

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い申し上げます。

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。

ユーザーズ・マニュアル

μ SAP77016-B01

音響エコー・キャンセラ・ミドルウェア

対象デバイス

μ PD77015
 μ PD77016
 μ PD77017
 μ PD77018A
 μ PD77019
 μ PD77110
 μ PD77111
 μ PD77112
 μ PD77113
 μ PD77113A
 μ PD77114
 μ PD77115

資料番号 U13130JJ4V0UM00 (第4版)

発行年月 April 2001 NS CP (K)

[× モ]

目 次 要 約

第1章 概 説 … 10

第2章 ライブラリ仕様 … 13

第3章 インストレーション … 21

付録 サンプル・ソース (sample.asm) … 24

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

- **本資料の内容は予告なく変更することがありますので、最新のものであることをご確認の上ご使用ください。**
- 文書による当社の承諾なしに本資料の転載複製を禁じます。
- 本資料に記載された製品の使用もしくは本資料に記載の情報の使用に際して、当社は当社もしくは第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。上記使用に起因する第三者所有の権利にかかわる問題が発生した場合、当社はその責を負うものではありませんのでご了承ください。
- 本資料に記載された回路、ソフトウェア、及びこれらに付随する情報は、半導体製品の動作例、応用例を説明するためのものです。従って、これら回路・ソフトウェア・情報をお客様の機器に使用される場合には、お客様の責任において機器設計をしてください。これらの使用に起因するお客様もしくは第三者の損害に対して、当社は一切その責を負いません。

M7A 98.8

本版で改訂された主な箇所

箇 所	内 容
p.11	1.3.2 (1) 動作対象 DSP に μ PD77113A を追加, μ PD77018, 77116 を削除
p.12	1.3.4 ディレクトリ構成の説明を変更

本文欄外の★印は、本版で改訂された主な箇所を示しています。

巻末にアンケート・コーナーを設けております。このドキュメントに対するご意見をお気軽にお寄せください。

はじめに

対象者 このユーザーズ・マニュアルは、 μ PD77016 ファミリの機能を理解し、それを用いたアプリケーション・プログラムを設計するユーザを対象とします。

μ PD77016 ファミリは、 μ PD7701 $\times$ ファミリ (μ PD77015, 77016, 77017, 77018A, 77019) と、 μ PD77111 ファミリ (μ PD77110, 77111, 77112, 77113, 77113A, 77114, 77115) の総称です。

目的 このユーザーズ・マニュアルは、 μ PD77016 ファミリの基礎的な機能について、応用プログラムを用いてユーザに理解していただくことを目的とします。

構成 このユーザーズ・マニュアルでは、基本的な数値演算プログラムなどについて説明しています。

第1章 概 説

第2章 ライブラリ仕様

第3章 インストレーション

付 録 サンプル・ソース (sample.asm)

読み方 このマニュアルの読者は、論理回路やマイクロコンピュータに関する一般的知識が必要となります。

μ PD7701 $\times$ ファミリのハードウェア機能を知りたいとき

→ μ PD7701 $\times$ ファミリ ユーザーズ・マニュアル アーキテクチャ編を参照してください。

μ PD77111 ファミリのハードウェア機能を知りたいとき

→ μ PD77111 ファミリ ユーザーズ・マニュアル アーキテクチャ編を参照してください。

μ PD77016 ファミリの命令機能を知りたいとき

→ μ PD77016 ファミリ ユーザーズ・マニュアル 命令編を参照してください。

凡 例 データ表記の重み：左が上位桁、右が下位桁

アクティブ・ロウの表記： $\overline{\times\times\times}$ (端子、信号の名称に上線)

注 : 本文中につけた注の説明

注意 : 気をつけて読んでいただきたい内容

備考 : 本文中の補足説明

数の表記 : 2 進数 $\cdots\times\times\times\times$ または 0b $\times\times\times\times$

10 進数 $\cdots\times\times\times\times$

16 進数 $\cdots0x\times\times\times\times$

関連資料 関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。あらかじめご了承ください。

デバイスに関する資料

資料名 品名	パンフレット	データ・シート	ユーザーズ・マニュアル		アプリケーション・ノート	
			アーキテクチャ編	命令編	基本ソフトウェア編	ライブラリ編
μ PD77016	U12395J	U10891J	U10503J	U13116J	U11958J	U12021J
μ PD77015		U10902J				
μ PD77017						
μ PD77018A		U11849J				
μ PD77019						
μ PD77019-013		U13053J				
μ PD77110		U12801J	U14623J			
μ PD77111						
μ PD77112						
μ PD77113A		U14373J				
μ PD77114						
μ PD77115		U14867J				

開発ツールに関する資料

資 料 名		資料番号
HSM77016 ユーザーズ・マニュアル		U11602J
WB77016 ユーザーズ・マニュアル	言語編	U10078J
	操作編	U11506J
ID77016 ユーザーズ・マニュアル		U10118J
μ PD77016 スタータ・キット ユーザーズ・マニュアル		U13032J
RX77016 ユーザーズ・マニュアル	機能編	U14397J
	コンフィギュレーション・ツール編	U14404J
RX77016 アプリケーション・ノート	HOST API 編	U14371J

注意 上記関連資料は、予告なしに内容を変更することがあります。設計などには、必ず最新の資料をご使用ください。

目 次

第1章 概 説 … 10

- 1.1 ミドルウェア … 10
- 1.2 エコー・キャンセラ … 10
- 1.3 製品概要 … 11
 - 1.3.1 特 徴 … 11
 - 1.3.2 動作環境 … 11
 - 1.3.3 性 能 … 12
 - 1.3.4 ディレクトリ構成 … 12

第2章 ライブラリ仕様 … 13

- 2.1 エコー・キャンセル処理フロー … 13
- 2.2 関数仕様 … 14
 - 2.2.1 メモリ構造 … 14
 - 2.2.2 マクロ … 15
 - 2.2.3 aec_Initialize 関数 … 16
 - 2.2.4 aec_EchoCancel 関数 … 17
 - 2.2.5 aec_GetVersion 関数 … 18
- 2.3 特 性 … 19
- 2.4 注意事項 … 20

第3章 インストレーション … 21

- 3.1 インストレーション手順 … 21
- 3.2 サンプル作成手順 … 21
- 3.3 ロケーションの変更 … 23
- 3.4 シンボル名規約 … 23

付録 サンプル・ソース (sample.asm) … 24

図の目次

図番号	タイトル, ページ
1-1	エコー・キャンセラ概念図 … 10
2-1	アプリケーション処理 … 13
2-2	特性測定構成図 … 19
3-1	サンプル・プログラム評価システム … 22

表の目次

表番号	タイトル, ページ
1-1	μ SAP77016-B01 の性能 … 12
2-1	AEC_CreateBuffer マクロ使用時のシンボル名とメモリ・サイズ … 15
2-2	特性測定条件 … 19
3-1	セクション名 … 23
3-2	シンボル名 … 23

第1章 概 説

1.1 ミドルウェア

ミドルウェアとは、プロセッサの性能をできるだけ引き出せるようにチューニングされたソフトウェア群で、従来ハードウェアが行っていた処理をソフトウェアで実現したものです。DSP という高性能プロセッサの出現、そして DSP が手軽にシステムに組み込める環境が整ってきたために、ミドルウェアという概念が現実のものとなってきました。

NEC では、 μ PD77016 ファミリ用にマルチメディア・システムを実現する要素技術を提供しています。たとえば音声コーデック、画像データの圧縮／伸長といったミドルウェアをタイムリに提供し、お客様のシステム開発を支援します。

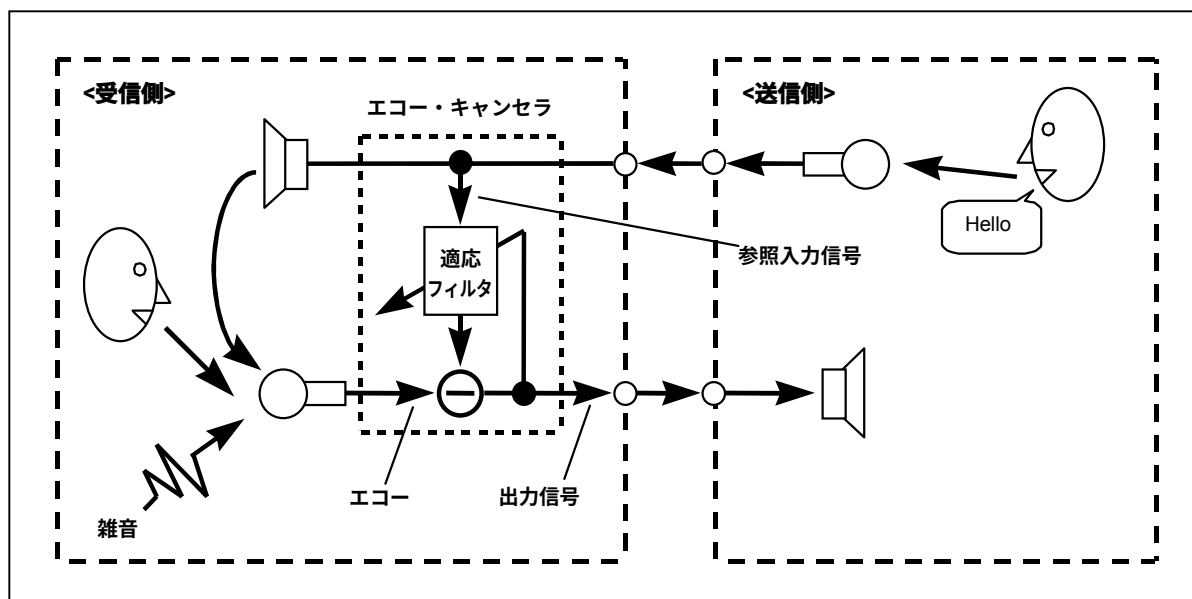
この製品は、音響エコー・キャンセル機能を提供するミドルウェアです。

1.2 エコー・キャンセラ

エコー・キャンセラには、ハンズ・フリー電話を使用中に発生する音響エコー（ハンズ・フリー電話を使用している環境で発生する反響）を予測し、受信側の音声信号（エコー＋音声＋雑音を含んだ音声信号）からエコー成分だけを除去する機能があります。

エコー・キャンセラを有効に活用できるアプリケーションとしては、自動車電話やテレビ会議などのハンズ・フリー・システムが考えられます。

図1-1 エコー・キャンセラ概念図



1.3 製品概要

1.3.1 特 徴

- NEC 独自方式による、雑音レベルの変動に強いエコー・キャンセル・アルゴリズムを採用
- 音声入出力データ（参照入力信号、エコー、出力信号）はすべて 16 ビット・リニア・データ
- 全二重方式エコー・キャンセラ
- 環境適応は適時学習タイプ
- エコー除去時間を任意に設定可能
- μ PD77016 スタータ・キット用ワーク・ベンチ（WB77016 Ver 2.21）Windows™ 版に対応

1.3.2 動作環境

★ (1) 動作対象 DSP

μ PD77015, 77016, 77017, 77018A, 77019, 77110, 77111, 77112, 77113, 77113A, 77114, 77115

(2) 必要メモリ・サイズ

用 途	サイズ（ワード）			
	インストラクション ROM	X データ RAM	Y データ RAM	合計
プログラム・コード	249	—	—	249
係数テーブル・バッファ	—	1[ワード／タップ]	1[ワード／タップ]	2[ワード／タップ]
遅延バッファ	—	1[ワード／タップ]	—	1[ワード／タップ]
ワーク・メモリ	—	—	19	19

備考 1. そのほかに、コピーライト格納用に 24 ワードが YRAM または YROM に必要です。

2. 係数テーブル、遅延バッファのメモリ・サイズは、1 タップあたりのサイズを表しています。

例：タップ数 512 の場合

係数テーブル・サイズ：1[ワード／タップ]×512 タップ×2=1024[ワード]

遅延バッファ・サイズ：1[ワード／タップ]×512 タップ=512[ワード]

(3) 対応 A/D, D/A スペック

A/D 2 チャンネル

D/A 1 チャンネル

16 ビット分解能

(4) ソフトウェア・ツール

DSP ツール： WB77016 Ver 2.21（Windows 版）

μ PD77016 スタータ・キット用 WB77016 Ver 2.21sk（Windows 版）

1.3.3 性 能

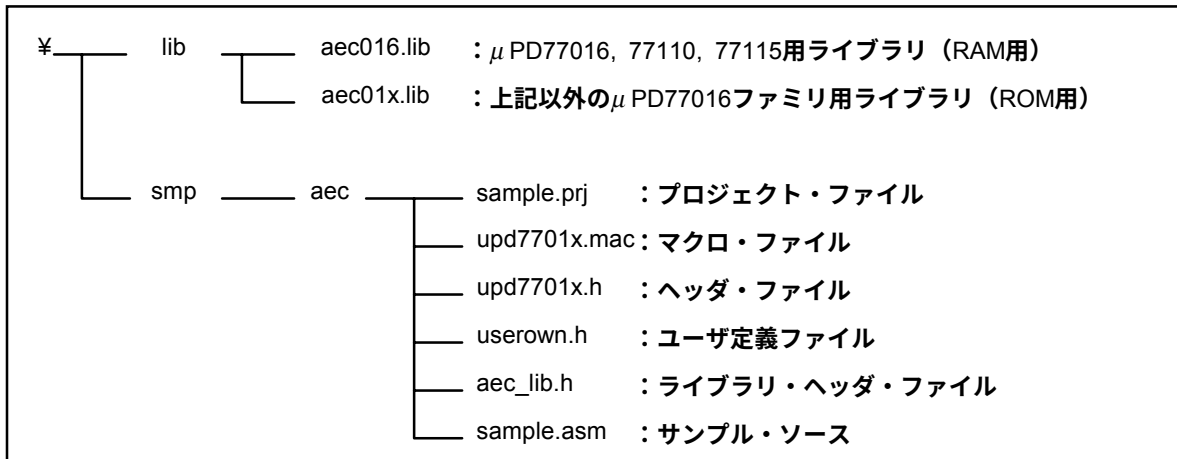
μ SAP77016-B01 の性能を次に示します。

表 1-1 μ SAP77016-B01 の性能

動作環境	DSP：μ PD77016（動作周波数：33 MHz） タップ数：512 タップ サンプリング周波数 $f_s = 8 \text{ kHz}$
性能	・実環境下で使用情况（例：自動車内など） 1 サンプリングに対するエコー・キャンセル処理時間＝約 $68.88 [\mu\text{s}]$ タップ数 $\times 4 [\text{cycle}] \times (\text{メモリ・ウェイト数} + 1) + (170-259) [\text{cycle}]$ エコー除去量 30 [dB] (TYP.) エコー除去時間 64 [ms]

★ 1.3.4 ディレクトリ構成

パッケージの内容を次に示します。

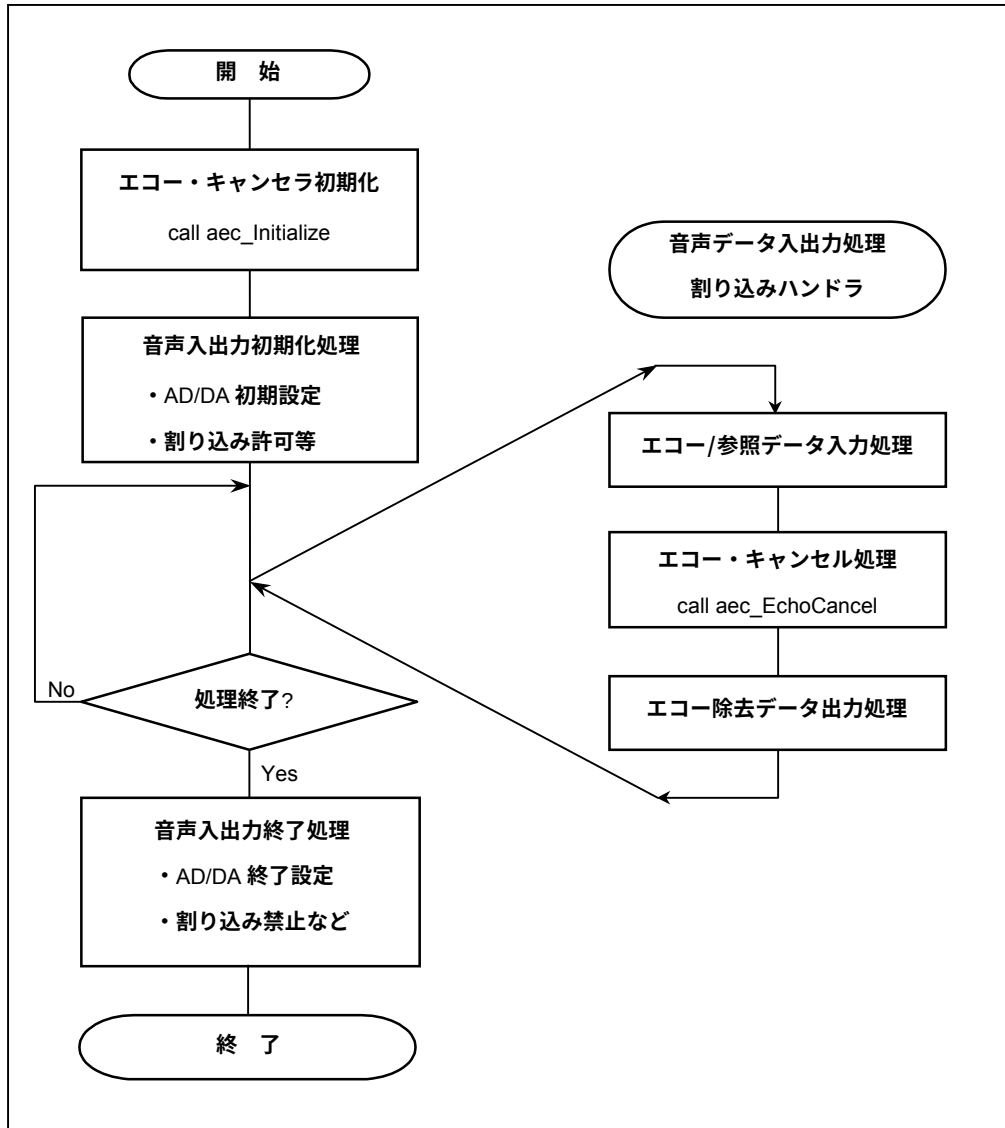


第2章 ライブラリ仕様

2.1 エコー・キャンセル処理フロー

μ PD77016 ファミリのシリアル I/F (SIO1, SIO2) に参照信号, エコー信号が入力される場合の処理を次に示します。

図 2-1 アプリケーション処理



2.2 関数仕様

2.2.1 メモリ構造

本ライブラリで必要とするメモリの構造について説明します。

・係数テーブル・バッファ

学習した係数を保存しておく領域です。1つの係数は32ビット長構成で、Xメモリ空間とYメモリ空間にそれぞれ下位16ビットと上位16ビットに分けて保存します。係数テーブルの大きさはタップ数に比例します。

例 `__AEC_WORK_X XRAMSEG`

`ch1_al_buff: ds tapnum ; 32bit coefficient of low word`

`__AEC_WORK_Y YRAMSEG`

`ch1_ah_buff: ds tapnum ; 32bit coefficient of high word`

・遅延バッファ

エコー・キャンセラに入力した参照信号を保存しておく領域です。遅延バッファを、Xメモリ空間に確保してください。また、確保した領域の先頭アドレスは、タップ数の大きさでアラインしてください[※]。遅延バッファの大きさはタップ数に比例します。

注 タップ数が 2^n (512, 256, 128 ...) の場合 → 2^n でアライン
 タップ数が $2^n + \alpha$ (500, 300, 200 ...) の場合 → 2^{n+1} でアライン

例 `__AEC_WORK_X XRAMSEG align at tapnum`

`ch1_z_buff: ds tapnum ; z buffer`

・ワーク・メモリ

エコー・キャンセラの各種状態を保存しておく領域です。ワーク・メモリは、Yメモリ空間に19ワードの大きさを確保してください。

例 `__AEC_WORK_Y YRAMSEG`

`ch1: ds 19 ; work memory`

- 注意 1.** `aec_initialize` 関数を呼び出したあとは、エコー・キャンセラを動作中に、係数テーブル・バッファ、遅延バッファ、ワーク・メモリの領域を壊さないようにしてください。これらの領域が壊れた場合、本ライブラリの正常な動作は保証できません。
- 2.** 係数テーブル領域（上位16ビット、下位16ビット）を両方とも外部メモリに配置することはできません。

2.2.2 マクロ

本ライブラリで使用するマクロは、aec_lib.h で定義されています。使用するためには、aec_lib.h をインクルードしてください。

•AEC_CreateBuffer マクロ

AEC_CreateBuffer マクロは、エコー・キャンセル処理に必要な 1 チャンネル分のメモリ領域を確保、宣言します。したがって、多チャンネル構成にする場合には、必要なチャンネルの数だけこのマクロを宣言してください。

【分類】 メモリ確保

【関数名】 AEC_CreateBuffer

【機能概要】 1 チャンネル分のメモリ領域を確保します。

【形式】 %AEC_CreateBuffer(name,tapnum)

【引き数】

型	引き数	説明
WORD	name	シンボル名
WORD	tapnum	タップ数

【返り値】 なし

【機能】 遅延バッファ、係数テーブル、ワーク・メモリをメモリ上に配置します。

例 シンボル名=ch1、タップ数 256 の設定で、AEC_CreateBuffer マクロを宣言した場合

```
%AEC_CreateBuffer(ch1, 256)
```

AEC_CreateBuffer マクロ使用時のシンボル名とメモリ・サイズは次のとおりです。

表 2-1 AEC_CreateBuffer マクロ使用時のシンボル名とメモリ・サイズ

シンボル名	サイズ[ワード]	X/Y 面	説明
ch1_z_buff	256	X	遅延バッファ領域
ch1_al_buff	256	X	係数テーブル領域（下位 16 ビット）
ch1_ah_buff	256	Y	係数テーブル領域（上位 16 ビット）
ch1	19	Y	ワーク・メモリの先頭アドレス

2.2.3 aec_Initialize 関数

aec_Initialize 関数は、各係数の設定、係数テーブル、遅延バッファの初期化を行う関数です。aec_Initialize 関数は、1 チャンネル分の初期化処理を行います。複数のチャンネルを初期化する場合は、各チャンネルごとに aec_Initialize 関数を呼び出して初期化してください。

【分類】 エコー・キャンセラ初期化処理

【関数名】 aec_Initialize

【機能概要】 エコー・キャンセラの使用する RAM 領域の初期化および各種パラメータを設定します。

【形式】 call aec_Initialize

【引き数】

型	引き数	説明
register	R0L	タップ数（除去時間×サンプリング周波数）
register	R1L	ステップ・サイズ（学習量）
register	R2L	α 係数（0-8）
register	R3L	β 係数（0.99-0.999）
register	DP0	係数テーブルの先頭アドレス（ロウ・ワード）
register	DP1	遅延バッファの先頭アドレス
register	DP4	係数テーブルの先頭アドレス（ハイ・ワード）
register	DP5	ワーク・メモリの先頭アドレス

【返り値】 なし

【機能】 本ライブラリの初期化、各種パラメータの設定などを行います。

【使用レジスタ】 R0, R1, R2, R3, DP0, DP1, DP4, DP5

【ループ段数】 1 段使用

【スタック・レベル】 0 段使用

注意 α , β 係数に許容範囲外の値を設定した場合、正常なエコー・キャンセル処理ができなくなる場合があります。

備考 雑音が大きくて、処理後の音声に乱れがある場合は、ステップ・サイズを小さくするか、 α 係数を大きくしてください。収束が遅い場合は、ステップ・サイズを大きくするか、 α 係数を小さくしてください。雑音パワーの変動が激しい場合は、 β 係数を小さくしてください（ただし、0.990-0.999 の範囲で設定してください）。ステップ・サイズ、 α 係数、 β 係数の推奨値は、次のとおりです。

ステップ・サイズ=0.1

α 係数=5

β 係数=0.995

2.2.4 aec_EchoCancel 関数

aec_EchoCancel 関数は、エコー信号と参照信号から、エコーを除去した信号を生成します。

【分類】 エコー・キャンセル処理部

【関数名】 aec_EchoCancel

【機能概要】 エコーを含んだ信号に対してエコー・キャンセル処理を行います。

【形式】 call aec_EchoCancel

【引き数】

型	引き数	説明
register	R0H	エコー信号 (16 ビット・リニア PCM)
register	R1H	参照信号 (16 ビット・リニア PCM)
register	DP0	係数テーブルの先頭アドレス (ロウ・ワード)
register	DP1	遅延バッファの先頭アドレス
register	DP4	係数テーブルの先頭アドレス (ハイ・ワード)
register	DP5	ワーク・メモリのアドレス

【返り値】

型	引き数	説明
register	R0H	エコー除去後の信号 (16 ビット・リニア PCM)

【機能】 参照信号からレプリカ・エコーを生成し、エコー信号からレプリカ・エコー信号を減算し、エコーを除去した信号を生成します。

【使用レジスタ】 R0, R1, R2, R3, R4, R5, DP0, DP1, DP4, DP5, DN1, DMX

【ループ段数】 1 段使用

【スタック・レベル】 0 段使用

2.2.5 aec_GetVersion 関数

aec_GetVersion 関数は、本ライブラリのバージョン情報を返します。

【分類】 エコー・キャンセラ・バージョン情報獲得関数

【関数名】 aec_GetVersion

【機能概要】 バージョン情報を返します。

【形式】 call aec_GetVersion

【引き数】 なし

【返り値】	型	返り値	説明
	register	R0H	メジャー・バージョン番号
	register	R0L	マイナ・バージョン番号

【機能】 本ライブラリのバージョン番号を 32 ビットの値で返します。

例 R0 = 0x00'0x0001'0x0100 の場合

バージョン：V1.01

【使用レジスタ】 R0

【ループ段数】 0 段使用

【スタック・レベル】 0 段使用

2.3 特 性

本ライブラリの参照信号レベル，エコー信号レベルに対するエコー除去量（ERLE）の特性を次に示します。

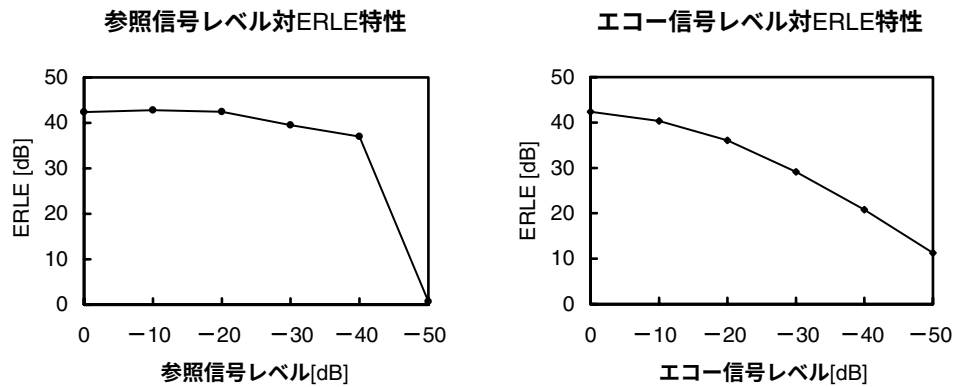
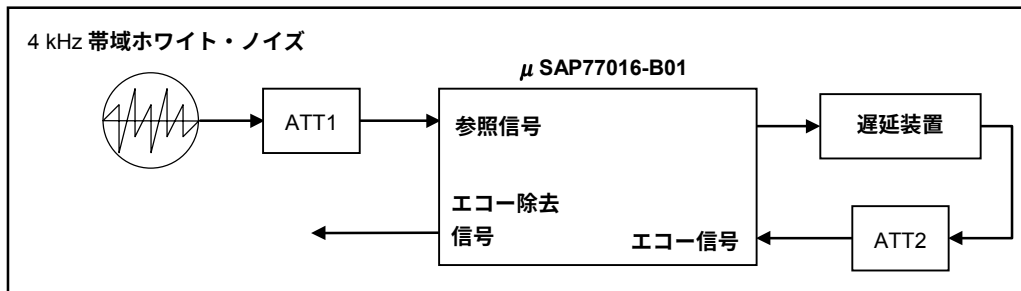


表 2-2 特性測定条件

	参照信号レベル対 ERLE	エコー信号レベル対 ERLE
参照信号	4 kHz 帯域ホワイト・ノイズ。 ホワイト・ノイズの平均パワーは 16 ビット量子化された値（-32768～32767）で 5.87×10^7 （平均値：7660）。	
遅延時間 [ms]	2	
ATT1 [dB]	0～-50	0
ATT2 [dB]	0	0～-50

図 2-2 特性測定構成図



2.4 注意事項

- ・エコー信号に含まれるエコーの戻り量は、減衰系になるように設定してください。エコーの戻り量が増幅系になるとエコーを消去できなくなります。
- ・参照信号は、CODEC でオーバフローしないようにレベル設定してください。
- ・参照信号には、トーン信号（DTMF 信号など）を入力しないでください。トーン信号を入力すると、エコーの除去量が低下します。
- ・エコー・キャンセラは、エコー・パス^{*}の変動が激しいと、エコー・パスの学習が収束しにくくなります。
- ・エコー・キャンセラを雑音環境下で使用的場合は、雑音のない環境下で使用的場合と比較して、エコー・パスの学習が収束するまでの時間が長くなる可能性があります。

注 エコー・パス：音声が反射して返ってくる経路

第3章 インストール

3.1 インストール手順

μ SAP77016-B01 の供給媒体は、3.5 インチ・フロッピー・ディスク（1.44 MB）です。ホスト・マシンへのインストール手順を次に示します。

- (1) 供給媒体をフロッピー・ディスク・ドライブにセットします。ソフトウェア・ツールが使用しているディレクトリ（例：C:\DSPTools）の下にファイルをコピーします。ここでは、A ドライブから C ドライブへコピーした場合を示します。

```
a:\>xcopy /s *.* c:\DSPTools <CR>
```

- (2) ファイルがコピーされたことを確認します。各ディレクトリについては、1.3.4 ディレクトリ構成を参照してください。

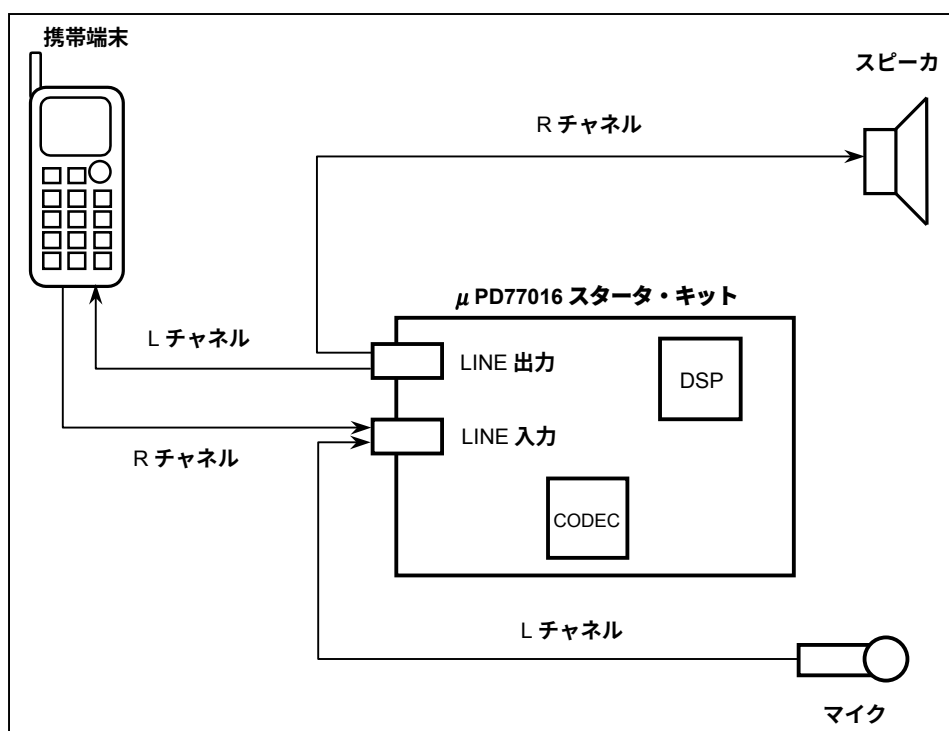
```
a:\>dir c:\DSPTools <CR>
```

3.2 サンプル作成手順

smp ディレクトリに、サンプル・プログラムを格納しています。sample.asm のソース・プログラムについては、**付録 サンプル・ソース（sample.asm）**を参照してください。

サンプル・プログラムは、マイク、スピーカなどを接続して、ハンズ・フリー・システムを評価することができます。なお、サンプル・プログラムは μ PD77016 スタータ・キット上で動作します。

図 3-1 サンプル・プログラム評価システム



次に、 μ SAP77016-B01 のサンプル・プログラムのビルド方法についての例を示します。

(1) WB77016 を起動します。

(2) sample.prj プロジェクト・ファイルを開きます。

例 Project メニューの Open Project コマンドで、C:\¥DSPTools¥smp¥aec¥sample.prj を指定します。

(3) ビルドを実行し、sample.lnk が生成されたことを確認します。

例 Make メニューの Build All コマンドを選択すると、sample.lnk ファイルができます。

(4) スタータ・キット用ディバッガ (ie77016s.exe) を使用して、ターゲット・システムにダウン・ロードして実行します。

注意 1. サンプル・ファイルの作成には、スタータ・キット用のワーク・ベンチを使用してください。

2. プロジェクト・ファイル (sample.prj) のファイル構成は絶対パスで指定しているので、ライブラリを「C:\¥DSPTools」以外のディレクトリにインストールした場合は、ファイル構成を設定し直してください。

3.3 ロケーションの変更

本ライブラリでは、表 3-1 に示すセクション名がつけられています。ユーザのターゲット・システムにあわせて、ロケーションを変更できます。

表 3-1 セクション名

セクション名	タイプ	内 容
__AEC_WORK_X	XRAMSEG	X メモリ領域データ RAM
__AEC_WORK_Y	YRAMSEG	Y メモリ領域データ RAM
__AEC_COPYRIGHT_Y	YROMSEG/YRAMSEG	コピーライト格納領域
__AEC_LIB	IROMSEG/IRAMSEG	エコー・キャンセラ・プログラム

3.4 シンボル名規約

本ライブラリで使用するシンボル名は、次に示す規約に従っています。ほかのアプリケーションと組み合わせて使用するときは、重複しないように注意してください。

表 3-2 シンボル名

分 類	規 約
関数	aec_XXXX
マクロ, 定数	AECXXXX

付録 サンプル・ソース (sample.asm)

```
/*-----*/
/*      File Information                                */
/*-----*/
/*      Name      : sample.asm                        */
/*      Type      : Assembler program module          */
/*      Version    : 1.00a                            */
/*      Date      : 1997 JAN 14                       */
/*      CPU       : uPD7701x Family                   */
/*      Assembler  : WB77016                           */
/*      About     : NEC uPD7701x Family Middle-Ware    */
/*                  Sample Program of Echo Canceller Library */
/*-----*/
/*      Copyright (C) NEC Corporation 1996            */
/*      NEC CONFIDENTIAL AND PROPRIETARY              */
/*      All rights reserved by NEC Corporation.        */
/*      Use of copyright notice does not evidence publication */
/*-----*/

#include "aec_lib.h"
#include "userown.h"
#include "upd7701x.mac"

/* =====
 *      Define CODEC(uPD63310) control register
 * ===== */
#define CODEC_CTL      *0x4000:X
#define CGAIN_ADR      *0x8000:X
#define CGAIN_DAT      *0x8001:X

/* =====
 *      Define echo canceller status
 * ===== */
#define TAPS      512
#define U0      0.1
#define ALFA      5
#define BETA      0.995

/* =====
 *      Allocate echo canceller buffer
 * ===== */
%AEC_CreateBuffer(ch1, TAPS);

/* =====
 *      Allocate work buffer
 * ===== */
WORK_Y  YRAMSEG
R0TMP:  ds      3
cnt:    ds      1
flag:   ds      1
echo:   ds      1
ref:    ds      1
err:    ds      1

#define CNT      *cnt:Y
#define FLAG     *flag:Y
#define ECHO     *echo:Y
```

```
#define REF      *ref:Y
#define ERR      *err:Y

/* =====
 *      Vector registration
 * ===== */
%BeginVector(StartUp)                                ;Register start up routine
    %NotUseVector(VectorINT1)                        ;
    %NotUseVector(VectorINT2)                        ;
    %NotUseVector(VectorINT3)                        ;
    %NotUseVector(VectorINT4)                        ;
    %RegistVector(VectorSI1, SI1Handler)              ;Regist SI1 handler
    %NotUseVector(VectorSO1)                          ;
    %NotUseVector(VectorSI2)                          ;
    %NotUseVector(VectorSO2)                          ;
    %NotUseVector(VectorHI)                          ;
    %NotUseVector(VectorHO)                          ;
%EndVector

/* =====
 *      Sample programe code section
 * ===== */
SAMPLE_CODE      IMSEG at 0x240
StartUp:
    ;;=====;;
    ;;      Initialize Register & Peripheral Units      ;;
    ;;=====;;
    %ClearAllRegister                                ;Clear all uPD7701x register
    %SetIWTR(0x0054)                                ;Set instruction ROM/RAM wait cycle
    %SetDWTR(0x3c3c)                                ;Set data ROM/RAM wait cycle
    %SetPort(P0_OUT_MODE|P0_L_LEVEL) ;Set P0 output mode & low level
    %SetPort(P1_OUT_MODE|P1_L_LEVEL) ;Set P1 output mode & low level
    %SetPort(P2_IN_MODE)                            ;Set P2 output mode & low level
    %SetPort(P3_IN_MODE)                            ;Set P3 input mode
    ;;=====;;
    ;;      Initialize CODEC chip(uPD63310)              ;;
    ;;=====;;
    clr(R0)                                          ;
    clr(R1)                                          ;
    R1L = 0x0002                                    ;
    CGAIN_ADR = R0L                                ;ch1-L
    nop                                             ;
    CGAIN_DAT = R1L                                ;
    R0L = 0x0001                                    ;ch1-R
    CGAIN_ADR = R0L                                ;
    nop                                             ;
    CGAIN_DAT = R1L                                ;
    R0L = 17                                         ;out-L
    CGAIN_ADR = R0L                                ;
    R0L = 0x0001                                    ;
    CGAIN_DAT = R0L                                ;
    R0L = 18                                         ;out-R
    CGAIN_ADR = R0L                                ;
    R0L = 0x0001                                    ;
    CGAIN_DAT = R0L                                ;
    R0L = 0x000f                                    ;
    CODEC_CTL = R0L                                ;
    ;;=====;;
    ;;      Initialize Echo cannceller module          ;;
    ;;=====;;
    R0L = TAPS                                      ;Initialize echo cancel module
```

```

R1L = U0
R2L = ALFA
R3L = BETA
DP0 = ch1_al_buff
DP4 = ch1_ah_buff
DP1 = ch1_z_buff
DP5 = ch1
call aec_initialize
;;=====;;
;;      Initialize work/flag memory
;;=====;;
clr(R0)
CNT = R0L
FLAG = R0L
ECHO = R0L
REF = R0L
;;=====;;
;;      Initialize Serial I/F
;;=====;;
R0L = 0x0200 ;Initialize Serial I/F 1
*SST1:X = R0L
;

%DisableMask(SR_ALL) ;Disable all interrupt mask
%EnableMask(SR_SI1) ;Enable SI1 interrupt mask
%EnableInterrupt ;Enable interrupt
;;=====;;
;;      Main routine
;;=====;;
loop1:
nop ;Wait Echo canceller start flag
R0 = FLAG
if(R0 == 0) jmp loop1
R0 = ECHO ;Set echo signal to R0
R1 = REF ;Set reference signal to R1
DP0 = ch1_al_buff ;Set coefficient table start address
DP4 = ch1_ah_buff
DP5 = ch1 ;Set AECINFO struck start address
call aec_EchoCancel ;Echo cancel process
ERR = R0H ;Get echo removabled signal
clr(R0)
FLAG = R0H
jmp loop1

;;=====;;
;;      SI1 handler
;;=====;;
SI1Handler:
*R0TMP+0:Y = R0L ;Save work register of R0
*R0TMP+1:Y = R0H
*R0TMP+2:Y = R0E
R0L = *PDT:X ;if P3 is low level, no echo canceller.
R0 = R0 & 0x0008
if(R0 == 0) jmp no_cancel

clr(R0) ;Echo cancel process
R0L = CNT
R0 = R0 ^ 0x0001
CNT = R0L
if(R0 == 0) jmp R_channel
L_channel:
R0 = *SDT1:X ;Get L-channel signal

```

```

        ECHO = R0H                ;Save echo signal
        R0 = ERR                  ;Load output signal
        jmp sil_end              ;
R_channel:
        R0 = FLAG                 ;
        if(R0 == 0) jmp no_error  ;
        %SetPort(P0_H_LEVEL)     ;
no_error:
        R0 = *SDT1:X              ;Get R-channel signal
        REF = R0H                 ;Save reference signal
        R0L = *PDT:X              ;
        R0 = R0 & 0x0004          ;
        if(R0 != 0) jmp $+3       ;
        R0H = ERR                 ;
        jmp $+2                   ;
        R0H = REF                 ;
        R0L = 0x0001              ;Set echo canceller start flag
        FLAG = R0L                ;
sil_end:
        *SDT1:X = R0H             ;Output signal
        R0E = *R0TMP+2:Y          ;Save work register of R0
        R0H = *R0TMP+1:Y          ;
        R0L = *R0TMP+0:Y          ;
        reti                      ;

no_cancel:                        ;No cancel process
        clr(R0)                   ;
        R0L = CNT                 ;
        R0 = R0 ^ 0x0001          ;
        CNT = R0L                 ;
        if(R0 == 0) jmp R_channel2 ;
        R0 = *SDT1:X              ;Get L-channel signal
        ECHO = R0H                ;
        jmp sil_end              ;
R_channel2:
        R0 = *SDT1:X              ;Get R-channel signal
        REF = R0H                 ;
        R0H = ECHO                ;
        jmp sil_end              ;

end

```

[× モ]

[× モ]

—— お問い合わせ先 ——

【技術的なお問い合わせ先】

NEC半導体テクニカルホットライン
(電話：午前 9:00～12:00, 午後 1:00～5:00)

電 話 : 044-435-9494
FAX : 044-435-9608
E-mail : s-info@saed.tmg.nec.co.jp

【営業関係お問い合わせ先】

第一販売事業部

東 京 (03)3798-6106, 6107,
6108
大 阪 (06)6945-3178, 3200,
3208, 3212
仙 台 (022)267-8740
郡 山 (024)923-5591
千 葉 (043)238-8116

第二販売事業部

東 京 (03)3798-6110, 6111,
6112
立 川 (042)526-5981, 6167
松 本 (0263)35-1662
静 岡 (054)254-4794
金 沢 (076)232-7303
松 山 (089)945-4149

第三販売事業部

東 京 (03)3798-6151, 6155, 6586,
1622, 1623, 6156
水 戸 (029)226-1702
広 島 (082)242-5504
前 橋 (027)243-6060
鳥 取 (0857)27-5313
太 田 (0276)46-4014
名古屋 (052)222-2170, 2190
福 岡 (092)261-2806

【資料の請求先】

上記営業関係お問い合わせ先またはNEC特約店へお申しつけください。

【NECエレクトロニクス ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.ic.nec.co.jp/>

キリトリ

[ドキュメント名] μ SAP77016-B01 ユーザーズ・マニュアル

【お名前など】（さしつかえのない範囲で）

2000.6