    /* m = 0, 1 */

    /* CH0:for timing */
    TMR00 = 0b0000000000000000; /* Timer Mode Register 00 */
    /*|||||||||++++--- : [1]Operation mode of channel 0 */
    /*||||||||| [2]Count operation of TCR */
    /*||||||||| [3]Independent operation */
    /*||||||||| [4]Setting of starting counting and interrupt */
    /*||||||||| : 0 0 0 0 : [1]Interval timer mode */

```

```

/*|||||||      [2]Counting down */
/*|||||||      [3]Possible */
/*|||||||      [4]Timer interrupt is not generated */
/*|||||||      when counting is started */
/*|||||||      (timer output does not change, either). */
/*||||||| */
/*||||||| : 0 0 0 1 : [1]Interval timer mode */
/*|||||||      [2]Counting down */
/*|||||||      [3]Possible */
/*|||||||      [4]Timer interrupt is generated */
/*|||||||      when counting is started */
/*|||||||      (timer output also changes). */
/*||||||| */
/*||||||| : 0 1 0 0 : [1]Capture mode */
/*|||||||      [2]Counting up */
/*|||||||      [3]Possible */
/*|||||||      [4]Timer interrupt is not generated */
/*|||||||      when counting is started */
/*|||||||      (timer output does not change, either). */
/*||||||| */
/*||||||| : 0 1 0 1 : [1]Capture mode */
/*|||||||      [2]Counting up */
/*|||||||      [3]Possible */
/*|||||||      [4]Timer interrupt is generated */
/*|||||||      when counting is started */
/*|||||||      (timer output also changes). */
/*||||||| */
/*||||||| : 0 1 1 0 : [1]Event counter mode */
/*|||||||      [2]Counting down */
/*|||||||      [3]Possible */
/*|||||||      [4]Timer interrupt is not generated */
/*|||||||      when counting is started */
/*|||||||      (timer output does not change, either). */
/*||||||| */
/*||||||| : 1 0 0 0 : [1]One-count mode */
/*|||||||      [2]Counting down */
/*|||||||      [3]Impossible */
/*|||||||      [4]Start trigger is invalid */
/*|||||||      during counting operation. */
/*|||||||      At that time, interrupt */
/*|||||||      is not generated, either. */
/*||||||| */
/*||||||| : 1 0 0 1 : [1]One-count mode */

```



```

/*|||||||          [2]Counting down */
/*|||||||          [3]Impossible */
/*|||||||          [4]Start trigger is valid */
/*|||||||          during counting operation. */
/*|||||||          At that time, interrupt */
/*|||||||          is also generated. */
/*||||||| */
/*||||||| : 1 1 0 0 : [1]Capture & one-count mode */
/*|||||||          [2]Counting up */
/*|||||||          [3]Possible */
/*|||||||          [4]Timer interrupt is not generated */
/*|||||||          when counting is started */
/*|||||||          (timer output does not change, either). */
/*|||||||          Start trigger is invalid */
/*|||||||          during counting operation. */
/*|||||||          At that time interrupt */
/*|||||||          is not generated, either. */
/*||||||| */
/*||||||| : Other than above : Setting prohibited */
/*||||||| */
/*|||||||++----- : Be sure to set 00 */
/*||||||| */
/*|||||||++----- : Selection of T100 pin input signal, */
/*|||||||          fSUB/2, fSUB/4, or INTRTC1 valid edge */
/*|||||||          (the timer input used with channel 0 */
/*|||||||          is selected by using TIS0 register). */
/*||||||| : 0 0 : Falling edge */
/*||||||| : 0 1 : Rising edge */
/*||||||| : 1 0 : Both edges (when low-level width is measured) */
/*|||||||          Start trigger: Falling edge, Capture trigger: Rising edge */
/*||||||| : 1 1 : Both edges (when high-level width is measured) */
/*|||||||          Start trigger: Rising edge, Capture trigger: Falling edge */
/*||||||| */
/*|||||+++----- : Setting of start trigger or capture trigger of channel 0 */
/*||||| : 0 0 0 : Only software trigger start is valid */
/*|||||          (other trigger sources are unselected). */
/*||||| : 0 0 1 : Valid edge of T100 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          is used as both the start trigger and capture trigger. */
/*||||| : 0 1 0 : Both the edges of T100 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*|||||          are used as a start trigger and a capture trigger. */
/*||||| : 1 0 0 : Interrupt signal of the master channel is used */
/*|||||          (when the channel is used as a slave channel */
/*|||||          with the combination operation function). */

```

```

/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||+----- : Selection of slave/master of channel 0 */
/*||||| : 0 : Operates as slave channel with combination operation function. */
/*||||| : 1 : Operates as master channel with combination operation function. */
/*||||| */
/*|||+----- : Selection of count clock (TCLK) of channel 0 */
/*||| : 0 : Operation clock MCK specified by CKS00 bit */
/*||| : 1 : Valid edge of input signal input from T100 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*||| (the timer input used with channel 0 is selected by using TISO register). */
/*||| */
/*|++----- : Be sure to set 00 */
/*| */
/*+----- : Selection of operation clock (MCK) of channel 0 */
/* : 0 : Operation clock CK00 set by TPS0 register */
/* : 1 : Operation clock CK01 set by TPS0 register */
/* CK00 = fCLK/2^3 = 5MHz -> 10ms = 0.2[us/clock] * 50000[count] */
TDR00 = (50000 - 1); /* set interval time to 10 ms */

TMMK00 = 1; /* disable interrupt */

/* CH4:for play back sampling timing */
TMR04 = 0b1000000000000000; /* Timer Mode Register 04 */
/*|||||||+----- : [1]Operation mode of channel 4 */
/*||||||| [2]Count operation of TCR */
/*||||||| [3]Independent operation */
/*||||||| [4]Setting of starting counting and interrupt */
/*||||||| : 0 0 0 0 : [1]Interval timer mode */
/*||||||| [2]Counting down */
/*||||||| [3]Possible */
/*||||||| [4]Timer interrupt is not generated */
/*||||||| when counting is started */
/*||||||| (timer output does not change, either). */
/*||||||| */
/*||||||| : 0 0 0 1 : [1]Interval timer mode */
/*||||||| [2]Counting down */
/*||||||| [3]Possible */
/*||||||| [4]Timer interrupt is generated */
/*||||||| when counting is started */
/*||||||| (timer output also changes). */
/*||||||| */
/*||||||| : 0 1 0 0 : [1]Capture mode */
/*||||||| [2]Counting up */

```

```

/*||||||||| [3]Possible */
/*||||||||| [4]Timer interrupt is not generated */
/*||||||||| when counting is started */
/*||||||||| (timer output does not change, either). */
/*||||||||| */
/*||||||||| : 0 1 0 1 : [1]Capture mode */
/*||||||||| [2]Counting up */
/*||||||||| [3]Possible */
/*||||||||| [4]Timer interrupt is generated */
/*||||||||| when counting is started */
/*||||||||| (timer output also changes). */
/*||||||||| */
/*||||||||| : 0 1 1 0 : [1]Event counter mode */
/*||||||||| [2]Counting down */
/*||||||||| [3]Possible */
/*||||||||| [4]Timer interrupt is not generated */
/*||||||||| when counting is started */
/*||||||||| (timer output does not change, either). */
/*||||||||| */
/*||||||||| : 1 0 0 0 : [1]One-count mode */
/*||||||||| [2]Counting down */
/*||||||||| [3]Impossible */
/*||||||||| [4]Start trigger is invalid */
/*||||||||| during counting operation. */
/*||||||||| At that time, interrupt */
/*||||||||| is not generated, either. */
/*||||||||| */
/*||||||||| : 1 0 0 1 : [1]One-count mode */
/*||||||||| [2]Counting down */
/*||||||||| [3]Impossible */
/*||||||||| [4]Start trigger is valid */
/*||||||||| during counting operation. */
/*||||||||| At that time, interrupt */
/*||||||||| is also generated. */
/*||||||||| */
/*||||||||| : 1 1 0 0 : [1]Capture & one-count mode */
/*||||||||| [2]Counting up */
/*||||||||| [3]Possible */
/*||||||||| [4]Timer interrupt is not generated */
/*||||||||| when counting is started */
/*||||||||| (timer output does not change, either). */
/*||||||||| Start trigger is invalid */
/*||||||||| during counting operation. */

```

```

/*||||| At that time interrupt */
/*||||| is not generated, either. */
/*||||| */
/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||++----- : Be sure to set 00 */
/*||||| */
/*|||||++----- : Selection of TI04 pin input signal, */
/*||||| fSUB/2, fSUB/4, or INTRTC1 valid edge */
/*||||| (the timer input used with channel 4 */
/*||||| is selected by using TIS0 register). */
/*||||| : 0 0 : Falling edge */
/*||||| : 0 1 : Rising edge */
/*||||| : 1 0 : Both edges (when low-level width is measured) */
/*||||| Start trigger: Falling edge, Capture trigger: Rising edge */
/*||||| : 1 1 : Both edges (when high-level width is measured) */
/*||||| Start trigger: Rising edge, Capture trigger: Falling edge */
/*||||| */
/*|||||+++----- : Setting of start trigger or capture trigger of channel 4 */
/*||||| : 0 0 0 : Only software trigger start is valid */
/*||||| (other trigger sources are unselected). */
/*||||| : 0 0 1 : Valid edge of TI04 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*||||| is used as both the start trigger and capture trigger. */
/*||||| : 0 1 0 : Both the edges of TI04 pin input signal, fSUB/2, fSUB/4, or INTRTC1 */
/*||||| are used as a start trigger and a capture trigger. */
/*||||| : 1 0 0 : Interrupt signal of the master channel is used */
/*||||| (when the channel is used as a slave channel */
/*||||| with the combination operation function). */
/*||||| : Other than above : Setting prohibited */
/*||||| */
/*|||||+----- : Selection of slave/master of channel 4 */
/*||||| : 0 : Operates as slave channel with combination operation function. */
/*||||| : 1 : Operates as master channel with combination operation function. */
/*||||| */
/*|||||+----- : Selection of count clock (TCLK) of channel 0 */
/*||||| : 0 : Operation clock MCK specified by CKS04 bit */
/*||||| : 1 : Valid edge of input signal input from TI04 pin, fSUB/2, fSUB/4, or INTRTC1 */
/*||||| (the timer input used with channel 4 is selected by using TIS0 register). */
/*||||| */
/*|||||+----- : Be sure to set 00 */
/*||||| */
/*|+----- : Selection of operation clock (MCK) of channel 4 */
/* : 0 : Operation clock CK00 set by TPS0 register */

```

```

/* : 1 : Operation clock CK01 set by TPS0 register */
/* CK01 = fCLK/2 = 20MHz -> 8kHz(0.125ms) = 0.05[us/clock] * 2500[count] */
TDR04 = (2500 - 1); /* set interval time to about 125us(= 8kHz) */

TMMK04 = 1; /* disable interrupt */
}

/*-----*/
/* Module:      fn_InitVr
                */
/* Description:  Setting of Voltage reference                               */
/* parameter: --
                */
/* return  : --
                */
/*-----*/
static void fn_InitVr(void)
{
    unsigned char work;

    ADVRC = 0b00001000; /* Analog reference voltage control register */
    /*|||+|+--- : [1]Positive reference voltage supplies selection of A/D and D/A converters */
    /*||| |      [2]Operationcontrol of voltage reference */
    /*||| |      [3]Output voltage selection of voltage reference */
    /*||| |      [4]Operation control of input gate voltage boost circuit for A/D converter */
    /*||| |      [5]Relationship with the conversion mode used */
    /*||| | */
    /*||| | : 0 0 0 : [1]AVREFP (external voltage reference input) */
    /*||| |      [2]Stops operation (Hi-Z) */
    /*||| |      [3]2.5 V */
    /*||| |      [4]Stops operation */
    /*||| |      [5]Can be set in conversion mode 1 */
    /*||| | */
    /*||| | : 0 1 0 : [1]AVREFP (external voltage reference input) */
    /*||| |      [2]Stops operation (Hi-Z) */
    /*||| |      [3]2.0 V */
    /*||| |      [4]Enables operation */
    /*||| |      [5]Can be set in conversion mode 2 or 3 */
    /*||| | */
    /*||| | : 1 0 0 : [1]VREFOUT (voltage reference output) */
    /*||| |      [2]Stops operation (pull-down output) */
    /*||| |      [3]2.5 V */
    /*||| |      [4]Stops operation */

```

```

/*||| |          [5] - */
/*||| | */
/*||| | : 1 0 1 : [1]VREFOUT (voltage reference output) */
/*||| |          [2]Enables operation */
/*||| |          [3]2.5 V */
/*||| |          [4]Enables operation */
/*||| |          [5]Can be set in conversion mode 2 or 3 */
/*||| | */
/*||| | : 1 1 0 : [1]VREFOUT (voltage reference output) */
/*||| |          [2]Stops operation (pull-down output) */
/*||| |          [3]2.0 V */
/*||| |          [4]Enables operation */
/*||| |          [5] - */
/*||| | */
/*||| | : 1 1 1 : [1]VREFOUT (voltage reference output) */
/*||| |          [2]Enables operation */
/*||| |          [3]2.0 V */
/*||| |          [4]Enables operation */
/*||| |          [5]Can be set in conversion mode 2 or 3 */
/*||| | */
/*||| | : Other than the above : Setting prohibited */
/*||| | */
/*|+++----- : Be sure to set 0000 */
/*| */
/*+----- : Reference voltage supply (negative side) of A/D converter selection */
/* : 0 : AVSS */
/* : 1 : AVREFM (external voltage reference input) */

```

ADVRC.0 = 1; /\* Enables operation \*/

ADVRC.1 = 1; /\* Output 2.0 V \*/

/\* wait for settling time to 20ms (over 17msec) \*/

TSOL.0 = 1; /\* start TAU0 CH0 \*/

for(work = 2; work > 0; work--){ /\* wait 10msec\*2 \*/

    TMIF00 = 0; /\* clear interrupt request flag \*/

    while(!TMIF00){

        NOP(); /\* wait 10msec \*/

    }

}

TMIF00 = 0; /\* clear interrupt request flag \*/

TTOL.0 = 1; /\* stop TAU0 CH0 \*/

}

```

/*-----*/
/* Module:      fn_InitAmp
                */

/* Description:  Setting of Operational amplifier          */
/* parameter: --
                */

/* return  : --
                */

/*-----*/
static void    fn_InitAmp(void)
{
    ADPC =    0b00000000;    /* A/D Port Configuration Register */
    /*|||+++++--- : Analog input (A)/digital I/O (D) switching */
    /*||| :      +----- ANI15/AVREFM/P157 */
    /*||| :      | ++++----- ANI10/P152 - ANI8/AMP2+/P150 */
    /*||| :      ||| |+++++----- ANI7/AMP2O/P27 - ANI0/AMP0-/P20 */
    /*||| : 0 0 0 0 0 : A A A A A A A A A A A A */
    /*||| : 0 0 0 0 1 : A A A A A A A A A A A D */
    /*||| : 0 0 0 1 0 : A A A A A A A A A A D D */
    /*||| : 0 0 0 1 1 : A A A A A A A A A D D D */
    /*||| : 0 0 1 0 0 : A A A A A A A A D D D D */
    /*||| : 0 0 1 0 1 : A A A A A A A D D D D D */
    /*||| : 0 0 1 1 0 : A A A A A A D D D D D D */
    /*||| : 0 0 1 1 1 : A A A A A D D D D D D D */
    /*||| : 0 1 0 0 0 : A A A A D D D D D D D D */
    /*||| : 0 1 0 0 1 : A A A D D D D D D D D D */
    /*||| : 0 1 0 1 0 : A A D D D D D D D D D D */
    /*||| : 0 1 1 1 1 : A D D D D D D D D D D D */
    /*||| : 1 0 0 0 0 : D D D D D D D D D D D D */
    /*||| */
    /*+++----- : Be sure to set 000 */

    OAC.1 = 0;    /* operational amplifier (AMP1) disable */
    OAC.2 = 0;    /* operational amplifier (AMP2) disable */
}

/*-----*/
/* Module:      fn_InitDac
                */

/* Description:  Setting of D/A Converter          */
/* parameter: --
                */

/* return  : --
                */

```

```

                                                    */
/*-----*/
static void    fn_InitDac(void)
{
    DAM =      0b01000101;    /* D/A Converter Mode Register */
    /*|||||+--- : DAMD0 D/A converter operation mode selection */
    /*|||||+---- : DAMD1 D/A converter operation mode selection */
    /*||||| : 0 : Normal mode */
    /*||||| : 1 : Real-time output mode */
    /*||||| */
    /*||||+----- : DARES0 D/A converter resolution selection */
    /*|||+----- : DARES1 D/A converter resolution selection */
    /*||| : 0 : 8-bit */
    /*||| : 1 : 12-bit */
    /*||| */
    /*||+----- : D/A conversion operation Control (channel 0) */
    /*|+----- : D/A conversion operation Control (channel 1) */
    /*| : 0 : Stops conversion operation */
    /*| : 1 : Enables conversion operation */
    /*| */
    /*|+----- : Positive reference voltage supply selection of D/A converter */
    /*| : 0 : AVDD1 (power supply for D/A converter analog circuit) */
    /*| : 1 : VREFOUT (voltage reference output) / AVREFP (external voltage reference input) */
    /*|      (Reference voltage supply negative side is AVSS, positive side is AVREFP -> AVREFP) */
    /*|      (Reference voltage supply negative side is AVREFM, positive side is VREFOUT ->
VREFOUT) */
    /*| */
    /*+----- : Be sure to set 0 */
    DACSW0 = 0x0800;    /* set initial data */
}

/*-----*/
/* Module:      fn_PlayDac
                                                    */
/* Description:  Play PCM data by D/A
    */
/*   parameter: --
                                                    */
/*   return   : --
                                                    */
/*-----*/
static void    fn_PlayDac(void)
{

```



```

unsigned char *      pucPlayData;      /* start playing addr */
unsigned short      ushPlayCount;      /* output data counter */
unsigned short      ushData; /* decompression data */
unsigned short      loop; /* waiting counter */

/*-----*/
/*      prepare for playing      */
/*-----*/
/* set play data addr and size */
pucPlayData = aPlayData;      /* set start playing address */

adpcm_init(ushAdpcmWork);      /* ADPCM process Initialization */

/* operational amplifier setting */
OAC.1 = 1; /* operational amplifier (AMP1) enable */
OAC.2 = 1; /* operational amplifier (AMP2) enable */
/* Use software to wait until the operational amplifier stabilizes(20us(max.)) */
for(loop = 40; loop > 0; loop--){
    NOP();
}

/* D/A converter setting */
DACE0 = 1; /* D/A converter CH0 enable */

/* TAU0 CH4 setting for output timing */
TMIF04 = 0; /* clear interrupt request flag */
TSOL.4 = 1; /* start TAU0 CH4 */

/*****/
/*-----*/
/*      decode and play PCM data      */
/*-----*/
/*****/
for(ushPlayCount = 0; ushPlayCount < ushDataSize; ushPlayCount++){
    /*****/
    /*      play low 4bit      */
    /*****/
    /* decompression of ADPCM data (low 4bit -> 16bit)*/
    DIVMODE = 0; /* set multiplication mode (for _adpcm_l32_dec) */
    ushData = (unsigned short)adpcm_l32_dec(pucPlayData[ushPlayCount] & 0x0f, ushAdpcmWork);

    /* adjust sign & right-align data */
    ushData = (unsigned short)((ushData + 0x8000) >> (16-12));
}

```

```
/* set play data */
DACSW0 = ushData;

/* waiting for the output to be completed */
while(!TMIF04){
    NOP();
}
TMIF04 = 0;      /* clear interrupt request flag */

/*****
/*   play high 4bit   */
*****/

/* decompression of ADPCM data (high 4bit -> 16bit) */
DIVMODE = 0;    /* set multiplication mode (for _adpcm_l32_dec) */
ushData = (unsigned short)adpcm_l32_dec((pucPlayData[ushPlayCount] >> 4) & 0x0f, ushAdpcmWork);

/* adjust sign & right-align data */
ushData = (unsigned short)((ushData + 0x8000) >> (16-12));

/* set play data */
DACSW0 = ushData;

/* waiting for the output to be completed */
while(!TMIF04){
    NOP();
}
TMIF04 = 0;      /* clear interrupt request flag */
}

/*-----*/
/*      finish playing      */
/*-----*/
TT0L.4 = 1;  /* stop TAU0 CH4 */
DACE0 = 0;  /* D/A converter CH0 disable */
OAC.1 = 0;  /* operational amplifier (AMP1) disable */
OAC.2 = 0;  /* operational amplifier (AMP2) disable */

}

/*-----*/
/* Module:      main
*/
```

```

/* Description:    Main process
                */
/*    parameter: --
                */
/*    return    : --
                */
/*-----*/
void main(void)
{
    /*=====*/
    /*-----*/
    /*          Main Loop          */
    /*-----*/
    /*=====*/
    while(1){
        /*-----*/
        /*          */
        /*    play melody when the key is input    */
        /*          */
        /*-----*/
        /*-----*/
        /*    wait key input    */
        /*-----*/
        KRMK = 0;          /* clear key interrupt */
        KRIF = 0; /* clear key interrupt request flag */
        HALT(); /* sets the HALT mode */
        KRMK = 1;          /* set key interrupt */
        KRIF = 0; /* clear key interrupt request flag */

        /*-----*/
        /* remove key input noise */
        /*-----*/
        if(P7.0 == 0){          /* key input ? */
            TS0L.0 = 1; /* start TAU0 CH0 */
            TMIF00 = 0;
            while(!TMIF00){
                NOP(); /* wait 10msec */
            }
            TMIF00 = 0;
            TT0L.0 = 1; /* stop TAU0 CH0 */

            if(P7.0 == 0){          /* key input ? */
                /*-----*/

```

```
        /*      play melody      */
        /*-----*/
        fn_PlayDac();
    }
}

/*****/
/*                                  */
/*  The main processing writes here  */
/*      if there is something      */
/*                                  */
/*****/

}
}
```

## 付録B 改版履歴

版 数	発行年月	改版箇所	改版内容
第1版	September 2009	-	-

【発行】 NECエレクトロニクス株式会社 (<http://www.necel.co.jp/>)

【問い合わせ先】 <http://www.necel.com/contact/ja/>